Biological molecular function: Beyond homology

Jason E. McDermott

Integrated Discovery Sciences Directorate, Pacific Northwest National Laboratory, Richland, Washington, USA
Jason.McDermott@pnnl.gov

Yana Bromberg
Department of Biology, Emory University, Atlanta, Georgia, USA
yana.bromberg@emory.edu

Hannah Carter
Department of Medicine, Division of Medical Genetics, University of California San Diego, La Jolla, California, USA
hkcarter@health.ucsd.edu

Travis Wheeler
Pharmacy Practice & Science Department, University of Arizona, Tucson, Arizona, USA
twheeler@arizona.edu

Computational annotation of molecular function has long relied on a powerful but fundamentally retrospective strategy: identify a sequence or structure similar to a characterized molecule and transfer what is known. Improved alignment algorithms, structure prediction, and biological language models (LMs) have extended the reach of this paradigm, yet large numbers of genes and proteins, particularly those from viruses, microbiomes, and non-model organisms, remain without useful functional descriptions. Moreover, similarity does not by itself resolve how molecular activity changes with context, i.e. cellular state, genomic neighborhood, interaction network, etc. This session explores complementary routes to function annotation beyond exclusive reliance on homology. The papers in the session use genomic neighborhoods, genetic fitness measurements, knowledge graphs, biological network topology, metabolic feasibility, learned sequence representations, open-ended reaction generation, and context-dependent descriptions of retained protein elements. Together, they suggest that "beyond homology" is not a single replacement technology but rather a shift toward integrating evidence across scales, representing function in context, evaluating methods under genuinely difficult generalization settings, and reporting uncertainty or failure when available data do not support a functional claim. The session papers also expose unresolved challenges: context can improve system-level coherence without identifying the responsible protein; generative models can produce plausible chemistry without assigning it correctly; and calibrated abstention may be more informative than a forced prediction. This session frames these advances and limitations as a research agenda for moving from annotation transfer toward mechanistic and testable hypotheses of biological function.

Keywords: molecular function; protein function prediction; biological dark matter; genomic context; knowledge graphs; metabolic modeling; network biology; protein language models; uncertainty calibration.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

1. Introduction

Advances in sequencing technology have vastly outpaced experimental characterization of protein function. It is estimated that in well-characterized organisms 30-60% of proteins can not be assigned specific functional roles. The situation is even worse in metagenomes and viromes, where diversity and evolutionary distance from the known collection of molecules is a drastically limiting factor (Hatfull, 2015; Lobb, et al., 2020; Schloss and Handelsman, 2005). The vast majority of functional annotations are assigned on the basis of gene/protein homology, captured by sequence similarity. Sequence alignment, profile Hidden Markov Models, and increasingly sensitive database-search methods are used across the sequence landscape (Altschul, et al., 1990; Buchfink, et al., 2021; Eddy, 2011; Roddy, et al., 2024; Steinegger and Soding, 2017). Predicted three-dimensional structures now permit searches across structural space, recovering relationships that may not be apparent from primary sequence (van Kempen, et al., 2023) and emerging protein LMs aim to identify distributed sequence patterns to produce representations useful for many downstream predictive tasks (Bileschi, et al., 2022; Rives, et al., 2021). These advances have greatly expanded the detectable neighborhood around known biology.

Yet expanding that neighborhood is not the same as understanding function. Most similarity-based approaches remain anchored to an existing annotation: a new molecule is interpreted through resemblance to something previously studied. Their success therefore depends on the coverage, accuracy, and granularity of reference knowledge. Existing labels often describe only one layer of function, for example catalytic activity, while omitting the conditions, partners, location, regulation, or phenotypic consequences, that determine what the molecule does in a living system.

The phrase “molecular function” itself hides several distinct questions. What chemical transformation can a protein catalyze? What substrate does it act upon? In which compartment, condition, or organism does that activity matter? What role does it play in a pathway or interaction network? How do regulatory elements or domain combinations alter the effect of an otherwise familiar component? A protein can retain a recognizable domain yet have a wholly different biological role when found in a different context (Jeffery, 1999; Piatigorsky and Wistow, 1991) or included within a different domain architecture (Bashton and Chothia, 2007; Todd, et al., 2001). Conversely, proteins without detectable homology can converge on related system-level roles (McDermott, et al., 2019). Function is therefore not only a property of an isolated sequence; it is an interaction among molecular capabilities, environment, and the level at which the question is asked.

This session, “Biological molecular function: Beyond homology,” builds on the PSB 2026 session on methods and benchmarks for finding function in biological dark matter (McDermott, et al., 2026). Discussion at that session identified the dependence of most annotation strategies on sequence or structural similarity as both an enduring strength and an important frontier. The challenge is not to discard homology, as evolutionary relationships remain among the strongest sources of functional evidence and establish a demanding baseline (Hamp, et al., 2013). Rather, the goal is to determine what additional evidence becomes informative where direct annotation transfer becomes unreliable, incomplete, or conceptually insufficient.

The accepted papers in this session span microbes, bacteriophages, metagenomes, metabolism, fusion proteins, and cancer signaling. Their methods differ substantially, but they collectively move away from treating a protein as an isolated sequence requiring a single inherited label. They use

neighboring genes, condition-specific mutant fitness, heterogeneous knowledge graphs, network topology, organism-level biochemical constraints, and retained regulatory elements. Other contributions ask whether learned molecular representations can support direct enzyme discovery or generation of biochemical reactions, RNA–protein interactions, pharmacogenomics, and structure-based cryptic-pocket detection, extending context from molecular partners and substrates to explicit local geometry. Importantly, these papers do not present a simple story of homology being replaced. Several report that performance declines under stricter generalization tests, that improvements at the system level do not translate to protein-level labels, or that generated chemistry is not correctly assigned. These outcomes help improve problem definition and point toward evaluation, uncertainty, and abstention as essential components of functional inference.

2. Session Contributions

The session papers can be organized around four complementary ideas: (1) context can supply functional evidence that is absent from an isolated sequence, (2) models can infer biochemical capability directly from molecule representations, while system-level constraints can filter molecular predictions for biological plausibility, (3) progress depends on representations that capture context-dependent effects and on uncertainty estimates to distinguish supported hypotheses from extrapolation, and (4) molecular partners, substrates, and local structural environments define function in ways that require interaction-aware, condition-specific, or auditable representations.

2.1. Function from genomic, phenotypic, and network context

Genomic organization is one of the most direct forms of non-homology evidence available at scale. In prokaryotes and viruses, neighboring genes often participate in related processes or functional modules. Hoang et al. introduce Synnotate, which masks a target gene and annotates it using the standardized annotations and arrangement of genes in a 21-gene neighborhood (Hoang, et al., 2027). The model uses a sparse-attention transformer and provides confidence estimates, attribution to individual neighbors, and retrieved reference neighborhoods. Under genus-disjoint cross-validation, it reaches high accuracy in annotating prokaryotic and bacteriophage genes. Combining model confidence with a synteny score further defines trusted regions. Synnotate is able to confidently annotate nearly 400K prokaryotic dark genes and 42K of unannotated phage proteins.

The method identifies candidates missed by direct sequence and structural searches while retaining interpretable evidence for each prediction demonstrating a practical model for inference beyond homology: use an independent contextual signal, test it across taxonomic boundaries, expose the evidence supporting a prediction, and provide boundaries on the calibrated region. Its application to bacteriophages is especially relevant because viral genomes contain extensive uncharacterized sequence space and often lack tractable experimental systems.

Experimental phenotypes provide another layer of context. Anthony et al. describe GIMME, a heterogeneous biological knowledge graph (KG) integrating nearly eight million random-barcode transposon sequencing fitness measurements with genes, experiments, strains, experimental metadata, metabolic knowledge, text embeddings, and protein language-model embeddings (Anthony, et al., 2027). Their graph-attention model, GFM1, predicts condition-specific gene knockout-determined fitness across ten *Pseudomonas* strains. Relative to a matched multilayer

perceptron, message passing significantly improves predictions for held-out experiments and held-out genes. The largest benefits are observed for highly connected, conditionally essential genes, recovering many new true-essential gene–condition observations.

These results illustrate the value of combining molecular representations with experimental phenotype under many conditions, but also point out limitations. The graph model's advantage disappears under a more stringent cluster-aware split, indicating that successful message passing still depends on informative connections to related training genes. Thus, context can extend functional prediction beyond isolated sequence similarity without eliminating dependence on the structure and coverage of known biology. The study reveals this limitation by using increasingly demanding splits, an evaluation practice that should become standard for claims of generalization beyond homology.

At a different scale, Jacobs et al. ask whether non-homologous cancer driver genes can be grouped by convergent roles in signaling networks (Jacobs, et al., 2027). INSPIRE performs within-species alignment of signaling subnetworks associated with colorectal cancer driver genes, comparing interaction patterns and topology rather than relying on gene-level resemblance alone. Consensus clusters show stronger within-group CRISPR co-essentiality than between-group pairs and identify exploratory epistatic relationships. Aggregating genes into the inferred modules also reveals significant mutual exclusivity that is not apparent in single-gene analysis. Although molecular annotations such as Pfam and Gene Ontology contribute to alignment of interaction endpoints, the relationships among drivers emerge from network organization and shared system-level roles. INSPIRE predicts the impact that distinct genes have on a common network program, broadening the notion of function beyond transfer of a biochemical label.

Together, Synnotate, GIMME, and INSPIRE move across local genome organization, condition-specific phenotype, and cellular signaling topology. They show that context can identify functional patterns at multiple scales, but also that the strength of functional inference depends on graph connectivity, available measurements, and the biological level at which function is defined.

2.2. From molecular representations to biochemical capability

An ambitious route beyond homology is to predict general biochemical activity directly from learned representations, common to specific functions, rather than retrieving a label from a similar annotated protein. REFinder, presented by Prabakaran and Bromberg, targets enzyme discovery in metagenomic sequence data (Prabakaran and Bromberg, 2027). Its first stage uses REBEAN, an alignment-free DNA language model, to classify reads into broad Enzyme Commission (EC) classes. Reads predicted to encode enzymes are then routed into focused assembly, reducing the need to assemble an entire metagenome and functionally labeling previously unseen, "dark matter" protein genes/sequences. Optional sequence-, domain-, structure-, metal-binding-, and representation-reliability analyses provide additional evidence for the resulting candidates.

Across human saliva and murine gut metagenomes, REFinder recovers approximately 2-3 times more enzymes than sequence similarity alone producing many annotated sequences without close Swiss-Prot matches. Note that in one sample, thousands of these sequence-divergent candidate oxidoreductases had strong structural matches to known enzyme folds. Class-targeted assembly can cost up to 6.4-fold less than whole-metagenome assembly in terms of computation and can recover poorly covered (lower-depth sequencing) genes. The framework therefore links functional inference

to the design of the computational discovery process itself: predicted activity determines which reads are assembled.

The interpretation of these candidates requires care. Structural correspondence supports a fold but does not establish catalytic activity, broad EC classes provide limited specificity, and the present evaluation covers two comparatively well-studied microbiome types. Nonetheless, REFinder demonstrates how alignment-free learned signal can prioritize previously inaccessible sequence space and generate testable enzyme candidates at metagenomic scale.

Balamukundhan et al. push direct inference further with RxnPred, an encoder–decoder transformer architecture that generates reaction SMILES from protein sequence and structure embeddings (Balamukundhan, et al., 2027). The model combines ESM-2 sequence representations, ESM-IF1 structural representations, and ChemBERTa metabolite representations. Unlike classification into an EC category or retrieval from a supplied reaction pool, RxnPred attempts open-ended generation of the biochemical transformation itself. This formulation is appealing because the reaction is closer to mechanistic function than a broad class label and because generation could, in principle, describe chemistry absent from the training set.

RxnPred is competitive for reactions observed during training, trading precision for increased recall. It can also generate previously unseen reaction strings, including some valid reactions found in the Rhea reaction knowledgebase (Bansal, et al., 2022). However, these reactions are generally assigned to the wrong enzyme, and the model does not recover the correct unseen reactions in evaluations where retrieval methods can rank them because those reactions are supplied at inference. Performance also decreases as test proteins become less similar to training proteins. These findings distinguish two capabilities that are easily conflated: generating chemically plausible reactions and grounding a particular reaction in the molecular determinants of a particular protein. The first is emerging; the second remains a central challenge.

METEOR, developed by Niu and Hoehndorf, addresses biochemical function from the opposite direction (Niu and Hoehndorf, 2027). Instead of asking only which reaction a protein resembles, it asks which collection of predicted reactions can coexist in a mass-balanced metabolic network capable of supporting growth. METEOR aggregates protein-level EC predictions into reaction confidence and uses confidence-weighted mixed-integer optimization under mass-balance and biomass-feasibility constraints to select an organism-level reaction repertoire. Across enzyme predictors and bacterial genomes, the resulting networks contain significantly fewer dead-end metabolites than raw thresholded predictions. All repaired METEOR networks grow under the defined medium, whereas none of the raw thresholded networks do. The method also recovers more curated sub-threshold EC assignments vs. conventional thresholding followed by repair.

The constraints improve global biochemical coherence but do not consistently improve per-protein annotation accuracy or pathway inference against a size-matched baseline, and selected reactions are not assigned to specific proteins or paralogs. This distinction is consequential. Organism-level feasibility can reveal that a capability is probably present without identifying the molecule that carries it out. METEOR therefore suggests that function prediction should sometimes be decomposed into at least two tasks: determining the biological capabilities required by the system and attributing those capabilities to molecular actors.

Taken together, REFinder, RxnPred, and METEOR span molecular function discovery, generation, and constraint. Learned representations expand the space of candidate activities; generative models test whether chemistry can be produced without a closed label set; and metabolic models enforce physical and physiological consistency. Their limitations suggest that robust enzyme-function inference will likely require these forms of evidence to interact rather than compete – for example, using generative proposals as hypotheses, genomic or structural evidence for molecular grounding, and system constraints to evaluate organism-level plausibility.

2.3. Representing context-dependent effects and calibrated uncertainty

Moving beyond homology also requires better representations of what a functional annotation means. Murakami examines fusion proteins, for which identifying retained domains or protein segments may be straightforward, while predicting the biological consequence of their combination is not (Murakami, 2027). The proposed signed element-contribution framework represents fusion function as a sum of context-dependent contributions from retained elements. Elements may act as effectors, autoinhibitory regions, oligomerization modules, or neutral components, and their default positive or negative contributions can reverse under specified conditions.

This representation addresses a gap in common resources. A fusion can preserve an enzymatic or interaction domain while losing an autoinhibitory segment, converting a familiar molecular component into a constitutively active or pathogenic one. Here, homology may correctly identify every retained part yet fail to describe the sign of its contribution in the new architecture. The study finds that retention of effector elements and loss of autoinhibitory elements depart from random breakpoint expectations, supporting the relevance of signed roles. It also highlights that autoinhibitory regions remain poorly represented in machine-readable annotations.

The work uses conformal prediction to retrieve precedents and does not provide a prediction when no known mechanism is sufficiently similar (~15% of the queries). At the same time, mechanism-class identification does not outperform baseline because the current role vectors mostly encode how much of a protein is retained rather than which specific mechanism remains. This negative result is informative: no uncertainty procedure can recover distinctions that the representation and source annotations do not contain. Progress therefore depends not only on larger models but also on richer, explicitly mechanistic data resources.

Calibration and limitation connect this work to the other contributions. Synnotate limits high-confidence annotation to a trusted region. GFM1's split-dependent performance identifies settings where graph context does not support generalization. RxnPred exposes the difference between plausible output and correct molecular assignment. METEOR separates system coherence from protein-level certainty. Across these papers, a responsible beyond-homology method is not one that always emits a label. It is one that communicates which evidence supports a claim, the biological scale at which the claim is valid, and when the available evidence is insufficient.

2.4. Molecular interactions, substrate-dependent effects, and auditable structural inference

Islam et al. investigate RNA–protein interaction prediction with Qfusion, which combines fixed ESM-2 protein and RNA-FM RNA embeddings through an eight-qubit, eight-layer circuit with cross-register correlators (Islam, et al., 2027). In family-and-clan-disjoint evaluation, Qfusion attains

a mean AUROC somewhat higher than a parameter-matched classical control. The AUPRC difference from the matched control is significant across evaluation folds, whereas the AUROC difference is not. Although the circuit itself has no trainable parameters, the complete model includes trainable projection and readout components. Because evaluation uses a classical statevector simulator, the study claims neither runtime nor sample-complexity quantum advantage. Its contribution is a competitive cross-modal coupling mechanism whose ablations identify cross-register correlators as important to the observed gain.

Xu et al. examine variant-effect prediction for six cytochrome P450 pharmacogenes, with activity modeling focused on CYP2C9 (Xu, et al., 2027). Ambiguous AlphaMissense calls are enriched across these proteins. Within the ambiguous CYP2C9 class, AlphaMissense scores correlate weakly with measured activity, while ESM-2 retains some signal. A CYP2C9-specific nearest-neighbor model combining ESM-2 embeddings with AlphaMissense and ESM-2 scores improves overall activity correlation and ambiguous-class correlation. These biochemical improvements do not reliably transfer to sparse clinical labels. Moreover, more than a tenth of the variants annotated for multiple drugs have distinct effects across substrates. This finding motivates substrate-conditioned prediction as a future direction rather than establishing a validated substrate-aware model, and underscores that biochemical activity prediction and clinical interpretation are distinct targets.

Yu and Wu present an algebraic counting field for detecting cryptic pocket-lining residues from apo protein structures. At inference, the method uses no sequence search, structural-template transfer, or protein language model. It expands explicit geometric, physicochemical, and topological quantities into discretized neighborhood features, with scores reconstructable from lookup-table counts, integer multiplicities, and spatial aggregation. This provides exact numerical traceability, although not by itself a causal or mechanistic explanation.

On a locked external set of apo–holo pairs, the method performance was somewhat better than that of P2Rank. A second predeclared external evaluation did not confirm improvement from an expanded feature set. The principal result is therefore competitive prediction with an auditable scoring path and explicit external validation, not universal state-of-the-art performance. The output identifies benchmark cryptic residues, not ligand identity, affinity, druggability, or therapeutic efficacy.

3. Conclusions

The papers in this session present “beyond homology” not as a rejection of evolutionary inference but as an expansion of the evidence and representations used to understand molecular function. Sequence and structural similarity remain indispensable. However, they are incomplete when no characterized relative is detectable, when an annotation lacks mechanistic specificity, or when biological consequence depends on context that an isolated sequence cannot encode.

Several common principles emerge. First, context is measurable and computationally useful. Gene neighborhoods, perturbation phenotypes, interaction networks, and organism-level metabolic constraints each add information that cannot be reduced to pairwise sequence similarity. Second, function exists at multiple scales. A model may correctly infer a system capability without assigning it to a protein, or may identify a network role without proposing a biochemical mechanism. Claims

and benchmarks should state clearly which level is being predicted. Third, direct and generative models require grounding. Producing a plausible enzyme class or reaction is not equivalent to establishing that a particular molecule performs it. Independent genomic, structural, chemical, phenotypic, and physiological evidence will be needed to convert predictions into compelling hypotheses. Molecular partners, substrate identity, and local structural geometry further show that the relevant context may reside at the interface between molecules or in a specific physical environment.

Finally, rigorous negative results, calibrated confidence, and clearly accounting for limitations are central to progress. The hardest test of a beyond-homology method is not performance on a random split populated by close relatives. It is whether useful information survives lineage separation, sequence-cluster separation, unseen chemistry, sparse context, or a missing mechanistic precedent. The studies represented here make those boundaries visible. Collectively, they point toward a future in which functional annotation is less often a single inherited label and more often an evidence-weighted, context-specific, and testable account of what a molecule can do, where it does it, and how confidently we know.

Acknowledgments

Funding to support this work was provided to J.E.M. by Pacific Northwest National Laboratory's Predictive Phenomics Initiative conducted under the Laboratory Directed Research and Development Program. PNNL is a multiprogram national laboratory operated by Battelle for the U.S. Department of Energy under Contract No. DE-AC05-76RL01830. Funding to support J.E.M. and T.J.W. was provided under NIH/NIDCR U01DE034176-01. Y.B. was supported by was supported by National Science Foundation award number 2310114.

References

Altschul, S.F.*, et al.* Basic local alignment search tool. *J Mol Biol* 1990;215(3):403-410.

Anthony, W.*, et al.* Predicting microbial gene fitness with a biological knowledge graph. *Pacific Symposium on Biocomputing* 2027.

Balamukundhan, S., Jagarlapudi, S.N. and Raman, K. RxnPred: A generative framework for enzymatic reaction prediction. *Pacific Symposium on Biocomputing* 2027.

Bansal, P.*, et al.* Rhea, the reaction knowledgebase in 2022. *Nucleic Acids Res* 2022;50(D1):D693-D700.

Bashton, M. and Chothia, C. The generation of new protein functions by the combination of domains. *Structure* 2007;15(1):85-99.

Bileschi, M.L.*, et al.* Using deep learning to annotate the protein universe. *Nat Biotechnol* 2022;40(6):932-937.

Buchfink, B., Reuter, K. and Drost, H.G. Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nat Methods* 2021;18(4):366-368.

Eddy, S.R. Accelerated Profile HMM Searches. *PLoS Comput Biol* 2011;7(10):e1002195.

Hamp, T.*, et al.* Homology-based inference sets the bar high for protein function prediction. *BMC Bioinformatics* 2013;14 Suppl 3(Suppl 3):S7.

Hatfull, G.F. Dark Matter of the Biosphere: the Amazing World of Bacteriophage Diversity. *J Virol* 2015;89(16):8107-8110.

Hoang, K.*, et al.* Synnotate: Genomic-context based annotation of the prokaryotic and viral dark proteome. *Pacific Symposium on Biocomputing* 2027.

Islam, A., Subramanian, G. and Singh, R. RNA-Protein Interaction Prediction Using Hybrid Quantum-Classical Fusion. *Pacific Symposium on Biocomputing* 2027.
Jacobs, C.*, et al.* INSPIRE: A network-based method for quantifying functional convergence in cancer driver genes. *Pacific Symposium on Biocomputing* 2027.
Jeffery, C.J. Moonlighting proteins. *Trends Biochem Sci* 1999;24(1):8-11.
Lobb, B.*, et al.* An assessment of genome annotation coverage across the bacterial tree of life. *Microb Genom* 2020;6(3).
McDermott, J.*, et al.* Prediction of bacterial E3 ubiquitin ligase effectors using reduced amino acid peptide fingerprinting. *PeerJ* 2019;7(e7055).
McDermott, J.E.*, et al.* Session Introduction: Biological molecular function: methods and benchmarks for finding function in biological dark matter. *Pac Symp Biocomput* 2026;31:417-424.
Murakami, K. A signed element-contribution framework for describing the function of fusion proteins. *Pacific Symposium on Biocomputing* 2027.
Niu, K. and Hoehndorf, R. METEOR: Evidence-weighted reaction selection reconciles enzyme prediction with growth feasibility. *Pacific Symposium on Biocomputing* 2027.
Piatigorsky, J. and Wistow, G. The recruitment of crystallins: new functions precede gene duplication. *Science* 1991;252(5009):1078-1079.
Prabakaran, R. and Bromberg, Y. REFinder: Mining new enzymes from metagenomes. *Pacific Symposium on Biocomputing* 2027.
Rives, A.*, et al.* Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc Natl Acad Sci U S A* 2021;118(15).
Roddy, J.W., Rich, D.H. and Wheeler, T.J. nail: software for high-speed, high-sensitivity protein sequence annotation. *bioRxiv* 2024.
Schloss, P.D. and Handelsman, J. Metagenomics for studying unculturable microorganisms: cutting the Gordian knot. *Genome Biol* 2005;6(8):229.
Steinegger, M. and Soding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat Biotechnol* 2017;35(11):1026-1028.
Todd, A.E., Orengo, C.A. and Thornton, J.M. Evolution of function in protein superfamilies, from a structural perspective. *J Mol Biol* 2001;307(4):1113-1143.
van Kempen, M.*, et al.* Fast and accurate protein structure search with Foldseek. *Nat Biotechnol* 2023.
Xu, H.*, et al.* Homology-Based Variant-Effect Predictors Break Down on Cytochrome P450 Pharmacogenes. *Pacific Symposium on Biocomputing* 2027.

Synnotate: Interpretable Genomic-Context-Based Annotation of the Prokaryotic and Viral Dark Proteome

Khoa Hoang*
Department of Bioengineering, Stanford University,
Stanford, CA, USA
E-mail: khoang99@stanford.edu

Swan Van
Department of Biochemistry and Molecular Biophysics, University of California San Diego
San Diego, CA, USA
E-mail: nvan@ucsd.edu

Janice Subroto
Computational and Systems Biology, University of California Los Angeles
Los Angeles, CA, USA
E-mail: jsubroto@g.ucla.edu

Nguyen Nguyen
Faculty of Biology and Biotechnology, VNUHCM University of Science
Ho Chi Minh City, Vietnam
E-mail: 23C64016@student.hcmus.edu.vn

Michael Snyder
Department of Genetics, Stanford University,
Stanford, CA, USA
E-mail: mpsnyder@stanford.edu

*Research reported in this publication was supported by the Human Virome Program, a program of the National Institutes of Health under award number U54AG089334. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

Advances in genome sequencing have dramatically expanded the catalog of microbial and viral proteins, yet millions remain unannotated. Functional annotation relies predominantly on homology-based transfer, which performs poorly for proteins lacking informative sequence or structural similarity. Conserved genomic neighborhoods provide complementary evidence for functional inference by capturing evolutionarily conserved functional relationships among neighboring genes. Existing methods, however, either identify conserved neighborhoods without predicting protein function or rely on black-box machine learning models with limited interpretability. Here, we present Synnotate, an interpretable genome-context framework for microbial dark proteome annotation beyond sequence and structural homology. Synnotate represents genomic neighborhoods as standardized functional annotation tokens and infers the function of a target gene from its neighbors, along with calibrated confidence, exact neighbor-level attribution, and retrieved reference neighborhoods. Applied at scale, Synnotate assigns high-confidence functional predictions to 4.9% of previously unannotated genes in bacteriophage genomes and 2.0% in prokaryotic genomes, extending functional annotation and genome mining beyond the limits of sequence and structural similarity alone.

Keywords: Functional annotation; genome context; dark proteome; microbes; bacteriophages; machine learning; interpretability.

1. Introduction

The microbial and viral dark proteome represents one of the largest unresolved challenges in functional genomics, with an estimated 20–30% of bacterial proteins and nearly 65% of bacteriophage proteins remaining functionally uncharacterized.[1,2] This persistent annotation gap constrains our ability to interpret microbial and viral genomes at scale. Current annotation pipelines rely predominantly on sequence- and structure-based search and clustering for functional annotation,[3,17] with emerging protein language model (pLM)-based approaches improving the detection of remote homologs.[4] Because these strategies rely on previously characterized proteins in reference databases, they perform poorly for lineage-restricted, rapidly evolving, and highly divergent proteins.

Genomic context provides functional information complementary to sequence and structural similarity. As functionally related genes are frequently co-localized within conserved genomic neighborhoods, the genes surrounding an uncharacterized protein can provide evidence for its potential function even in the absence of detectable homologs. Tools such as MultiGeneBlast,[5] cblaster,[6] and antiSMASH[7] identify homologous gene clusters and can support post hoc functional inference from matched loci. However, they are designed primarily for cluster discovery rather than direct protein annotation, and their large-scale homology searches can be costly for genome-wide use. Deep learning methods such as Gaia[8] and Phynteny[9] have incorporated genomic context into scalable protein-function prediction. Because context is combined with pLM representations or other sequence-derived features, however, its independent contribution to individual predictions is difficult to disentangle.

Here, we present Synnotate, a scalable and interpretable framework for prokaryotic and viral protein annotation using genomic context. Synnotate takes as input a fixed-size window of annotated neighboring genes represented as standardized functional annotation tokens and

predicts the function of the central gene using a Transformer-based neural network (Figure 1). Synnotate uses genomic context as its primary predictive input and returns calibrated confidence, neighbor-level attribution, and retrieved reference neighborhoods for each prediction. This enables scalable, evidence-backed annotation of proteins lacking detectable sequence or structural homologs.

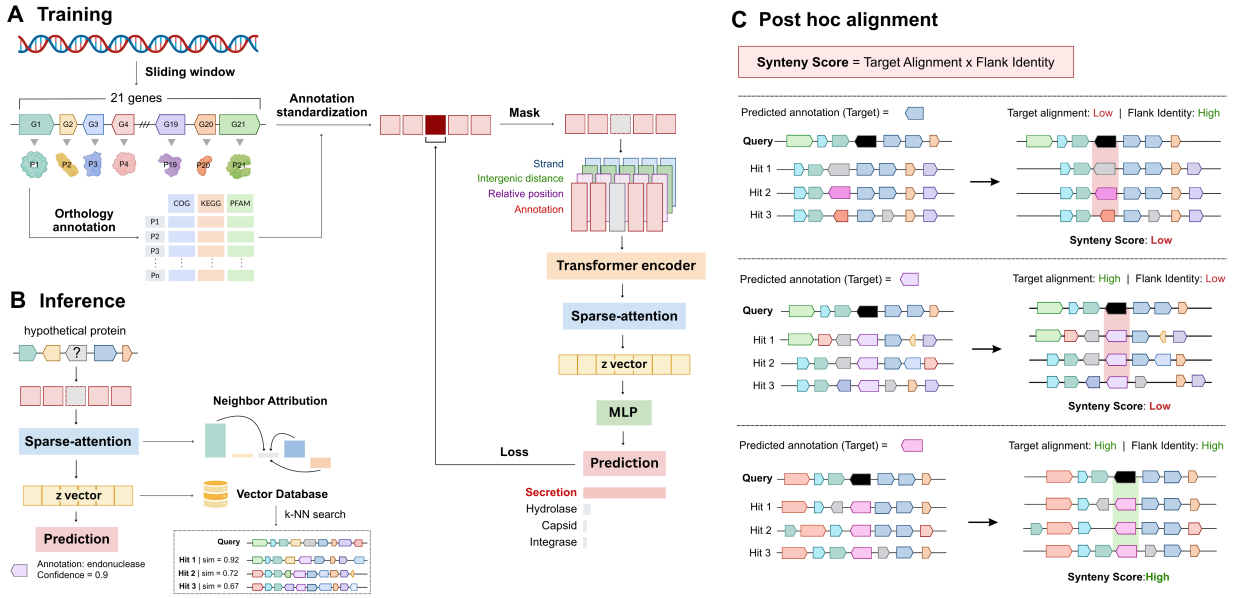


Figure 1. Overview of the Synnotate training, inference, and interpretation workflow. (A) Synnotate training schema. (B) Synnotate inference and interpretation schema. (C) Post hoc annotation alignment for synteny scoring.

2. Methods

2.1. *Training datasets*

2.1.1. *Prokaryotic dataset*

Bacterial and archaeal genomes were obtained from the Genome Taxonomy Database (GTDB)[10] release 232. GTDB species-representative genomes with NCBI RefSeq annotations were retained after quality filtering ($\geq$95% completeness, $\leq$5% contamination, and $\leq$500 contigs), excluding metagenome-assembled genomes, single-amplified genomes, and environmental assemblies. Protein-coding genes were extracted from RefSeq annotations, and GTDB taxonomy was used for phylogenetically informed dataset partitioning. The final dataset consisted of 33,496 genomes spanning 4,900 GTDB genera and approximately 143 million proteins.

2.1.2. *Bacteriophage dataset*

16,479 complete bacteriophage genomes were obtained from the INfrastructure for a PHAge REference Database (INPHARED)[11] (4 January 2026 release). Protein-coding genes were annotated using curated PHROG assignments[12] provided in the database, and phage taxonomy was obtained from the INPHARED metadata table.

2.2. *Standardized annotation vocabulary construction*

A standardized annotation vocabulary was generated by integrating orthology assignments, protein product annotations, and homology-based label propagation. Prokaryotic proteins were annotated with eggNOG-mapper (v2.1.13)[13] using DIAMOND[14] in sensitive mode. For each protein, the KO identifier[15] assigned to its eggNOG-mapper orthologous group was used when available. Otherwise, the group's first annotated Pfam domain was used.[16] Proteins lacking both assignments were retained without an initial orthology-derived group. Phage proteins were assigned to PHROG families.[12] Annotation tokens were defined by group identifiers and matched with the most frequent informative RefSeq product annotation. Genes lacking a group identifier but possessing an informative RefSeq product annotation were then grouped by their product annotation. Remaining unlabeled genes were annotated by homology-based label propagation. Labeled and unlabeled sequences were clustered together in a single MMseqs2 Linclust[17] run (30% amino-acid identity, 80% coverage). For an unlabeled gene, the cluster's majority label was transferred only when at least two labeled members were present and the majority label accounted for at least 80% of the labeled members in the cluster. Otherwise, the gene remained unlabeled and assigned a *DARK* token. Rare annotations (<50 occurrences) were collapsed into an *OTHER* token. Summary statistics of the training datasets and the proportions of *DARK* and *OTHER* tokens are provided in Table S1.

2.3. *Synnotate model architecture and training*

Synnotate was trained as a masked-token prediction model over genomic neighborhoods. For each target gene, a symmetric window of $\pm w$ neighboring genes was extracted from the same contig, yielding a fixed input of $2w{+}1$ positions ($w = 10$ by default). Windows extending beyond contig boundaries were padded with a reserved *CONTIG-EDGE* token. Each position was represented by the sum of 384-dimensional embeddings from four tracks: annotation, coding strand, intergenic distance, and position within the window (0 to $2w$). During training and inference, only the target gene's annotation token was replaced by the reserved *MASK* token; its strand, intergenic distance, and position features were retained. The input sequence was encoded by a two-layer Transformer (embedding dimension 384, four attention heads, feed-forward dimension 512, GELU activation, dropout 0.1). Encoded neighbor representations were aggregated by a target-conditioned sparse-attention pooling layer, in which a query derived from the masked target attended over neighboring genes (excluding the target itself) using sparsemax.[18] The pooled representation was passed to a linear classifier to predict the target functional token. The sparsemax-weighted linear classifier permits exact decomposition of each prediction into per-neighbor contributions (Fig. 1A, B).

Models were trained using the AdamW optimizer, cross-entropy loss, and a batch size of 2,048 for up to 40 epochs. Early stopping (patience = 5) was based on validation top-1 accuracy, with the best-performing checkpoint retained. To mitigate class imbalance, each annotation token contributed at most 3,000 training examples. Model performance was evaluated using a genus-disjoint five-fold cross-validation protocol over 4,900 prokaryotic and 1,397 phage genera. In each fold, one genus-disjoint partition was held out for testing, and the remaining genera were split into disjoint training and validation sets. Each annotated gene was

evaluated once in an outer test fold using a model trained and validated only on other genera. The deployment model used for genome-wide annotation was trained on the full dataset using an analogous genus-disjoint validation set (∼2%) for early stopping and confidence calibration.

2.4. *Vector database construction and retrieval*

To provide reference examples for each prediction, Synnotate indexes annotated genomic neighborhoods in the learned context-embedding space (*z-vector*, Fig. 1B). Specifically, each neighborhood in the training set was represented by the pooled embedding produced by the trained deployment model immediately before the final linear classifier. Embeddings were L2-normalized and indexed using FAISS[19] together with the corresponding genome metadata. For each query gene, its genomic neighborhood was embedded using the same model and used to query the reference vector database to retrieve the k nearest neighborhoods (default $k = 5$) from distinct genomes, excluding the genome of the query itself. Retrieved neighborhoods were used for interpretation and post hoc synteny scoring.

2.5. *Post-hoc annotation-token alignment and synteny score*

Retrieved neighborhoods were aligned to the query neighborhood using Needleman–Wunsch alignment[20] over their annotation token sequences (Fig. 1C). Identical non-special tokens scored +2, while mismatches and gaps scored −1. *MASK* and *CONTIG-EDGE* tokens contributed 0. To account for inverted syntenic blocks, each neighborhood was aligned in both forward and reverse orientations, retaining the higher-scoring alignment. After alignment, flank identity was calculated as the mean fraction of matching flanking tokens, and target alignment as the fraction of retrieved neighborhoods whose aligned central token matched the prediction. The overall synteny score was defined as

$$\text{Synteny score} = \text{Flank identity} \times \text{Target alignment}. \tag{1}$$

2.6. *Structural search and clustering*

Protein structural similarity searches were performed using Foldseek with ProstT5-derived 3Di representations.[21] Proteins were searched against AFDB-SwissProt for prokaryotes and BFVD[22] for bacteriophages (E-value $\leq 10^{-3}$, maximum 10 hits), retaining the lowest-E-value hit and its target annotation. Structural clusters were generated using Foldseek *cluster* (greedy set-cover, 80% coverage).

2.7. *Structure prediction and alignment*

Representative proteins from structural clusters and selected case-study proteins were modeled using localColabFold (ColabFold[23] v1.6.1) with the AlphaFold2-ptm model (three models, three recycles, no structural templates or Amber relaxation). Multiple sequence alignments were generated through the ColabFold public MMseqs2 API using the *mmseqs2_uniref_env* database. Models were ranked by predicted local distance difference test (pLDDT), and the top model was summarized by mean pLDDT and predicted TM-score (pTM). Pairwise structural

comparisons were performed with TM-align.[24] Structural similarity was evaluated using the TM-score normalized to the reference chain length, root mean square deviation (RMSD), aligned length, and sequence identity over the aligned region.

2.8. *Reference phylogenetic trees*

Reference bacterial phylogenies were obtained by pruning the GTDB bac120 reference tree (release 232) to the taxa of interest using DendroPy.[25] Reference phage phylogenies were inferred from whole-proteome similarity. The most complete genome was selected from each INPHARED genus. Mash[26] amino acid sketches (size 1,000) were used to compute all-versus-all distances, from which a neighbor-joining tree was constructed with DendroPy; each tip represents one phage genus.

2.9. *SeqHub annotation*

SeqHub[8] annotation was generated using Protein Context Search APIs, with the target protein and ±2 neighboring genes as input. Searches were run at all four *diversityFilter* settings. The target annotation was assigned by majority vote across retrieved target-protein annotations.

2.10. *cblaster annotation*

cblaster[6] was used to identify homologous gene clusters through its remote search API. For each query, a 21-gene neighborhood centered on the target protein was searched against the NCBI non-redundant database with `-hs=5`, and annotations of matched target proteins were retrieved through the NCBI Entrez API for comparison with Synnotate predictions.

2.11. *Phynteny benchmark*

Synnotate was benchmarked against three Phynteny variants, each hiding a target gene's annotation while keeping all other annotated genes in the genome visible: (1) the released model (`phynteny_transformer` v0.1.3) applied to the same held-out test folds; (2) a per-fold model retrained on the same genus-held-out training genomes using the published recipe for 100 epochs; and (3) a context-only model directly comparable to Synnotate, trained and evaluated as in (2) but with the `--zero_idx` option, which zeroes the masked target's embedding so that no target-sequence information reaches the model. Both retrained variants were calibrated on their fold's genus-disjoint validation fold. As a sequence-only control, a small per-fold multilayer perceptron was trained on mean `esm2_t33_650M_UR50D` embeddings,[35] the same representation Phynteny uses, and calibrated on the same validation folds. To compare in Phynteny's nine-category label space, each Synnotate annotation was mapped to its PHROG v4 category, annotation probabilities were summed within each category, and the highest-scoring category was taken as the prediction.

3. Results

3.1. *Synnotate enables confident and interpretable protein function prediction from genomic context*

To assess whether genomic context alone contains sufficient information for protein function prediction, we masked annotated genes and inferred their functional tokens solely from the surrounding genomic neighborhood. Performance was evaluated by genus-disjoint five-fold cross-validation over 4,900 prokaryotic and 1,397 bacteriophage genera and compared with two context-only baselines: (1) a *Logistic Regression* Baseline that omits gene order, strand, and intergenic distance, and (2) a *Neural Baseline* that replaces the Transformer encoder and sparse-attention pooling with mean pooling. Synnotate consistently outperformed both baselines across all levels of neighborhood annotation, achieving top-1 accuracies of 50.7% in prokaryotes and 83.2% in bacteriophages (Fig. 2A). Prediction accuracy increased with neighborhood annotation coverage and remained higher in bacteriophages within matched coverage bins, consistent with bacteriophages' compact, modular genome organization. We benchmarked Synnotate against Phynteny[9] on predicting the nine PHROG functional categories in the same held-out test sets (Fig. 2D, Section 2.11). The sequence-only ESM-2 model achieved consistently high top-1 accuracy at all levels of neighborhood annotation. The released and retrained Phynteny models performed comparably to each other and below ESM-2, suggesting that for Phynteny, genomic context adds noise when there is strong target sequence information. Because Synnotate uses only genomic context, we also evaluated Phynteny with the target's sequence embedding zeroed. This context-only variant performed substantially worse than Synnotate ($\sim$27 pp), likely due to differences in tokenization and Synnotate's train–test consistency. Balanced accuracy and macro-F1 are reported in Fig. S3.

We next examined whether Synnotate's confidence scores reflect prediction reliability. Higher softmax probabilities generally corresponded to greater accuracy, but the raw scores were imperfectly calibrated, so we applied isotonic regression on the validation folds to map them to observed accuracy (Fig. 2B). A few high-confidence errors persisted, reflecting a failure mode we term *context leakage* (Fig. S1): a gene at a shifted or inserted position receives the function typically found at a nearby conserved position. To filter these cases, we added a synteny score based on annotation-token alignment between the query and retrieved reference neighborhoods (Fig. 1C, Section 2.5). Together with calibrated confidence, this score defined trusted prediction regions that achieved the target accuracy on held-out test genes (Fig. 2C) and were applied across all genomes for dark-proteome annotation.

Finally, we examined the genomic features underlying Synnotate's predictions. Accuracy increased with neighborhood size up to $\sim \pm 5$ genes before plateauing (Fig. S2), suggesting that most functional information lies within compact neighborhoods, consistent with typical operon and phage module sizes.[27,28] Feature ablation showed that neighbor annotations contributed most of the predictive signal, while genomic organization (relative position, coding strand, and intergenic distance) provided smaller, complementary gains (Fig. 2E). Organization alone achieved 34.2% accuracy on bacteriophages, indicating that gene arrangement itself carries substantial functional information. Neighbor-level attribution showed that Synnotate relies mainly on immediately adjacent genes, with attribution decreasing monotonically with

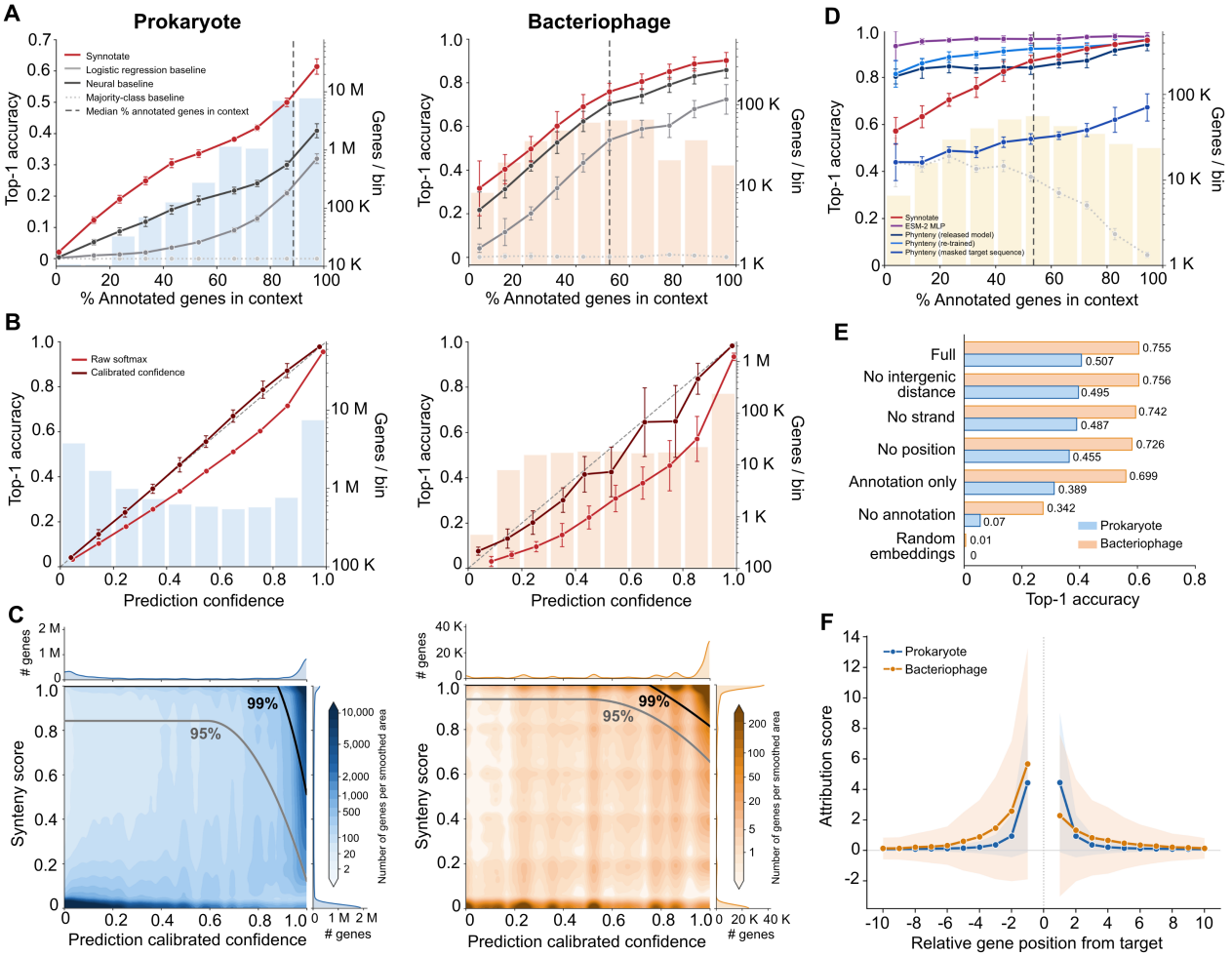


Figure 2. Evaluation of Synnotate on held-out annotated genes. (A) Top-1 accuracy on prokaryote and bacteriophage datasets, stratified by the percentage of annotated genes in the 21-gene neighborhood. (B) Observed top-1 accuracy versus raw softmax probability and isotonic-calibrated confidence. (C) Calibrated confidence versus synteny score for test genes pooled across five folds; grey and black lines mark thresholds for 95% and 99% expected accuracy. (D) Top-1 accuracy on bacteriophages of Synnotate, an ESM-2 embedding MLP, and three Phynteny variants (released, retrained, and retrained with the target sequence masked), stratified as in (A). (E) Feature ablation showing the effect of removing each feature on performance. (F) Neighbor attribution scores across five test folds. Error bars (A, B, D) and shading (F) denote ±1 SD across folds.

distance (Fig. 2F), and in bacteriophages was biased toward upstream neighbors, consistent with directional operonic organization.

3.2. *Confident annotation beyond sequence- and structure-based search: a Desulfovibrio piger case study*

We next tested whether Synnotate could annotate proteins left unresolved by sequence- and structure-based pipelines using *Desulfovibrio piger*. Of 2,430 protein-coding genes, 2,105 received functional annotations from RefSeq and eggNOG-mapper (Fig. 3A), while Foldseek

structural searches provided additional annotations for 45 proteins. Among the remaining 280 proteins lacking sequence- or structure-based annotations, Synnotate confidently resolved five unannotated proteins (Fig. 3B).

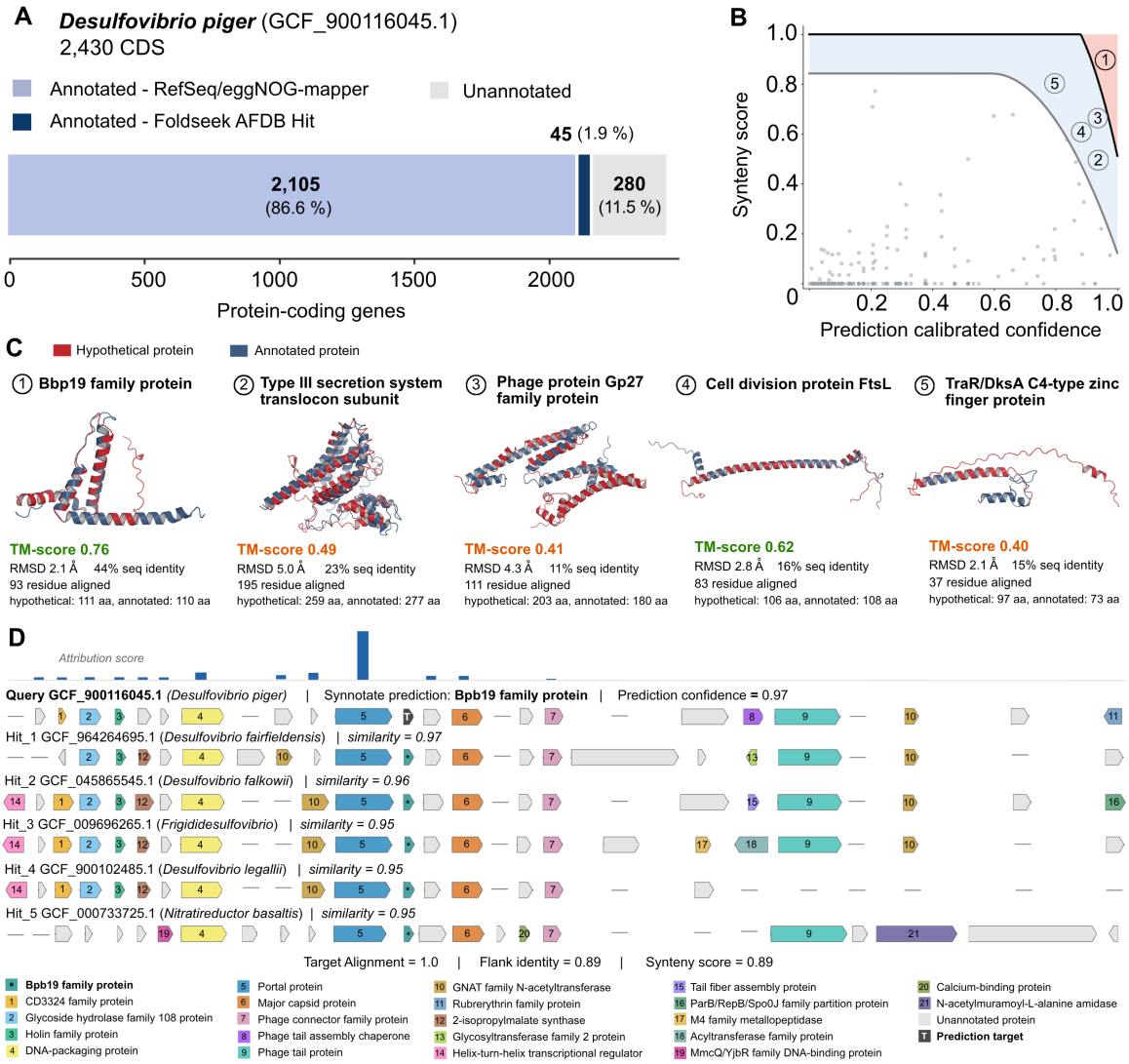


Figure 3. Case study of Synnotate annotation of previously unannotated genes in *Desulfovibrio piger*. (A) Breakdown of protein-coding genes by annotation tier: (i) genes annotated by RefSeq/eggNOG-mapper; (ii) genes unannotated in (i) but annotated via Foldseek against AFDB-SwissProt; (iii) genes remaining unannotated after (i) and (ii). (B) Synnotate prediction confidence and synteny scores for genes in tier (iii); five genes exceeding the 95% accuracy threshold are highlighted. (C) Structural alignments of the five Synnotate-predicted proteins to annotated proteins with the same predicted label from the nearest retrieved reference neighborhood. (D) Example interpretation output for the predicted Bbp19 gene, showing neighbor attribution scores, nearest-neighbor hits from the vector database, and post hoc annotation-token alignment.

The five predictions were (1) Bbp19 family protein, (2) type III secretion system translocon subunit SctB, (3) phage Gp27-family protein, (4) cell division protein FtsL, and (5)

TraR/DksA C4-type zinc-finger protein. For each prediction, the annotated protein from the nearest retrieved neighborhood carrying the predicted token was used as the structural reference. Two predictions (1, 4) showed strong structural agreement (TM-scores 0.76 and 0.62), whereas the remaining predictions showed conserved structural topologies and genomic neighborhoods (Fig. S4) despite low TM-scores (0.40–0.49) and sequence identity (5–23%), indicating that genomic context can reveal functional annotations for divergent proteins. Beyond prediction, Synnotate provides inspectable evidence through neighbor-level attribution and retrieval of conserved genomic neighborhoods. For example, annotation of the Bbp19 family protein was primarily supported by its immediately upstream portal protein and downstream major capsid protein, while retrieved neighborhoods independently confirmed conservation of the surrounding gene organization (Fig. 3D). We next compared Synnotate with existing context-aware annotation tools. SeqHub Protein Context Search[8] produced matching predictions for two of the five genes (SctB and FtsL), assigned a different function to the TraR/DksA C4-type zinc-finger protein (lipoprotein/spanin subunit), and returned uncharacterized predictions for the remaining two genes. cblaster[6] correctly recovered only SctB and failed to resolve the remaining targets (Sections 2.9 and 2.10). These results demonstrate that Synnotate can infer protein function from transferable signals encoded by neighboring genes within conserved genomic neighborhoods, enabling interpretable annotation of proteins unresolved by a variety of homology and deep learning methods.

3.3. *Synnotate recovers divergent prokaryotic dark proteins*

We then applied Synnotate to 17.6 million dark genes from 33,496 prokaryotic genomes. Using the trusted-region thresholds, Synnotate confidently annotated 108,991 genes (0.6%) at the 99% threshold and 359,060 genes (2.0%) at the 95% threshold, 17% of which returned a named hit via Foldseek search. Synnotate predictions were broadly distributed across the prokaryotic phylogeny, with a median of 8 genes annotated per genome (Fig. 4A, Fig. S5). Among the most frequently recovered functions was the inner-membrane protein YgfX, with 1,463 previously unannotated proteins expanding the known family by 30% across 475 genera, including 307 without previously annotated YgfX (Fig. 4B, Fig. 4E). Predicted YgfX proteins exhibited length distributions nearly identical to annotated YgfX proteins (Fig. 4C). A Foldseek structural similarity search against AFDB-SwissProt identified structural homologs for only 14.7% of the predicted proteins (Fig. 4D). However, when predicted and annotated proteins were structurally clustered together (Section 2.6), 97.7% of the predicted YgfX proteins clustered with previously annotated YgfX proteins, indicating that the apparent lack of structural homologs primarily reflected incomplete reference databases rather than novel protein folds. We next examined a subset of structurally unresolved YgfX proteins: a representative 96-amino-acid protein from *Aliivibrio* (confidence = 0.99, synteny support = 1.0) that lacked database matches and structural clustering, yet retained structural similarity to a context-retrieved YgfX reference (TM-score = 0.41) within a conserved genomic neighborhood (Fig. 4F–H). Together, these findings suggest that Synnotate can recover highly divergent members of conserved protein families that are missed by conventional homology-based annotation.

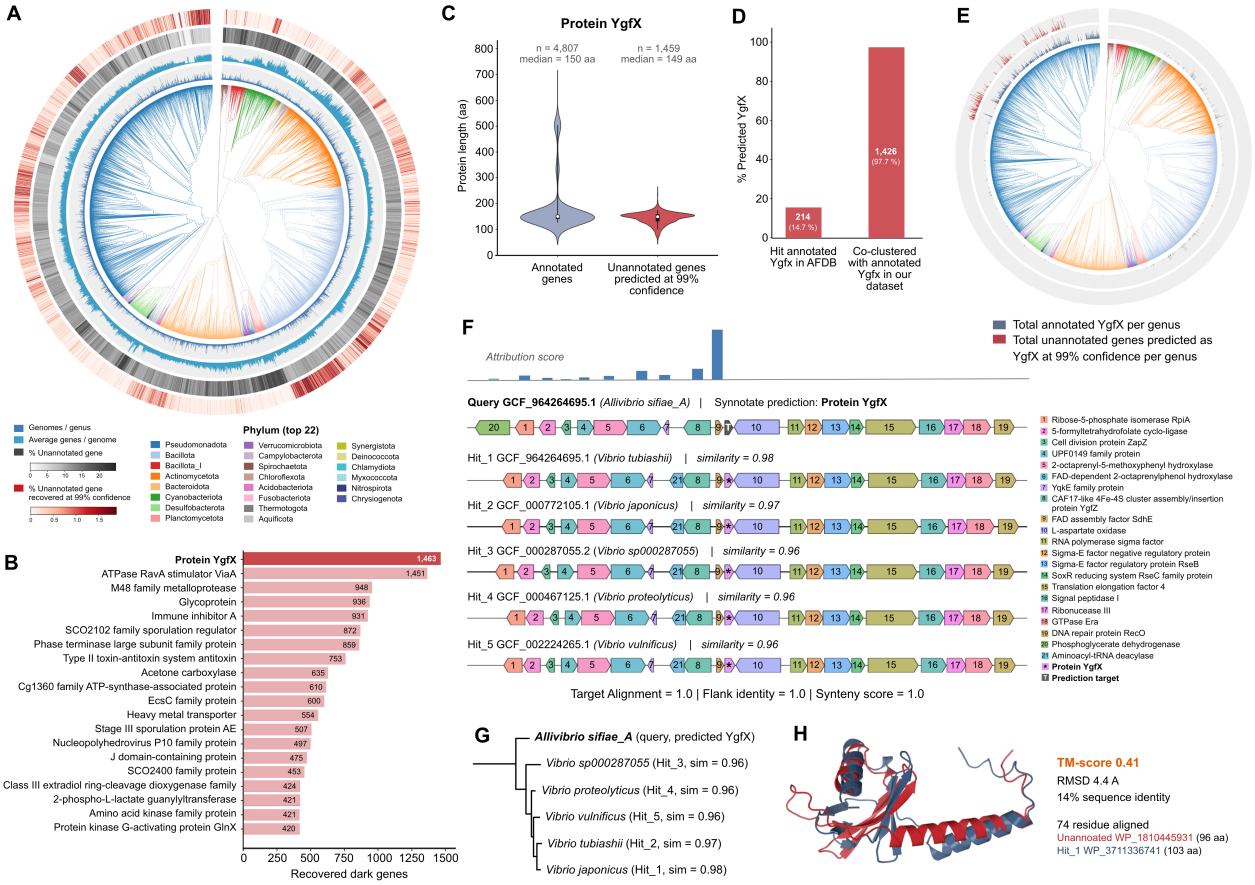


Figure 4. Synnotate annotation of the prokaryotic dark proteome. (A) Prokaryotic phylogeny showing dark-proteome genes and their Synnotate annotations, aggregated by genus. (B) Top 20 predicted annotations at 99% confidence. (C) Length distributions of annotated YgfX proteins and unannotated proteins predicted as YgfX. (D) Foldseek search of predicted YgfX proteins against AFDB and their clustering with annotated proteins from the prokaryotic dataset. (E) Distribution of annotated and predicted YgfX across the prokaryotic phylogeny. (F) Interpretation output (neighbor attributions, retrieved reference neighborhoods, and post hoc annotation-token alignment) for a predicted YgfX that neither co-clusters with nor matches annotated YgfX in AFDB. (G) Phylogeny of bacteria containing "dark" YgfX and its k nearest-neighbor hits from (F). (H) Structural alignment of a dark/predicted YgfX with the annotated YgfX from its nearest retrieved neighborhood.

3.4. *Synnotate expands annotation of the bacteriophage dark proteome*

Applying Synnotate to 873,701 unannotated genes from 16,479 bacteriophage genomes, we confidently annotated 3.7% at the 99% threshold and 4.9% of dark proteins at the 95% threshold, 22% of which returned a named hit via Foldseek search. Homing endonuclease-like proteins represented the most frequent high-confidence prediction (Fig. 5A), with 1,832 additional candidates expanding the annotated repertoire by 62% relative to 2,967 known members. Interestingly, these predicted proteins were substantially shorter than previously annotated homing endonucleases (median length 77 versus 233 amino acids) (Fig. 5B). Direct structural search and clustering provided little evidence for these candidates: Foldseek search against BFVD[22] identified informative structural homologs for only 0.7% of predicted proteins, and

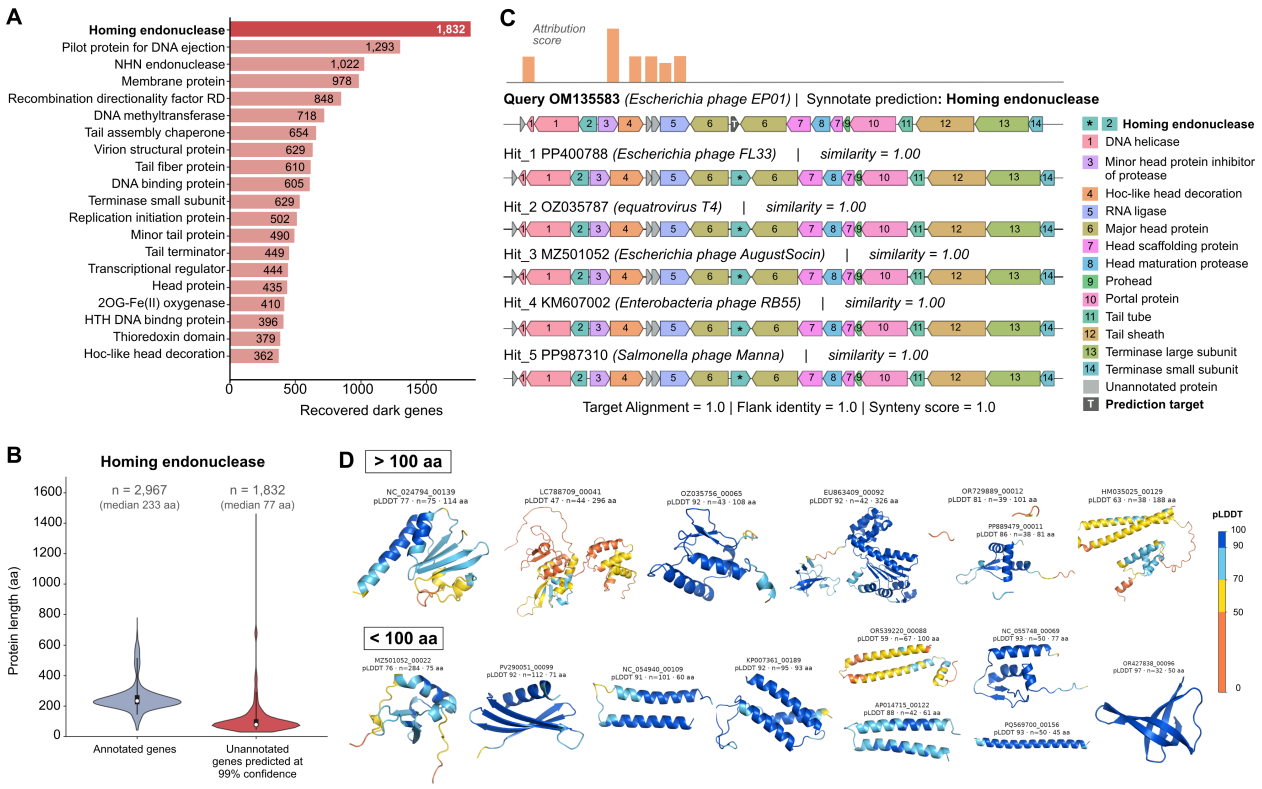


Figure 5. Synnotate annotation of the viral dark proteome. (A) Top 20 Synnotate-predicted annotations at 99% confidence. (B) Length distributions of annotated homing endonucleases and proteins predicted as homing endonucleases. (C) Neighbor attribution and synteny alignment between the query genomic context and retrieved examples. (D) Representative proteins from the top 16 clusters of predicted homing endonucleases.

only 0.6% clustered with previously annotated homing endonucleases (Section 2.6). Yet these proteins retained the canonical genomic position between capsid protein genes (Fig. 5C) and broadly overlapped the phylogenetic distribution of known homing endonucleases (Fig. S6B). When clustered independently, approximately twelve structural clusters accounted for half of all predicted homing endonucleases (Fig. S7, Fig. 5D). The predominance of short proteins was notable as GIY-YIG homing endonucleases contain compact catalytic domains[29,30] that can be separated from larger accessory regions, suggesting that some predictions may represent highly divergent, minimized forms, or highly degraded remnants of mobile elements.[31] Together, these findings suggest that conserved genomic position can persist even when the central proteins show little detectable sequence or structural similarity, allowing context-based models to prioritize otherwise overlooked viral proteins.

4. Discussion

Protein functional annotation has been largely driven by sequence and structural similarity, yet these relationships become progressively less informative for rapidly evolving and highly divergent proteins. Here, we show that genomic context provides functional information com-

plementary to sequence and structure. Synnotate translates this signal into a predictive and interpretable framework for large-scale protein function annotation beyond homology.

The proteins recovered by Synnotate illustrate three ways in which genomic context can extend annotation. First, some proteins appear to be known family members that are missing from current reference databases because sequence and structural diversity are incompletely sampled. In *Desulfovibrio piger*, the predicted Bbp19 and FtsL proteins lacked informative database hits but showed strong structural agreement with context-retrieved references (TM-scores 0.76 and 0.62). Likewise, 97.7% of predicted YgfX proteins clustered with annotated YgfX proteins despite limited direct database support. Second, some proteins may be more deeply diverged members of known families. This pattern includes the three lower-TM-score *D. piger* predictions—SctB, a Gp27-family protein, and a TraR/DksA C4-type zinc-finger protein—as well as a small subset of predicted YgfX proteins. These proteins retained genomic-context support and weak structural similarity to annotated references (TM-scores 0.41–0.49). Third, some predictions may represent poorly characterized or previously unrecognized evolutionary variants. The phage homing endonuclease-like proteins showed little detectable sequence or structural similarity to known examples but repeatedly occurred at the same conserved genomic position and within overlapping phage lineages. They may represent highly divergent variants, compact forms, decaying mobile-element remnants, or previously unrecognized branches of related proteins. Together, these examples show that genomic context can recover both overlooked members of known families and more deeply divergent proteins that fall outside the reach of current sequence- and structure-based annotation.

Several challenges remain. First, Synnotate depends on informative neighboring annotations and conserved gene order, so performance may decline in sparsely annotated or highly rearranged regions. Its functional vocabulary also inherits gaps and biases from existing annotation methods. In addition, confidence thresholds were determined based on annotated genes and applied to unannotated ones. Finally, although our datasets cover broad bacterial and viral diversity, they represent only a small fraction of known microbial genomes. Expanding training to larger and more diverse genome collections and metagenomes[32] may improve performance and extend annotation to less explored protein families.

More broadly, Synnotate can support genome mining beyond protein annotation. Its neighborhood embeddings and vector database make it possible to retrieve genes that occur in similar genomic contexts, even when their annotations or protein sequences differ. By combining context similarity with the diversity of proteins found at the central position, these representations could help identify mobile-element insertion sites, cassette-exchange hotspots, and regions of rapid gene turnover. In this way, genomic context could become not only an annotation signal but also a searchable representation of genome organization, revealing conserved loci and exchange hotspots that sequence-based comparisons fail to connect.

5. Code and Data Availability

Code and data: `https://github.com/khoa-yelo/Synnotate`. Supplementary materials: `https://github.com/khoa-yelo/Synnotate/blob/main/paper/supp.pdf`.

References

1. P. Escudeiro, C. S. Henry and R. P. M. Dias, Functional characterization of prokaryotic dark matter: the road so far and what lies ahead, *Curr. Res. Microb. Sci.* **3**, p. 100159 (2022), https://doi.org/10.1016/j.crmicr.2022.100159.
2. S. Nayfach *et al.*, Metagenomic compendium of 189,680 DNA viruses from the human gut microbiome, *Nat. Microbiol.* **6**, 960 (July 2021), https://doi.org/10.1038/s41564-021-00928-6.
3. M. van Kempen *et al.*, Fast and accurate protein structure search with Foldseek, *Nat. Biotechnol.* **42**, 243 (February 2024), https://doi.org/10.1038/s41587-023-01773-0.
4. Z. N. Flamholz, S. J. Biller and L. Kelly, Large language models improve annotation of prokaryotic viral proteins, *Nat. Microbiol.* **9**, 537 (February 2024), https://doi.org/10.1038/s41564-023-01584-8.
5. M. H. Medema, E. Takano and R. Breitling, Detecting sequence homology at the gene cluster level with MultiGeneBlast, *Mol. Biol. Evol.* **30**, 1218 (May 2013), https://doi.org/10.1093/molbev/mst025.
6. C. L. M. Gilchrist *et al.*, cblaster: a remote search tool for rapid identification and visualization of homologous gene clusters, *Bioinform. Adv.* **1**, p. vbab016 (January 2021), https://doi.org/10.1093/bioadv/vbab016.
7. K. Blin *et al.*, antiSMASH 8.0: extended gene cluster detection capabilities and analyses of chemistry, enzymology, and regulation, *Nucleic Acids Res.* **53**, W32 (July 2025), https://doi.org/10.1093/nar/gkaf334.
8. N. Jha *et al.*, Gaia: an AI-enabled genomic context-aware platform for protein sequence annotation, *Sci. Adv.* **11**, p. eadv5109 (June 2025), https://doi.org/10.1126/sciadv.adv5109.
9. S. R. Grigson *et al.*, Synteny-aware functional annotation of bacteriophage genomes with Phynteny bioRxiv preprint (July, 2025), https://doi.org/10.1101/2025.07.28.667340.
10. D. H. Parks, M. Chuvochina, C. Rinke, A. J. Mussig, P.-A. Chaumeil and P. Hugenholtz, GTDB: an ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy, *Nucleic Acids Res.* **50**, D785 (September 2021), https://doi.org/10.1093/nar/gkab776.
11. R. Cook *et al.*, INfrastructure for a PHAge REference database: identification of large-scale biases in the current collection of cultured phage genomes, *Phage (New Rochelle)* **2**, 214 (December 2021), https://doi.org/10.1089/phage.2021.0007.
12. P. Terzian *et al.*, PHROG: families of prokaryotic virus proteins clustered using remote homology, *NAR Genom. Bioinform.* **3**, p. lqab067 (August 2021), https://doi.org/10.1093/nargab/lqab067.
13. C. P. Cantalapiedra, A. Hernández-Plaza, I. Letunic, P. Bork and J. Huerta-Cepas, eggNOG-mapper v2: functional annotation, orthology assignments, and domain prediction at the metagenomic scale, *Mol. Biol. Evol.* **38**, 5825 (October 2021), https://doi.org/10.1093/molbev/msab293.
14. B. Buchfink, K. Reuter and H.-G. Drost, Sensitive protein alignments at tree-of-life scale using DIAMOND, *Nat. Methods* **18**, 366 (April 2021), https://doi.org/10.1038/s41592-021-01101-x.
15. X. Mao, T. Cai, J. G. Olyarchuk and L. Wei, Automated genome annotation and pathway identification using the KEGG orthology (KO) as a controlled vocabulary, *Bioinformatics* **21**, 3787 (October 2005), https://doi.org/10.1093/bioinformatics/bti430.
16. E. L. Sonnhammer, S. R. Eddy and R. Durbin, Pfam: a comprehensive database of protein domain families based on seed alignments, *Proteins* **28**, 405 (July 1997), https://doi.org/10.1002/(sici)1097-0134(199707)28:3<405::aid-prot10>3.0.co;2-l.
17. M. Steinegger and J. Söding, MMseqs2 enables sensitive protein sequence searching for the

analysis of massive data sets, *Nat. Biotechnol.* **35**, 1026 (November 2017), https://doi.org/10.1038/nbt.3988.
18. A. F. T. Martins and R. F. Astudillo, From softmax to sparsemax: a sparse model of attention and multi-label classification arXiv:1602.02068 (February, 2016), https://doi.org/10.48550/arXiv.1602.02068.
19. M. Douze *et al.*, The Faiss library arXiv:2401.08281 (October, 2025), https://doi.org/10.48550/arXiv.2401.08281.
20. S. B. Needleman and C. D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, *J. Mol. Biol.* **48**, 443 (March 1970), https://doi.org/10.1016/0022-2836(70)90057-4.
21. M. Heinzinger, K. Weissenow, J. G. Sanchez, A. Henkel, M. Steinegger and B. Rost, ProstT5: bilingual language model for protein sequence and structure bioRxiv preprint (July, 2023), https://doi.org/10.1101/2023.07.23.550085.
22. R. S. Kim, E. L. Karin, M. Mirdita, R. Chikhi and M. Steinegger, BFVD—a large repository of predicted viral protein structures, *Nucleic Acids Res.* **53**, D340 (January 2025), https://doi.org/10.1093/nar/gkae1119.
23. M. Mirdita, K. Schütze, Y. Moriwaki, L. Heo, S. Ovchinnikov and M. Steinegger, ColabFold: making protein folding accessible to all, *Nat. Methods* **19**, 679 (June 2022), https://doi.org/10.1038/s41592-022-01488-1.
24. Y. Zhang and J. Skolnick, TM-align: a protein structure alignment algorithm based on the TM-score, *Nucleic Acids Res.* **33**, 2302 (2005), https://doi.org/10.1093/nar/gki524.
25. J. Sukumaran and M. T. Holder, DendroPy: a Python library for phylogenetic computing, *Bioinformatics* **26**, 1569 (June 2010), https://doi.org/10.1093/bioinformatics/btq228.
26. B. D. Ondov *et al.*, Mash: fast genome and metagenome distance estimation using MinHash, *Genome Biol.* **17**, p. 132 (June 2016), https://doi.org/10.1186/s13059-016-0997-x.
27. T. Conway *et al.*, Unprecedented high-resolution view of bacterial operon architecture revealed by RNA sequencing, *mBio* **5**, e01442 (July 2014), https://doi.org/10.1128/mBio.01442-14.
28. E. S. Miller, E. Kutter, G. Mosig, F. Arisaka, T. Kunisawa and W. Rüger, Bacteriophage T4 genome, *Microbiol. Mol. Biol. Rev.* **67**, 86 (March 2003), https://doi.org/10.1128/MMBR.67.1.86-156.2003.
29. J. C. Kowalski *et al.*, Configuration of the catalytic GIY-YIG domain of intron endonuclease I-TevI: coincidence of computational and molecular findings, *Nucleic Acids Res.* **27**, 2115 (May 1999), https://doi.org/10.1093/nar/27.10.2115.
30. K. W. Loedige *et al.*, Modular determinants of cleavage preference in GIY-YIG nucleases revealed by block-based DNA shuffling and directed evolution, *Nucleic Acids Res.* **54**, p. gkaf1430 (January 2026), https://doi.org/10.1093/nar/gkaf1430.
31. A. Barzel, U. Obolski, J. P. Gogarten, M. Kupiec and L. Hadany, Home and away—the evolutionary dynamics of homing endonucleases, *BMC Evol. Biol.* **11**, p. 324 (November 2011), https://doi.org/10.1186/1471-2148-11-324.
32. G. Ta *et al.*, MGnify genomes: a resource for biome-specific microbial genome catalogues, *J. Mol. Biol.* **435**, p. 168016 (July 2023), https://doi.org/10.1016/j.jmb.2023.168016.
33. K. Katoh, K. Misawa, K. Kuma and T. Miyata, MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform, *Nucleic Acids Res.* **30**, 3059 (July 2002), https://doi.org/10.1093/nar/gkf436.
34. Y. Shimoyama, pyMSAviz: MSA visualization python package for sequence analysis https://moshi4.github.io/pyMSAviz/, (2026), Accessed: Jul. 28, 2026.
35. Z. Lin, H. Akin, R. Rao, B. Hie *et al.*, Evolutionary-scale prediction of atomic-level protein structure with a language model, *Science* **379**, 1123 (March 2023), https://doi.org/10.1126/science.ade2574.

Predicting Microbial Gene Fitness with a Biological Knowledge Graph

Winston E. Anthony[1*], Sam Donald[1*], Adam Deutschbauer[2], Sumit Purohit[1], Kaizad Patel[1] and Robert Egbert[1†]

[1]*Integrated Discovery Sciences Directorate, Pacific Northwest National Laboratory, Richland, WA, USA.* [2]*Environmental Genomics and Systems Biology Division, Lawrence Berkeley National Laboratory, Berkeley, CA, USA*

[†]*E-mail: robert.egbert@pnnl.gov*

[*]*These authors contributed equally to this work.*

Predicting the condition-dependent fitness effects of gene inactivation remains a central challenge in microbial phenomics. We introduce GIMME-KG (**G**raph **I**nference for **M**icrobial Dark **M**atter **E**xploration with **K**nowledge **G**raphs), a knowledge graph (KG) integrating 7.94 million RB-TnSeq fitness measurements spanning 55,421 genes, 1,770 experiments, and 10 Pseudomonas strains with experimental metadata, metabolic annotations, and pretrained text and protein-sequence representations. We train GFM1 (GIMME-KG Fitness Model 1), a heterogeneous graph attention network (GAT) to predict gene-inactivation fitness effects under three held-out evaluations: experiments, genes, and ortholog clusters. Message passing over a subset of the knowledge graph increases performance: In the held-out-gene setting, message passing reduced mean squared error (MSE) by 10.1% and increased Pearson correlation by 12.0% relative to a multi-layer perceptron (MLP) using the same multimodal node-encoder design and decoder template. Regression gains were larger among well-connected genes and partly reflected their enrichment for strong fitness defects. On held-out genes, GFM1 had higher precision than the MLP, but lower recall when predicting strong fitness effects. It also outperformed a genome-scale metabolic model in a matched *Pseudomonas putida* benchmark. Together, these results indicate that message passing can leverage knowledge-graph structure to improve gene-fitness prediction when informative relationships to training genes are available. GIMME-KG and GFM1 represent a step forward in predictive inference of microbial metabolism and a new tool for *in silico* phenotype prediction. Appendix is available here: `https://github.com/PNNL-Predictive-Phenomics/GIMME-KG-manuscript`

Keywords: Knowledge graph; microbial fitness; graph neural network; fitness prediction; Pseudomonas.

1. Introduction

The fitness effect of a genetic perturbation depends on the environment in which it is measured. Characterizing these genotype–environment interactions is important for understanding microbial physiology and predicting phenotypes under unmeasured conditions. Predicting these interactions could reduce the experimental burden of mapping genotype–phenotype fitness landscapes and enable *in silico* inference for unmeasured genes and conditions.[1,2] High

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

1 INTRODUCTION

throughput genome-wide fitness assays provide a powerful method for rapid experimental assessment of the function and metabolic plasticity of microbes. One such technique, randomly barcoded transposon sequencing (RB-TnSeq), uses barcoded transposon insertions to generate pooled mutant libraries containing disruptions across much of the genome.[3] RB-TnSeq assays have been used to identify conditionally important genes, generate hypotheses about gene function, and associate poorly characterized genes with biological pathways through correlated fitness profiles.[4] The Fitness Browser provides a large collection of RB-TnSeq measurements from 9,532 experiments spanning nutrient and stress conditions across diverse bacterial lineages.[5] These measurements provide a foundation for training models that predict fitness effects for gene–condition combinations not observed during training.

The Gram-negative genus *Pseudomonas* includes environmentally, industrially, and clinically important species with diverse metabolic capabilities.[6,7] Despite extensive experimental and computational characterization, a substantial fraction of genes in *Pseudomonas putida* remain without characterized functions.[8] The combination of large fitness datasets, established metabolic models, and persistent annotation gaps makes *P. putida* and related pseudomonads useful systems for evaluating predictive models of genotype–environment interactions.

Conventional computational annotation commonly relies on sequence similarity to characterized homologs.[9] Profile hidden Markov models extend this approach by capturing position-specific patterns of conservation and variation within protein families.[10] Although these methods are valuable for functional annotation, functional similarity alone does not determine the fitness effect of gene disruption under a particular environmental condition. Many poorly characterized proteins, sometimes described as "protein dark matter," also have little detectable similarity to experimentally characterized proteins.[11] Genome-scale metabolic models (GSMMs) provide a mechanistic approach by representing gene–reaction relationships and using constraint-based optimization to simulate metabolic flux and growth.[12,13] GSMMs can simulate the effects of single-gene disruptions, but their predictions are restricted to genes and processes represented in the metabolic network.[14]

Protein language models (PLMs) for protein sequences are promising candidates to address the "dark matter problem".[15] PLMs encode protein sequences as dense representations that capture statistical patterns associated with structural and functional properties. These representations can be combined with fitness measurements, experimental metadata, and metabolic annotations to predict condition-specific phenotypes.[16] Knowledge graphs (KGs) provide a structured framework for integrating such heterogeneous information by representing biological entities and their relationships under an explicit schema.[17,18] Graph neural networks (GNNs) can then aggregate information across relationships such as orthology, pathway membership, gene–reaction associations, and shared experimental context.[19–21] Message passing could therefore allow a predictive model to use relational information integrated from resources such as KEGG,[22] MetaCyc,[23] and Swiss-Prot.[24] Whether this graph structure improves fitness prediction beyond the underlying sequence, text, and metadata features, however, remains an empirical question.

Here, we introduce GIMME-KG (**G**raph **I**nference for **M**icrobial Dark **M**atter **E**xploration with **K**nowledge **G**raphs), a heterogeneous biological knowledge graph integrating RB-TnSeq

fitness measurements with gene, experiment, metabolic, text-derived, and protein-sequence information. We train GFM1 (GIMME-KG Fitness Model 1), a heterogeneous graph neural network for predicting condition-specific gene fitness. We compare graph-based models with a multimodal multilayer perceptron (MLP) that uses the same node-encoder design and decoder template but does not perform message passing. Models are evaluated under three held-out settings: experiments, genes, and family-style gene clustering, to test generalization to unobserved gene–experiment combinations. We examine how the GAT–MLP performance difference varies with network connectivity, annotation coverage, gene-family conservation, conditional essentiality, and Enzyme Commission class, and compare GFM1 with a GSMM in a matched *P. putida* benchmark. Message passing improves prediction for held-out experiments and genes, but its advantage is not significant during the cluster-aware split. Together, these analyses demonstrate biological knowledge-graph structure provides predictive information beyond multi-modal node features and establishes a framework for graph-based microbial fitness prediction.

2. Related Work

Prior machine-learning approaches to microbial gene function and fitness have primarily used either protein-sequence representations or experimental phenotype data. PLMs such as ESM-3 encode amino acid sequences as dense representations, and recent work has combined PLM embeddings with GNNs to predict gene essentiality.[25,26] Sequence-only models, however, do not explicitly represent the environmental conditions under which fitness is measured. Conversely, phenotype-based methods capture relationships within observed assay data. Borchert et al., for example, identified 84 co-fitness modules from a 179-condition *P. putida* fitness matrix using independent component analysis,[27] but such matrix-based approaches do not directly generalize to genes, strains, or conditions absent from the observed data.

Knowledge graphs can integrate sequence-derived, experimental, and functional information. Heterogeneous GNNs have been applied to biomedical knowledge graphs for drug–target, drug–drug interaction, and essentiality prediction tasks;[28–31] FuseLinker additionally combines frozen language-model text embeddings with GNN-based link prediction.[32] Our framework instead targets condition-specific microbial gene fitness by integrating text-derived annotations, protein-sequence representations, experimental context, and heterogeneous graph structure.

3. Methodology

3.1. *Datasets*

We obtained 7,937,507 gene–experiment fitness measurements spanning 55,421 genes, 1,770 experiments, and 10 *Pseudomonas* strains from the Fitness Browser including growth media composition and experimental metadata such as pH and temperature. Gene-level fitness scores are calculated from changes in relative mutant-barcode abundance and reported on a $\log_2$ scale.[5] We define a conditional fitness defect using the "extreme fitness value" of ≤ -2.[27]

The Fitness Browser gene set is integrated with four external reference databases when membership is known: SEED,[33] MetaCyc,[23] KEGG,[22] and UniProt annotations, including Enzyme Commission (EC) assignments. We integrated the full EC hierarchy using the EC

RDF,[34] and added MetaCyc reactions and compounds. The released KG contains 141,947 nodes across 19 node types, 884,487 structural edges across 27 relationship types, and 7,937,507 `FITNESS` supervision edges. Model node types and features are documented in Appendix Tables 2-3, together with the architecture sweep and feature-ablation results (Appendix C and D). Figure 1 shows the schema of the released KG, including the `FITNESS` supervision edges. The split-specific training subgraph is derived from this inventory for each evaluation protocol and excludes `FITNESS` edges from structural message passing. No text or annotations embedded in the training data mention essentiality or fitness outcomes.

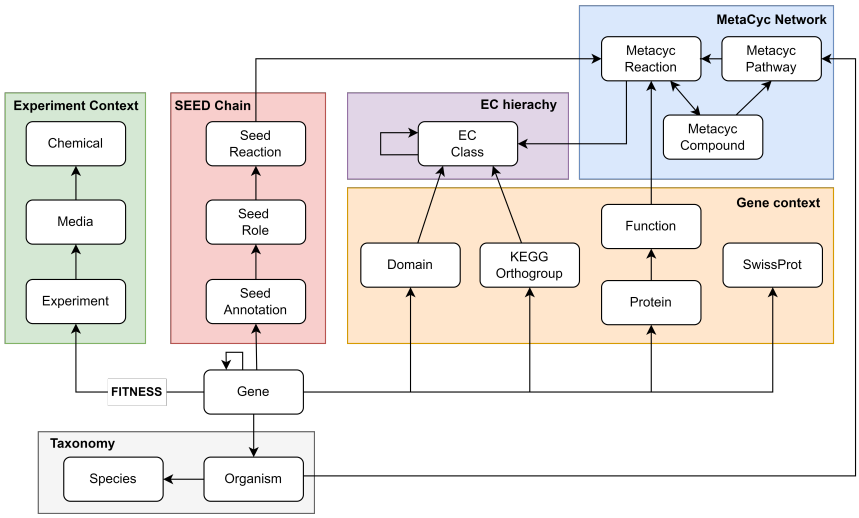


Fig. 1. Schema of the released GIMME-KG knowledge graph, including `FITNESS` supervision edges connecting gene and experiment subnetworks. Split-specific training subgraphs apply additional relation filtering before bidirectional structural message passing.

3.2. *Data Processing*

To reduce the influence of extreme fitness values, we fit a quantile transformer on the training fold that maps the target distribution to an approximate standard normal using 10,000 quantile bins. Transformed targets were used for model optimization and validation-based early stopping. All predictions were inverse-transformed before computing the reported metrics using the original $\log_2$ fitness scale. We evaluated three split protocols representing generalization to held-out experiments, genes, and gene clusters. Split counts below are based on assignment seed 42 and are available in Appendix section A.

- **Experiments:** We assigned the 1,770 experiments and their associated fitness cells to one partition.
- **Genes:** We assigned the 55,421 genes and their associated fitness cells to one fold.
- **Clusters:** To reduce homology across partitions, we assigned each gene to exactly one cluster using the following priority order: KEGG orthogroup, Swiss-Prot group, connected component of `ORTHOLOGOUS` edges, or singleton, emulating CAFA gene clustering.[22,35]

3.3. *Multi-Modal Node Encoder*

As shown in Figure 2, we encode each node within the heterogeneous graph by projecting every available feature type through its own branch, then fusing the results to a common hidden vector of width H. Each branch b produces a raw vector $\mathbf{x}_v^{(b)}$ for node v, and we define $\mathcal{A}_k$ as the set of branches active on node type k. The four supported branch types are:

- **Text** ($\mathbb{R}^{d_T}$). A frozen text representation embedded with `gemini-embedding-001`[36] into a fixed 3072-dimensional vector.
- **Numeric** ($\mathbb{R}^{2n_k}$). For a node type k with n_k scalar properties, the branch takes the vector of z-normalized values paired with n_k binary flags indicating if each value is present.
- **Categorical** ($\mathbb{R}^{\sum_p e_p}$). Each categorical property p has a learned embedding table $\mathbf{E}_p$ of width $e_p = \max\big(2, \lceil\sqrt{|c_p|}\rceil\big)$, where $|c_p|$ is the property's cardinality, and are concatenated to form the branch input.
- **Vector** ($\mathbb{R}^{d_V}$). Pre-trained ESM-3[25] embeddings using sequence-length hidden states extracted from the last transformer layer and compressed into a single, 1536-d vector via mean-pooling across the amino acid dimension to return a fixed length.

The output vectors are combined into a final node representation in two stages, both using the same reusable module $\mathcal{B}$ consisting of a layer-normalization operator LN, a learned linear layer with weights $\mathbf{W}$ and bias $\mathbf{b}$, a nonlinearity $\phi = \mathrm{GELU}$, and an optional dropout layer D_p with probability p during training. For numeric and categorical input branches this layer-normalization operation is disabled, as their low dimensionality makes normalization degenerate, with it remaining active for text and ESM-3 vector branches and in the post-concatenation fusion block.

$$\mathcal{B}(\mathbf{x}) = D_p\big(\,\phi\big(\,\mathbf{W}\,\mathrm{LN}(\mathbf{x}) + \mathbf{b}\,\big)\,\big). \tag{1}$$

Each branch is first projected by its own instance of the module, $\mathcal{B}_b$, differing in both input and output width based on branch type. The branch outputs are then concatenated and passed through $\mathcal{B}_k$ (per node type as $\mathcal{A}_k$ varies across nodes), mapping the concatenation to a hidden vector of width H:

$$\mathbf{h}_{k,v}^{(0)} = \mathcal{B}_k\Big(\,\big[\,\mathcal{B}_b(\mathbf{x}_v^{(b)})\,\big]_{b\in\mathcal{A}_k}\,\Big) \in \mathbb{R}^H. \tag{2}$$

3.4. *Heterogeneous Message Passing*

Stacking the encoded representations across nodes gives $\mathbf{H}_k^{(0)} \in \mathbb{R}^{N_k\times H}$ per node type k. As shown in Figure 2, we propagate information across the split-specific training subgraph using L typed heterogeneous convolution layers. Structural relations are represented in both directions during message passing. The `FITNESS` relation is retained as the supervision signal but excluded from message passing. For the cluster-aware split, cross-partition `ORTHOLOGOUS` and `BEST_HIT` relations are also removed. Our final convolutional layers use the multi-head GATv2 attention[37] backbone. The exploratory screen also evaluated GraphSAGE as an alternative message-passing backbone, with the complete comparison provided in Appendix Table 4.

3 METHODOLOGY

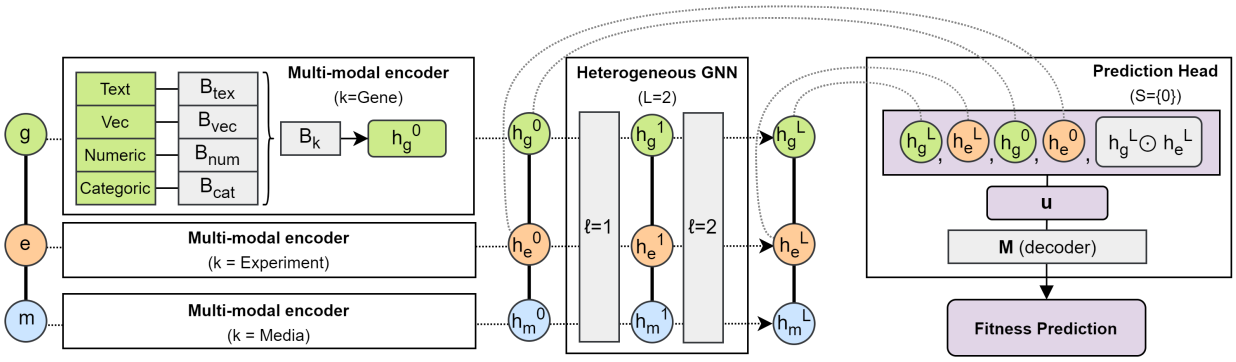


Fig. 2. Architecture schematic, detailing the Multi-modal encoders, variable depth GNN, and prediction head. Note that within the prediction head, the concatenation (cat) fusion method is displayed, and that the larger KG is represented by a simple three node graph containing a single gene, experiment and media node.

3.5. *Prediction Head*

For an input gene-experiment pair (g, e), let $\mathbf{h}_g^{(\ell)}, \mathbf{h}_e^{(\ell)}$ denote the gene and experiment representations at depth ℓ, with $\ell = 0$ being the encoder output (Equation (2)) and $\ell = L$ the final message-passing output. Let $\mathcal{S} \subseteq \{0, 1, \ldots, L-1\}$ be an optional subset of retained intermediate depths, $\mathcal{D} = \{L\} \cup \mathcal{S}$ be the full set of retained depths, and for $t \in \{g, e\}$ the stack of depth-ℓ be $\mathbf{H}_t$, corresponding with Equation (3).

$$\mathbf{H}_t = \left[\mathbf{h}_t^{(\ell)} \right]_{\ell \in \mathcal{D}} \in \mathbb{R}^{|\mathcal{D}| \times H} \tag{3}$$

The final models use two prediction-head fusion variants:

- *Concatenation (**cat**).* $\mathbf{H}_g$ and $\mathbf{H}_e$ are flattened, giving a head input whose width grows linearly with $|\mathcal{S}|$.

$$\mathbf{u} = \left[\mathrm{vec}(\mathbf{H}_g) \,;\, \mathrm{vec}(\mathbf{H}_e) \,;\, \mathbf{h}_g^{(L)} \odot \mathbf{h}_e^{(L)} \right] \tag{4}$$

- *Cross-attention (**xa**).* Each side's final representation $\mathbf{h}_t^{(L)}$ queries the *other* side's $\mathbf{H}_{t'}$ through an independent multi-head attention block.[38] Cross attention is weighted independently across depths. Final representations, cross-attention contexts, and the Hadamard interaction are then concatenated to produce a fixed width vector of size $5H$.

The final GAT models use cross-attention (**xa**), while the matched MLP uses concatenation (**cat**). The exploratory screen also evaluated skip-attention (**sa**), GraphSAGE backbones, and alternative depths, with these variants defined and compared in Appendices B and C.

Finally, the head input $\mathbf{u}$ is passed through a decoder $\mathcal{M} : \mathbb{R}^{|\mathbf{u}|} \to \mathbb{R}$, a two layer MLP with input LayerNorm, GELU nonlinearities, and dropout, producing a scalar prediction $\hat{y} = \mathcal{M}(\mathbf{u})$ on the quantile-transformed target scale. We note that the MLP-only baseline is recovered as the $L = 0$ special case of the **cat** fusion method. It uses the same multi-modal node-encoder design and decoder template as the graph models but does not perform message passing. A complete description of the architecture is provided in Appendix section B.

3.6. *Training and Evaluation*

All models use hidden width 256, dropout 0.2, and minibatches of 16,384 gene–experiment fitness pairs. We optimize the mean squared error (MSE) on the quantile-transformed targets using the Adam optimizer with peak learning rate 10^{-3}, weight decay 10^{-4}, and a 5% linear warm-up. The learning rate is reduced by a factor of 0.3 after eight validation epochs without improvement, to a minimum of 10^{-5}. Training wall times, graph/cache footprint, and inference benchmarks for the selected GAT and matched MLP models are reported in Appendix section F. An architecture screen selected GAT/$L = 2$/xa for the experiments and genes splits and GAT/$L = 6$/xa for the cluster-aware split. The full architecture sweep is provided in Appendix C. To separate model selection from confirmation, these screening runs were excluded from primary inference, with fourteen additional GAT and MLP runs then trained for each split. Model checkpoints and analysis artifacts are provided in the supplementary data release. For feature ablation assays, described in Appendix section D, we retrained the selected genes-split GAT after disabling one feature branch while leaving the corresponding nodes and graph edges intact. Performance comparisons in Table 2, Figure 3, Appendix Figure 2, and Table 3 use the best model for each split based on the lowest validation MSE on the transformed target scale. These comparisons evaluate individual selected runs, with the bootstrap intervals in the structural analysis measuring variation across test genes. For the performance over coverage analysis we compared per-organism GAT/MLP MSE across the three evaluation splits using annotation coverage, defined as the fraction of organism genes linked to at least one of SEED, MetaCyc, KEGG, or UniProt in the original KG. 95% confidence intervals (CI) for MSE differences were estimated using 1,000 paired bootstrap resamples of genes or gene families.

3.7. *GSMM performance comparison*

The GSMM iJN1463 was downloaded from the BiGG database `http://bigg.ucsd.edu/`. FBA was computed across all genes and carbon source experiments and we define essential where growth_ratio to control 0.25 or gene knockout growth approaches 0. The condition and detailed comparison description is provided within Appendix section E, along with an inference runtime analysis (Appendix section F, table 17).

3.8. *Graph Structure Analysis*

Using test predictions, we evaluated when GAT outperformed MLP by comparing test-set MSE across genes grouped by graph connectivity and family prevalence. We adjusted comparisons for differences in fitness distributions, assessed whether access to measured neighbors improved performance, and examined the relationship between neighbor and test-gene fitness. Statistical uncertainty and significance were estimated using paired bootstrapping and multiple-testing correction. Full methods are provided in Appendix section E.

Table 1. Mean and standard deviation of test performance for the GAT and MLP baseline.

Split	Model	n	MSE↓	MAE↓	ρ↑
Experiments	MLP	14	0.1361 ± 0.0016	0.2127 ± 0.0006	0.7457 ± 0.0029
	GAT/$L = 2$/xa	14	$\mathbf{0.1289 \pm 0.0016}$	$\mathbf{0.2100 \pm 0.0007}$	$\mathbf{0.7657 \pm 0.0024}$
Genes	MLP	14	0.2410 ± 0.0054	0.2551 ± 0.0015	0.5324 ± 0.0113
	GAT/$L = 2$/xa	14	$\mathbf{0.2166 \pm 0.0102}$	$\mathbf{0.2484 \pm 0.0027}$	$\mathbf{0.5960 \pm 0.0189}$
Clusters	MLP	14	0.2965 ± 0.0092	0.2719 ± 0.0017	0.4237 ± 0.0250
	GAT/$L = 6$/xa	14	$\mathbf{0.2931 \pm 0.0090}$	$\mathbf{0.2704 \pm 0.0016}$	$\mathbf{0.4263 \pm 0.0300}$

Table 2. Comparison of best MLP and GAT performance across data splits in the high essentiality stratum. "Ess." refers to *conditionally essential* fitness pairs per split.

Split	Model	Truly ess.	Pred. ess.	TP	Recall	Precision	$\mathbf{F}_1$
Experiments	GAT	10,468	6,794	**5,919**	**0.565**	**0.871**	**0.686**
	MLP	10,468	**6,951**	5,815	0.556	0.837	0.668
Genes	GAT	17,742	5,687	5,194	0.293	**0.913**	0.443
	MLP	17,742	**6,575**	**5,661**	**0.319**	0.861	**0.466**
Clusters	GAT	13,748	2,347	**2,007**	**0.146**	**0.855**	**0.249**
	MLP	13,748	**2,524**	1,990	0.145	0.788	0.245

4. Results

4.1. *Model Predictive Performance*

For interpretability, relative MSE reduction is reported as $(\overline{\text{MSE}}_{\text{MLP}} - \overline{\text{MSE}}_{\text{GAT}})/\overline{\text{MSE}}_{\text{MLP}}$. The addition of message passing through the GAT layer improved the observed mean on all three metrics for held-out experiments and genes (Table 1). Relative to the MLP, mean MSE decreased by 5.3% for experiments and 10.1% for genes, corresponding to absolute reductions of 0.0072 (p=0.000366, Wilcoxon signed-rank test, Benjamini-Hochberg (BH) correction) and 0.0244 (p=0.000366, Wilcoxon signed-rank test, BH correction), respectively. By contrast, the difference in cluster-split MSE was not significant, and GAT had lower MSE in only 7 of 14 pairs. While increasing GAT layer depth to 6 during the more restrictive cluster-aware split enabled deeper exploration of the knowledge graph's integrated metabolic networks, there was no significant improvement over the MLP model.

We conduct an exploratory analysis to determine which node features support the increase in performance observed under gene-split fitness prediction for graph-observed nodes. Each run was compared with the full GAT using the same seed. The experiment-text ablation resulted in the largest observed change in performance (+0.0106 ΔMSE, paired-t $p = 0.00627$, BH-adjusted $p = 0.10659$); feature coverage, including field-level coverage for numeric inputs, and the complete ablation sweep are reported in Appendix D.

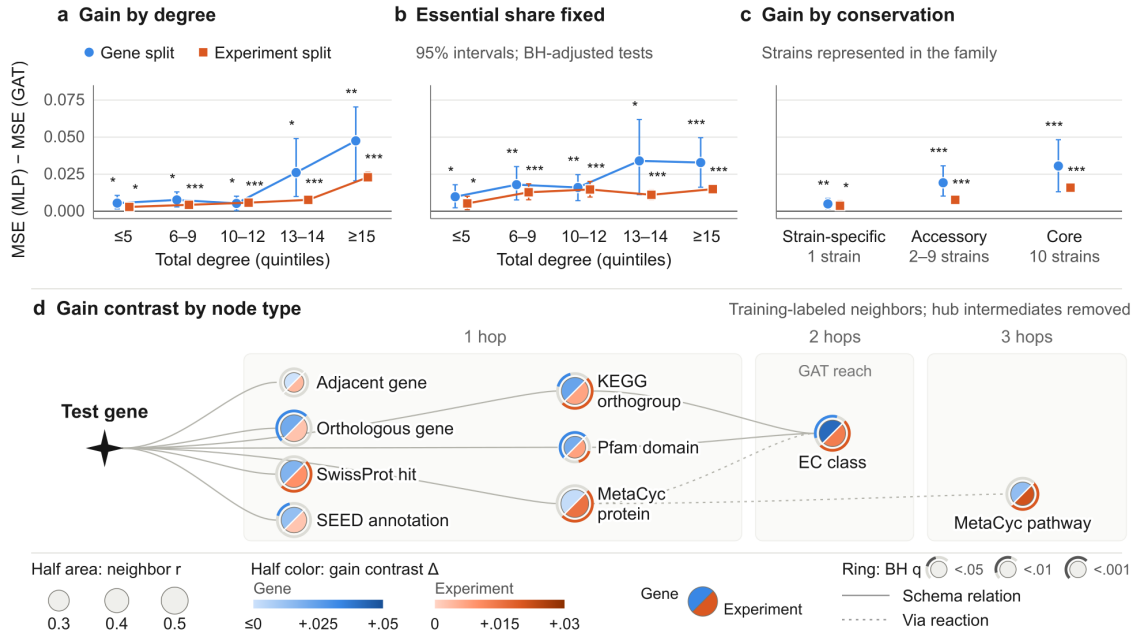


Fig. 3. GAT performance gains in relation to knowledge-graph structure. Gains over MLP are shown across test-gene connectivity before and after adjusting for fitness composition (a–b), and by gene-family conservation (c). Panel d relates gain to the availability and fitness similarity of training-labeled neighbors across graph node types and path lengths. Nodes diagonally display gene and experiment splits, shade denotes delta gain contrast and area neighbor r. Ring length around each half encodes the contrast BH-adjusted q at specified levels; an empty grey track indicates q $\leq$ 0.05. All panels use 0.05. 0.01, and 0.001 as confidence levels and 95% paired bootstrap intervals.

4.2. *Graph Structure, Conservation, and Strong-Defect Prediction*

Graph structure is not homogeneous at the gene level, with different levels of gene annotation, experimental validation, and integration into biological and metabolic knowledge bases. 39.7% of gene nodes belong to families represented in all ten sampled strains, 45.7% to families represented in two to nine strains, and 14.6% to strain-specific families (Appendix Table 10). Core genes are more connected, with mean degree 15.0 versus 3.8 for strain-specific genes. GAT's MSE gain increased from the lowest to highest degree quintile in both the experiment split (0.0029 to 0.0229) and the gene split (0.0055 to 0.0475; Figure 3a). Conditionally essential pairs were enriched in high-degree bins and holding their share fixed flattened the experiment-split curve above the lowest quintile, whereas a larger gain persisted in the upper gene-split quintiles (Figure 3b). Gains also increased from strain-specific to core families, from 0.0037 to 0.0159 in experiments and from 0.0049 to 0.0305 in genes (Figure 3c). In the gene split, labeled-neighbor coverage through orthology, KEGG, Pfam, SEED, and EC node types were associated with larger mean per-gene gains (BH q<0.05; Figure 3d). Regression gains did not translate into a uniform improvement in other performance metrics (Table 2). In the experiment split, GAT recovered 104 more true-essential pairs than MLP, with higher precision and recall over MLP (0.871 and 0.565 versus 0.837 and 0.556, respectively). In the gene split, GAT traded higher precision (0.913 versus 0.861) for lower recall (0.293 versus 0.319) and F1 (0.443 versus

Table 3. Comparison of flux balance analysis (FBA) and GFM1 predictions against experimental RB-TnSeq measurements for cold-start genes in *Pseudomonas putida* KT2440.

Model	AUROC	AUPRC	F_1	Precision	Recall	Accuracy	Pearson
FBA (iJN1463)	0.733	0.294	0.455	0.392	0.544	0.932	0.435
GFM1 + xa	**0.983**	**0.874**	**0.712**	**0.919**	**0.581**	**0.976**	**0.823**

0.466). In the cluster split, GAT precision was higher (0.855 versus 0.788), while recall and F1 differed only slightly.

Test genes with level-1 Enzyme Commission (EC) class (1–7) exhibited MAE gains varied across split protocols. GAT achieved lower MAE in all seven classes under experiment holdout, 5/7 under gene holdout, and 6/7 under cluster holdout. Isomerases and ligases favoured MLP under gene holdout, while lyases favored MLP under cluster holdout. GAT exhibited strong performance for the conditional essentiality set as well, with lower MAE in 7/7, 4/7, and 5/7 categories (Appendix Figure 2). Across the observed species gene annotation-coverage range (80.8–90.4%), GAT achieved strong performance in experiment and gene splits: the corresponding paired 95% CI for the MSE difference excluded zero in favor of GAT for 8/10, 6/10 organisms, while no organism had CI clearing zero for the cluster split.

4.3. *GFM1 outperforms a GSMM when predicting fitness growth defects*

GFM1 substantially outperforms iJN1463 derived fitness predictions across both classification and continuous prediction metrics (Table 3). Its average precision was 0.874 versus 0.294, approximately 3.0 times the FBA value and 16.8 times the 0.05195 positive prevalence. At the specified thresholds, GFM1 produced 7 false positives versus 115 for FBA, a reduction of 108 (93.9%), while recalling 79 of 136 strong-defect pairs (58.1%) versus 74 (54.4%). Pearson correlation with experimental fitness was 0.823 for GFM1 and 0.435 for FBA.

5. Discussion

5.1. *Fitness prediction under cluster-aware generalization remains challenging*

Our results show that microbial gene-knockout effects can be predicted in two settings: generalization from known genes to held-out experiments and cold-start prediction for held-out genes. Performance declined under the gene-cluster split as similar genes were withheld during training. In the experiment and gene splits, pretrained embeddings and biological context allow the model to transfer information across related conditions and genes. Much of this information is unavailable or restricted in the cluster split, reducing message passing performance gain over the non-graph baseline.

Class imbalance may contribute to this result: most gene–experiment pairs have little or no measured fitness effect, so a model can minimize average error by predicting values near the center of the distribution. Conditionally essential pairs comprised only 1.71% of the cluster test set but had a highly uncertain MSE gap, 0.483 (95% family-bootstrap interval -1.178, 2.044).

The gap on other pairs was smaller and positive, 0.0050 (0.0005, 0.0113), indicating that rare, large defects contribute disproportionately to uncertainty in the aggregate comparison.

5.2. *Relational structure improves prediction of conditionally essential phenotypes*

Message passing improves precision over matched non-graph baselines, whereas recall and F1 improve only in the experiment and cluster splits as listed in Table 2. GAT is numerically higher in seven of the nine split-by-metric comparisons, and Figure 3 links larger regression gains to connectivity, conservation, and selected annotation nodes. Fixing the conditionally essential share shows that test composition explains part of the connectivity association. These findings support the addition of message passing by adding more precise strong fitness defect calls and context-dependent regression benefits. They also demonstrate that large-scale RB-TnSeq measurements can serve as training data for predicting phenotypes in unmeasured gene–condition combinations.

Under the tested architectures and optimization settings, two-layer GATs performed best in the experiment and gene splits, so it is possible that local graph context is more useful than additional rounds of message passing. However, encoder depth alone does not establish that deeper resources, such as the EC hierarchy or MetaCyc compound network, are uninformative: their contributions may also depend on graph connectivity, feature quality, sampling, and optimization. The experiment-text ablation had the largest observed MSE increase, but its reported effect did not survive BH correction (adjusted p=0.10659). Weak or uncertain effects of individual ablations could reflect redundancy among node features, compensation through graph topology, or limited statistical sensitivity. Still, GFM1 showed strong MSE performance across the range of species gene-annotation coverage, indicating that coverage as low as 80% does not result in diminished performance. Future analysis of novel or poor coverage gene sets is needed to fully enumerate GAT performance in that regime.

5.3. *GFM1 complements mechanistic genome-scale metabolic models*

GFM1 showed stronger agreement with experimental RB-TnSeq measurements than flux balance analysis with iJN1463 for the evaluated *Pseudomonas putida* KT2440 benchmark. In particular, GFM1 achieved higher classification performance and stronger correlation with continuous fitness measurements while producing fewer false-positive growth-defect predictions (Table 3). These results support the value of learned models for predicting bacterial gene fitness effect.

The approaches nevertheless provide different forms of information. Genome-scale metabolic models impose stoichiometric and mass-balance constraints and generate reaction- and flux-level hypotheses, whereas GFM1 produces empirical fitness predictions without mechanistic explanation. Their strengths are therefore complementary rather than directly interchangeable. GFM1 could prioritize gene–condition pairs with predicted fitness defects, after which a genome-scale metabolic model could be used to investigate candidate metabolic mechanisms.

5.4. *Limitations and future work*

Several limitations should be considered. First, the training data are taxonomically restricted to ten *Pseudomonas* strains. An organism-level split would more closely represent prediction for a previously unmeasured species or strain and would test whether gene and experiment representations transfer across phylogenetic lineages. Expanding training to the broader Fitness Browser collection could improve the learning of conserved sequence-to-function relationships, although differences in experimental design, annotation quality, and phylogenetic representation would require careful control.

Structural graph metrics such as degree, annotation coverage, and conservation are correlated, and their associations with prediction gain do not identify causal edge contributions. The non-significant cluster-split comparison and the restricted organism/condition scope of the FBA benchmark limit evaluation of how the models generalize beyond tested conditions.

Finally, alternative architectures may also improve information flow through the heterogeneous graph. Because the GAT and MLP also use different prediction heads (**xa** vs **cat**), their performance difference reflects the complete model configurations and cannot be attributed solely to message passing. For example, knowledge-source-specific encoders or mixture-of-experts modules could process individual subnetworks before combining their representations through a learned gate. Protein structures, protein–protein interactions, transcriptomics, and proteomics could provide additional and potentially complementary signals. Future work should evaluate these modalities alongside alternative heterogeneous-graph encoders and more stringent phylogenetic generalization protocols.

6. Conclusion

We developed GFM1, a heterogeneous graph neural network for predicting condition-specific microbial fitness following gene inactivation. Using the GIMMME-KG microbial metabolic knowledge graph, GFM1 improved mean regression performance over the MLP in experiment and gene holdouts, while the cluster-split mean MSE advantage was not significant. Larger regression gains were associated with graph connectivity and conservation, with conditionally essential enrichment explaining part of the connectivity pattern. At a fixed strong-defect threshold, GFM1 improved precision in all splits but traded recall for precision in held-out genes. Generalization to withheld gene clusters and recovery of rare, large fitness defects remain important challenges.

GFM1 also showed stronger agreement with experimental RB-TnSeq measurements than flux balance analysis with iJN1463 on the evaluated *Pseudomonas putida* KT2440 benchmark. These results demonstrate that protein and text embeddings can be integrated with biological knowledge graphs to learn condition-dependent genotype–phenotype relationships. Future work should expand the taxonomic and experimental diversity of the training data, improve transfer to novel gene families and explore architecture changes, and evaluate the resulting representations in experimental-design and functional-annotation workflows. Together, these results highlight GFM1's potential to prioritize informative gene–condition combinations for experimental validation and accelerate microbial functional characterization.

Funding Statement

This work was supported in part by the U.S. Department of Energy (DOE), Office of Biological and Environmental Research (BER), as part of BER's Genomic Science Program (GSP), and is a contribution of the Pacific Northwest National Laboratory (PNNL) Secure Biosystems Design Science Focus Area "Persistence Control of Engineered Functions in Complex Soil Microbiomes." It is also part of the Predictive Phenomics Initiative and conducted under the Laboratory Directed Research and Development Program. Pacific Northwest National Laboratory is a multiprogram national laboratory operated by Battelle for the U.S. Department of Energy under Contract No. DE-AC05-76RL01830.

Competing Interests

No competing interest is declared.

References

1. A. Wagner, Genotype sampling for deep-learning assisted experimental mapping of a combinatorially complete fitness landscape, *Bioinformatics* **40**, p. btae317 (2024).
2. S. Valavarasu, Y. Sangu and T. Mahapatra, Prediction of antibiotic resistance from antibiotic susceptibility testing results from surveillance data using machine learning, *Scientific Reports* **15**, p. 30509 (2025).
3. K. M. Wetmore, M. N. Price, R. J. Waters, J. S. Lamson, J. He, C. A. Hoover, M. J. Blow, J. Bristow, G. Butland, A. P. Arkin *et al.*, Rapid quantification of mutant fitness in diverse bacteria by sequencing randomly bar-coded transposons, *MBio* **6**, 10 (2015).
4. A. J. Borchert, A. Bleem and G. T. Beckham, Experimental and analytical approaches for improving the resolution of randomly barcoded transposon insertion sequencing (rb-tnseq) studies, *ACS Synthetic Biology* **11**, 2015 (2022), PMID: 35657709.
5. M. N. Price, K. M. Wetmore, R. J. Waters, M. Callaghan, J. Ray, H. Liu, J. V. Kuehl, R. A. Melnyk, J. S. Lamson, Y. Suh *et al.*, Mutant phenotypes for thousands of bacterial genes of unknown function, *Nature* **557**, 503 (2018).
6. M. Rajkumar, L. B. Bruno and J. R. Banu, Alleviation of environmental stress in plants: The role of beneficial pseudomonas spp., *Critical Reviews in Environmental Science and Technology* **47**, 372 (2017).
7. M. G. Wilson and S. Pandey, Pseudomonas aeruginosa, in *StatPearls [Internet]*, (StatPearls Publishing, 2023)
8. H. Asif, D. J. Studholme, A. Khan, M. Aurongzeb, I. A. Khan and M. K. Azim, Comparative genomics of an endophytic pseudomonas putida isolated from mango orchard, *Genetics and Molecular Biology* **39**, 465 (2016).
9. T. Joshi and D. Xu, Quantitative assessment of relationship between sequence similarity and function similarity, *BMC genomics* **8**, p. 222 (2007).
10. Y. Ma, H. Chen, J. Kang, X. Guo, C. Sun, J. Xu, J. Tao, S. Wei, Y. Dong, H. Tian *et al.*, The hidden markov model and its applications in bioinformatics analysis, *Genes & Diseases* **13** (2026).
11. G. A. Pavlopoulos, F. A. Baltoumas, S. Liu, O. Selvitopi, A. P. Camargo, S. Nayfach, A. Azad, S. Roux, L. Call, N. N. Ivanova *et al.*, Unraveling the functional dark matter through global metagenomics, *Nature* **622**, 594 (2023).
12. J. Nogales, J. Mueller, S. Gudmundsson, F. J. Canalejo, E. Duque, J. Monk, A. M. Feist, J. L. Ramos, W. Niu and B. O. Palsson, High-quality genome-scale metabolic modelling of pseu-

domonas putida highlights its broad metabolic capabilities, *Environmental microbiology* **22**, 255 (2020).
13. A. Antonakoudis, R. Barbosa, P. Kotidis and C. Kontoravdi, The era of big data: Genome-scale modelling meets machine learning, *Computational and structural biotechnology journal* **18**, 3287 (2020).
14. A. M. Abdel-Haleem, H. Hefzi, K. Mineta, X. Gao, T. Gojobori, B. O. Palsson, N. E. Lewis and N. Jamshidi, Functional interrogation of plasmodium genus metabolism identifies species-and stage-specific differences in nutrient essentiality and drug targeting, *PLoS computational biology* **14**, p. e1005895 (2018).
15. Z. Harteveld, A. Van Hall-Beauvais, I. Morozova, J. Southern, C. Goverde, S. Georgeon, S. Rosset, M. Defferrard, A. Loukas, P. Vandergheynst *et al.*, Exploring "dark matter" protein folds using deep learning, *bioRxiv* , 2023 (2023).
16. J. Zhong, L. Li, R. Dannenfelser and V. Yao, Benchmarking gene embeddings from sequence, expression, network, and text models for functional prediction tasks, *bioRxiv* , 2025 (2025).
17. S. Ji, S. Pan, E. Cambria, P. Marttinen and P. S. Yu, A survey on knowledge graphs: Representation, acquisition, and applications, *IEEE Transactions on Neural Networks and Learning Systems* **33**, 494 (2022).
18. C. Peng, F. Xia, M. Naseriparsa and F. Osborne, Knowledge graphs: Opportunities and challenges, *Artificial intelligence review* **56**, 13071 (2023).
19. M. Schlichtkrull, T. N. Kipf, P. Bloem, R. v. d. Berg, I. Titov and M. Welling, Modeling relational data with graph convolutional networks (2017).
20. X. Wang, H. Ji, C. Shi, B. Wang, P. Cui, P. Yu and Y. Ye, Heterogeneous graph attention network (2021).
21. Z. Ye, Y. J. Kumar, G. O. Sing, F. Song and J. Wang, A comprehensive survey of graph neural networks for knowledge graphs, *IEEE Access* **10**, 75729 (2022).
22. M. Kanehisa, The kegg database, in *'In silico'simulation of biological processes: Novartis Foundation Symposium 247*, 2002.
23. R. Caspi, R. Billington, I. M. Keseler, A. Kothari, M. Krummenacker, P. E. Midford, W. K. Ong, S. Paley, P. Subhraveti and P. D. Karp, The metacyc database of metabolic pathways and enzymes-a 2019 update, *Nucleic acids research* **48**, D445 (2020).
24. A. Bairoch and R. Apweiler, The swiss-prot protein sequence database and its supplement trembl in 2000, *Nucleic acids research* **28**, 45 (2000).
25. T. Hayes, R. Rao, H. Akin, N. J. Sofroniew, D. Oktay, Z. Lin, R. Verkuil, V. Q. Tran, J. Deaton, M. Wiggert *et al.*, Simulating 500 million years of evolution with a language model, *Science* **387**, 850 (2025).
26. J. Ma, J. Song, N. D. Young, B. C. Chang, P. K. Korhonen, T. L. Campos, H. Liu and R. B. Gasser, 'bingo'—a large language model-and graph neural network-based workflow for the prediction of essential genes from protein data, *Briefings in Bioinformatics* **25**, p. bbad472 (2024).
27. A. J. Borchert, A. C. Bleem, H. G. Lim, K. Rychel, K. D. Dooley, Z. A. Kellermyer, T. L. Hodges, B. O. Palsson and G. T. Beckham, Machine learning analysis of rb-tnseq fitness data predicts functional gene modules in pseudomonas putida kt2440, *Msystems* **9**, e00942 (2024).
28. M. Li, X. Cai, S. Xu and H. Ji, Metapath-aggregated heterogeneous graph neural network for drug-target interaction prediction, *Briefings in Bioinformatics* **24**, p. bbac578 (January 2023).
29. Z. Li, X. Tu, Y. Chen and W. Lin, HetDDI: A pre-trained heterogeneous graph neural network model for drug-drug interaction prediction, *Briefings in Bioinformatics* **24**, p. bbad385 (September 2023).
30. Q. Ye, Y. Zeng, L. Jiang, Y. Kang, P. Pan, J. Chen, Y. Deng, H. Zhao, S. He, T. Hou and C.-Y. Hsieh, A knowledge-guided graph learning approach bridging phenotype- and target-based drug discovery, *Advanced Science* **12**, p. e2412402 (April 2025).

31. H. Wen, S. Carpenter, K. McGinnis, A. Nelson, K. Smith and T. Hong, Esssubgraph improves performance and generalizability of mammalian essential gene prediction with large networks, *GigaScience* **14**, p. giaf136 (01 2025).
32. Y. Xiao *et al.*, FuseLinker: leveraging LLM-generated text and domain knowledge for enhanced link prediction on biomedical knowledge graphs, *Journal of Biomedical Informatics* **158**, p. 104735 (2024).
33. R. Overbeek, R. Olson, G. D. Pusch, G. J. Olsen, J. J. Davis, T. Disz, R. A. Edwards, S. Gerdes, B. Parrello, M. Shukla, V. Vonstein, A. R. Wattam, F. Xia and R. Stevens, The SEED and the rapid annotation of microbial genomes using subsystems technology (RAST), *Nucleic Acids Research* **42**, D206 (2014).
34. A. G. McDonald and K. F. Tipton, Enzyme nomenclature and classification: the state of the art, *The FEBS Journal* **290**, 2214 (2023).
35. P. Radivojac, W. T. Clark, T. R. Oron, A. M. Schnoes, T. Wittkop, A. Sokolov, K. Graim, C. Funk, K. Verspoor, A. Ben-Hur *et al.*, A large-scale evaluation of computational protein function prediction, *Nature methods* **10**, 221 (2013).
36. Google, Gemini api: Embeddings `https://ai.google.dev/gemini-api/docs/embeddings`, Accessed: 2026-01-19.
37. S. Brody, U. Alon and E. Yahav, How attentive are graph attention networks?, in *International Conference on Learning Representations*, 2022.
38. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, Attention is all you need, in *Advances in neural information processing systems*, 2017.

INSPIRE: A Network-Based Method for Quantifying Functional Convergence in Cancer Driver Genes

Charles Jacobs[1], George C. Linderman[2,3], Jakob A. Woerner[4], Hyun Jo Kim[5], Gurkan Bebek[5,6,7†]

[1]*Department of Biochemistry, Case Western Reserve University, Cleveland, Ohio, USA.*
[2]*Department of Biomedical Engineering, Case Western Reserve University, Cleveland, Ohio, USA.*
[3] *Applied Mathematics Program, Yale University, New Haven, CT, USA.*
[4]*Genomics and Computational Biology, University of Pennsylvania, Philadelphia, PA, USA.*
[5]*Department of Nutrition, Case Western Reserve University, Cleveland, Ohio, USA.*
[6] *Center for Proteomics and Bioinformatics, Case Western Reserve Univ., Cleveland, Ohio, USA.*
[7]*Department of Computer and Data Sciences, Case Western Reserve Univ., Cleveland, Ohio, USA.*
[†]*E-mail: gurkan.bebek@case.edu*

Recent genomic studies have identified numerous cancer driver genes, many of which occur at low frequencies across patient populations. This long tail of rare driver events suggests that tumorigenesis can arise through alterations in distinct genes that converge on common functional pathways. Characterizing these relationships requires approaches that extend beyond individual gene properties to the network contexts in which driver genes operate. Here, we investigate whether functional similarities among cancer drivers are better captured by comparing the signaling networks they perturb than by conventional gene-centric measures. To address this question, we developed INSPIRE (Integrative Network Similarity PIpEline), a framework for quantifying similarity among within-species biological networks. Applying INSPIRE to the signaling subnetworks of 28 colorectal cancer driver genes mapped topological similarity to downstream functional convergence, clustering driver-gene pathways into distinct, biologically coherent modules based on network architecture. Independent validation using large-scale CRISPR dependency datasets demonstrated that genes assigned to the same INSPIRE cluster exhibited significantly stronger co-essentiality than genes from different clusters ($p - value < 0.038$, r_{rb}=0.567), supporting the functional relevance of the network-derived groupings. Exploratory epistasis analyses identified additional candidate interactions within these clusters. At the same time, aggregation of drivers into INSPIRE-defined functional modules revealed significant network-level mutual exclusivity (FDR = 0.0149) that was not apparent at the individual-gene level. Together, these results demonstrate that signaling-network architecture captures functional convergence among genetically heterogeneous driver genes and provides a complementary framework for studying pathway organization in cancer.

Keywords: Network Similarity, Cancer Driver Genes, Protein Interaction Networks, Functional Convergence, Colorectal Cancer

Availability: Source code and data are available at github.com/bebeklab/inspire.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

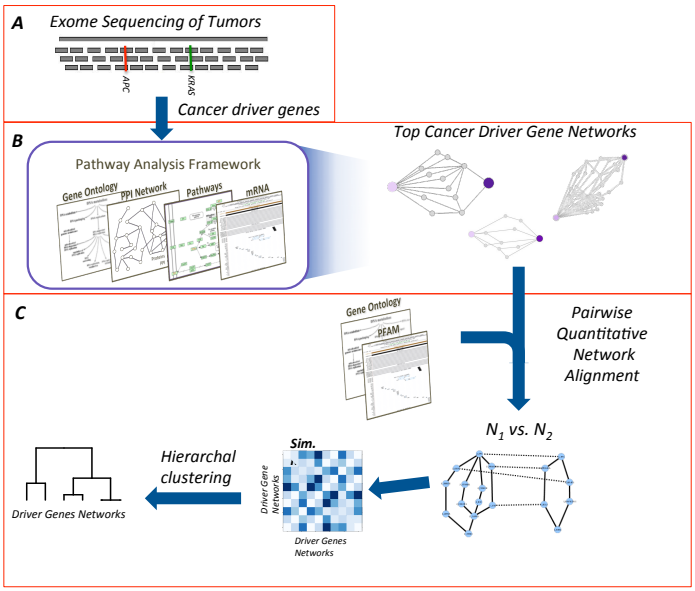


Fig. 1. **Overview of the study design and workflow.** (A) Identification of candidate cancer driver genes exhibiting co-occurrence or mutual exclusivity patterns in patient tumors. (B) Generation of functional signaling subnetworks connecting colorectal cancer (CRC) driver genes to the initiating driver *APC*.[6] (C) Schematic of the `INSPIRE` pipeline: pairwise subnetwork alignments are computed using protein family (Pfam) and Gene Ontology (GO) features to yield similarity scores, which are hierarchically clustered to identify functionally related driver genes based on network topology.

1. Introduction

Large-scale sequencing studies have identified numerous cancer driver genes across many tumor types.[1–4] Despite these advances, understanding how genetically diverse driver genes converge on common biological processes remains a major challenge.

A key challenge in cancer genomics is understanding how driver mutations give rise to malignant phenotypes. Although some driver genes are highly recurrent, many established drivers occur at modest frequencies, suggesting that distinct mutations may converge on common cellular processes. Numerous network-based approaches have therefore been developed to identify disease-relevant subnetworks.[5–10] In colorectal cancer, many well-established drivers beyond *APC* are individually mutated in fewer than 20% of patients, despite their recognized biological importance.[1,11]

The study of mutational patterns has revealed that some driver gene mutations tend to co-occur, suggesting they cooperate in tumor development, while others are mutually exclusive.[12–14] This mutual exclusivity suggests a functional redundancy, where a mutation in either gene is sufficient to dysregulate a common pathway, making a second mutation in the other gene redundant for cancer progression. This supports the notion that functionally similar genes can leverage convergent cellular processes to drive cancer. Understanding and quantifying this functional similarity is central to our work.

In this paper, we introduce INSPIRE , a novel comparative network analysis pipeline to investigate cancer driver genes and elucidate their functional interrelationships (Figure 1). We hypothesize that a pathway-centric approach, which quantifies the similarity between the molecular networks dysregulated by driver genes, will provide a more comprehensive and biologically accurate depiction of their relationships than traditional gene-centric comparisons based solely on sequence or ontology.

A noteworthy contribution of our methodology is our unique approach to quantifying similarity. To measure subnetwork similarity, we introduce a novel network-alignment algorithm

specifically designed to align functionally related PPI subnetworks from the same organism, termed *within-species network alignment*. Crucially, our algorithm seeks to align conserved patterns of interactions rather than just individual proteins, aiming to capture true functional conservation between pathways.

While there is a rich literature on comparing networks across species (cross-species alignment),[15–23] these methods are not directly suited to comparing functionally distinct subnetworks within the same organism. We identify the following as key requirements for our task:

(1) The alignment must be global, yielding a unified similarity score between the two networks.
(2) The mapping must be unambiguous; each protein from one network should map to at most one protein in the other.
(3) Identical proteins shared between two networks (a feature unique to within-species comparison) must be preferentially aligned to each other, reflecting their identical function.

Existing network alignment solutions often fail to meet these requirements. For instance, IsoRank[24] and its derivatives are global alignment tools, but they do not satisfy requirement (3). Because IsoRank's scoring is based on the PageRank algorithm, a protein's alignment score depends on its neighbors' scores. Consequently, two identical proteins may not receive the maximum score if their network neighborhoods differ. The GraphAlignment R package only handles cross-species network alignment and cannot align genes to functional equivalents while also mapping identical genes to each other between networks.[25] Other prominent algorithms focus on local alignments and thus do not satisfy requirement (1).[15,19] While network propagation methods (e.g., HotNet2[26]) identify genes close to each other in network space, and graph embeddings (e.g., node2vec[27]) capture global structural roles, neither explicitly solves the subgraph isomorphism problem to find topologically identical wiring patterns independent of sequence homology. INSPIRE fills this gap by providing a bespoke alignment strategy that quantifies both protein functional similarity and the topological consistency of their interactions.

The primary objective of INSPIRE is to provide a quantitative, pathway-level framework to measure the functional convergence of altered signaling networks. This approach leverages established molecular interactions and functional annotations to determine how distinct driver mutations converge on common mechanisms. By projecting disparate mutational events onto comparative alignment graphs, INSPIRE identifies systems-level functional redundancies that may be missed by gene-centric analyses.

2. Methods

2.1. *Cancer Driver Gene Networks*

To model signaling pathways of colorectal cancer (CRC) driver genes, we constructed protein-protein interaction (PPI) subnetworks connecting each driver to *APC*, a common initiating gene in CRC.[1] Starting from 140 candidate driver genes, putative signaling networks were generated using an adapted PETALS framework[6] on a CRC-specific PPI network derived from IntAct, STRING, and MSS TCGA-COAD expression data.[28–30]

Pathway significance was assessed using GO-derived association rules and a permutation-based empirical null model. Networks were retained at FDR ≤ 0.05, resulting in significant

subnetworks for 28 driver genes. Additional implementation details are provided in Supplementary Methods (Section S1.7).

2.2. *Quantifying Network Similarity*

For two subnetworks to be considered functionally similar, both the proteins and the topological context in which they interact should be similar.[19,31] To incorporate both factors, we formulate network similarity as a network alignment problem.

We represent two subnetworks as graphs $G_A = (V_A, E_A)$ and $G_B = (V_B, E_B)$, where vertices ($v \in V$) correspond to proteins and edges ($e \in E$) to interactions. We define an alignment $\mathbb{A}$ as a one-to-one mapping of a subset of edges from one network to the other. Formally, $\mathbb{A}$ is a function that maps a subset of edges $E_0 \subseteq E_A$ to E_B. The choice of E_0 is part of the optimization process. This edge-based mapping must induce a consistent, one-to-one mapping of the corresponding vertices. That is, if an edge $(u, v) \in E_0$ is mapped to $(x, y) \in E_B$, then no other edge involving u or v can be mapped to an edge involving x or y.

The quality of an alignment is given by an alignment score $W(\mathbb{A}) = \sum_{e \in E_0} w(e, \mathbb{A}(e))$, where $w(e, \mathbb{A}(e))$ is the score for aligning interaction e with its counterpart $\mathbb{A}(e)$. The goal is to find the alignment $\mathbb{A}$ that maximizes this total score.

2.3. *Constructing the Alignment Graph*

To efficiently find the best alignment between G_A and G_B, we adapt the alignment graph concept.[19] The alignment graph $\Psi(G_A, G_B) = (U_\Psi, F_\Psi)$ is constructed as follows:

- Vertices: Each vertex $u \in U_\Psi$ represents a potential alignment of one interaction from each network, i.e., $u = (e_A, e_B)$ where $e_A \in E_A$ and $e_B \in E_B$. Thus, $U_\Psi = E_A \times E_B$.
- Edges: Two vertices in the alignment graph, $u_i = (e_{A_i}, e_{B_i})$ and $u_j = (e_{A_j}, e_{B_j})$, are connected by an edge $f_{ij} \in F_\Psi$ if their corresponding interactions are topologically consistent. We consider three types of connectivities:

(1) Extension: The two pairs of aligned edges share a protein in both graphs (Edge in Graph A matches Edge in Graph B).
(2) Insertion/Deletion (Indel): The pairs share a protein in one graph but are separated by one hop in the other (Edge in Graph A matches a 2-hop path in Graph B).
(3) Jump: The pairs are separated by one hop in both networks (2-hop path in Graph A matches 2-hop path in Graph B).

The edges f_{ij} are weighted based on the scores of the nodes they connect (u_i, u_j) and a penalty for topological inconsistencies (indels and jumps). We then seek a path through this alignment graph that corresponds to a valid, high-scoring alignment.

2.4. *Alignment Scoring*

The scoring scheme is designed to reward both sequence/functional similarity of proteins and topological consistency of interactions.

Scoring Nodes of the Alignment Graph: A node $u = (e_A, e_B)$ in the alignment graph represents an aligned interaction pair. Its weight, $w(e_A, e_B)$, reflects the biological similarity of this pairing. This score is based on the sequence and functional similarity of the corresponding endpoint proteins, calculated using Pfam domains[32] and Gene Ontology (GO) annotations.[33]

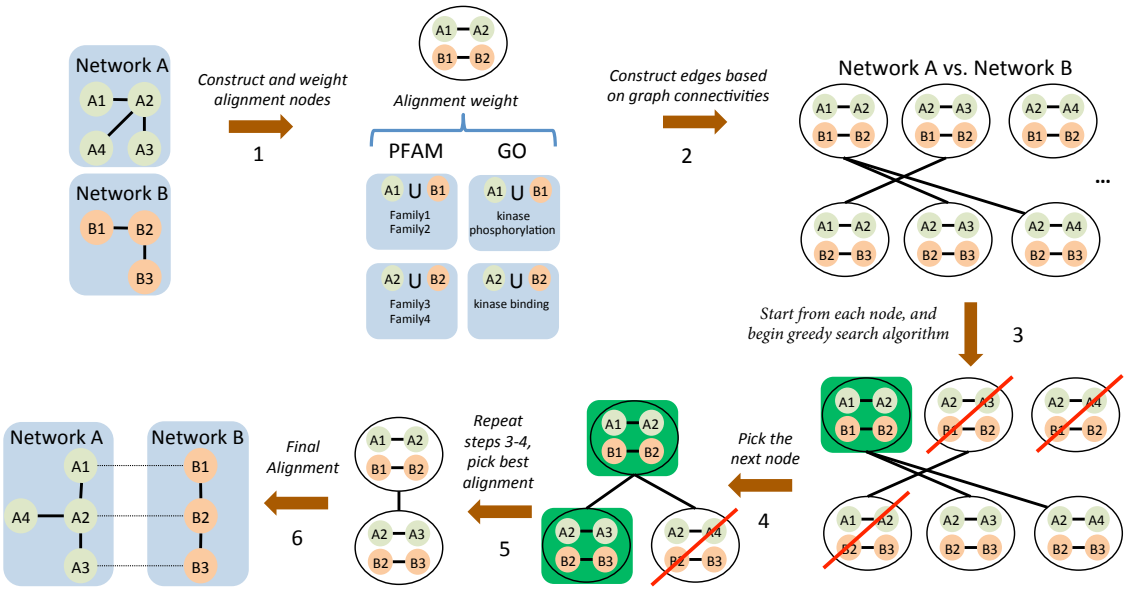


Fig. 2. The INSPIRE alignment algorithm. (1) An alignment graph is constructed in which each node represents a potential correspondence between interactions from networks A and B and is weighted by protein similarity. (2) Nodes are connected according to topological consistency (extensions, indels, and jumps). (3–5) Starting from each node, a greedy MST-inspired search iteratively adds the most favorable consistent correspondence (equivalently, the minimum-cost edge after similarity-to-cost transformation) while removing conflicting alternatives. (6) The procedure is repeated for all starting nodes, and the highest-scoring alignment is returned.

By aligning interactions (edges) rather than just proteins (nodes), we can better capture conserved functional associations.[19]

Scoring Sequence Similarity: We use the Jaccard index to compare the sets of Pfam domains for the proteins in the aligned interactions. For two interactions $e_A = (v_{A_i}, v_{A_j})$ and $e_B = (v_{B_k}, v_{B_l})$, the sequence similarity is:

$$s_{pfam}(e_A, e_B) = \frac{|\mathsf{PF}(v_{A_i}) \cap \mathsf{PF}(v_{B_k})|}{|\mathsf{PF}(v_{A_i}) \cup \mathsf{PF}(v_{B_k})|} \cdot \frac{|\mathsf{PF}(v_{A_j}) \cap \mathsf{PF}(v_{B_l})|}{|\mathsf{PF}(v_{A_j}) \cup \mathsf{PF}(v_{B_l})|} \tag{1}$$

Where $\mathsf{PF}(v)$ is the set of Pfam domains for protein v and $PF(v) = \emptyset$, the score is defined as 0.

Scoring Functional Similarity: We use a semantic similarity measure based on GO Biological Process (BP) annotations to assess functional relationships. The information content (IC) of the shared annotations for two proteins is given by:[34]

$$IC(S_i, S_j) = -\log_2 \frac{|G_{\Lambda(S_i,S_j)}|}{|G_r|} \tag{2}$$

Here, $G_{\Lambda(S_i,S_j)}$ is the set of all genes annotated with the most specific common ancestor of the GO terms for genes S_i and S_j, and G_r is the total set of genes in GO. We use this to compute the functional similarity of two interactions:

$$s_{GO}(e_A, e_B) = IC(v_{A_i}, v_{B_k}) \cdot IC(v_{A_j}, v_{B_l}) \tag{3}$$

This structure mirrors the sequence similarity score, ensuring that corresponding proteins in the two interactions are compared.

Overall Similarity Between Two Interactions: The total weight of aligning two interactions is the sum of these scores:

$$w(e_A, e_B) = s_{pfam}(e_A, e_B) + s_{GO}(e_A, e_B) \tag{4}$$

While semantic similarity packages like GOSemSim offer detailed hierarchy-aware scoring, we employed a Jaccard-based metric here to prioritize computational efficiency for large-scale graph comparisons. We found this provided a sufficient baseline for topological alignment.

2.5. *Scoring the Alignment Graph and Normalization of Scores*

Raw alignment scores were normalized by the maximum attainable score for the smaller network, producing similarities on the interval [0,1]. Pairwise distances were defined as $\hat{s}_{AB}$ and used for downstream clustering (See Section S1.1 for more details).

2.6. *Finding the Optimal Alignment*

Finding the globally optimal alignment is NP-hard.[35] We therefore employ a greedy heuristic search algorithm to find a high-scoring alignment. The challenge is to find a path or tree in the alignment graph Ψ that corresponds to a valid, one-to-one mapping. A Minimum Spanning Tree (MST) algorithm cannot be used because it does not enforce a one-to-one constraint.

Our algorithm is inspired by greedy methods like Prim's algorithm but includes a critical modification: when a node corresponding to the alignment (e_A, e_B) is added to the solution tree, all other nodes in the alignment graph that involve either edge e_A or edge e_B are immediately pruned (removed from consideration). This pruning step enforces the one-to-one mapping constraint and dramatically reduces the search space. The algorithm is initiated from every possible node in the alignment graph. The algorithm then scores the resulting alignments and returns the highest-scoring valid alignment as the final result. To enable MST-style greedy search, the algorithm transforms alignment similarity scores into edge costs using an inverse relationship. Consequently, the algorithm expands the alignment by selecting the minimum-cost candidate at each step.

The algorithm proceeds as follows:

`Initialize:`

$E_{traversed} = \emptyset$

$V_{traversed} = \{x\}$, where $x = (e_{A_x}, e_{B_x})$ is a node from $\mathbb{A}$.

$V_{available} = V_\Psi \backslash [(e_{A_x}, *), (*, e_{A_x}), (e_{B_x}, *), (*, e_{B_x})]$

`Repeat until` $V_{available} = \emptyset$

Select (x, y), where $y = (e_{A_y}, e_{B_y}) \in V_{available}$ with minimum weight in Ψ, $x \in V_{traversed}$, and $y \notin V_{traversed}$

$V_{traversed} = V_{traversed} \cup y$

$E_{traversed} = E_{traversed} \cup (x, y)$

$V_{available} = V_{available} \backslash [(e_{A_y}, *), (*, e_{A_y}), (e_{B_y}, *), (*, e_{B_y})]$

The algorithm is repeated from every node in the alignment graph. The traversal with

the minimum weight corresponds to the best alignment of the graphs. The time complexity of this algorithm depends on the choice of an edge at each step, which requires $O(|V|^2)$ time. However, this can be improved further by using heaps to find the minimum-weight edges in the algorithm's inner loop.

2.7. *Sequence and Gene Ontology-Based Classification of Genes*

As a control (gene-centric) method, we computed a pairwise similarity score w' directly between driver genes, omitting network information. This score combines sequence and GO similarity analogously to our interaction scoring:

$$w'(v_i, v_j) = s'_{pfam}(v_i, v_j) + s'_{GO}(v_i, v_j) = \frac{|\mathsf{PF}(v_i) \cap \mathsf{PF}(v_j)|}{|\mathsf{PF}(v_i) \cup \mathsf{PF}(v_j)|} + IC(v_i, v_j) \tag{5}$$

2.8. *Methodological Benchmarking against MAGNA++ and IsoRank*

To compare with established network alignment approaches, we also performed pairwise alignments using MAGNA++[36] and IsoRank[24] with the same subnetworks as input. Default parameters were used unless otherwise specified. We analyzed network similarity scores from each method using the same clustering framework described below.

2.9. *Clustering*

Stable clustering of the 28 CRC driver genes was performed using ConsensusClusterPlus[37] on the pairwise INSPIRE network similarity matrix. The similarity matrix was symmetrized and converted to a distance matrix as $d_{ij} = 1 - s_{ij}$. Consensus clustering used Partitioning Around Medoids (PAM) with 1,000 bootstrap resamplings of 80% of genes per iteration, testing $k = 2, \ldots, 8$.

2.10. *Co-essentiality Analysis*

CRISPR gene effect (Chronos) scores were obtained from the Cancer Dependency Map.[38] A selective essentiality filter was applied for each gene pair (A, B); only cell lines where at least one gene showed a Chronos score below $\tau = -0.3$ were included in the pairwise Spearman correlation. Common essential ($> 50\%$ of lines) and low-signal genes (< 4 essential lines) were excluded. The primary co-essentiality analysis used the resulting n=24 driver genes. Within-cluster vs. between-cluster co-essentiality was compared using a one-sided Wilcoxon rank-sum test, with effect size reported as the rank-biserial correlation $r_{rb} = W/(n_1 \times n_2)$.

2.11. *Somatic Mutation Profiles and Statistical Evaluation*

Mutational profiles for the 28 driver genes were obtained from TCGA MSS-COAD tumors.[30] We assessed associations between functional similarity and mutational co-occurrence using Mantel tests comparing network-based and mutation-based distances (10,000 permutations).

To overcome the statistical power limitations inherent to the low tumor mutational burden (TMB) in MSS COAD, we implemented a module-level mutual exclusivity framework. Using the Ward.D2 functional dendrogram, genes were partitioned into $K = 6$ clades via tree cutting. For each patient, a clade was designated as mutated if at least one constituent gene contained a somatic mutation. Pairwise mutual exclusivity between clades was evaluated using a one-sided Fisher's Exact Test (alternative = "less"), testing whether co-mutations occurred significantly less frequently than expected by chance. P-values were adjusted for multiple testing using the Benjamini-Hochberg False Discovery Rate (FDR) procedure.

2.12. *Epistasis Proxy Analysis*

To assess functional epistasis between within-cluster gene pairs, we stratified CRISPR dependency scores for a target gene (Gene B) by mutation status of a driver gene (Gene A) across MSS adenocarcinoma cell lines (n=251). Mutation status was derived from the DepMap damaging mutation matrix.[39] Differential dependency was tested using the Wilcoxon rank-sum test. Effect size was quantified using Cliff's delta (δ), a non-parametric dominance statistic ranging from -1 to $+1$, classified using thresholds: negligible ($|\delta| < 0.147$), small ($0.147 - 0.33$), medium (0.33–0.474), and large (≥ 0.474).[40] Given the small mutant group sizes (median $n_{mut} = 5$), results are reported as exploratory. Cliff's delta is the primary metric, and raw p-values are reported without multiple-testing correction.

3. Results

To investigate functional convergence among cancer driver genes, we developed `INSPIRE`, a network-alignment framework that quantifies similarity between protein-protein interaction (PPI) networks. We applied this framework to study driver genes in colorectal cancer (CRC), where a long tail of infrequently mutated genes suggests that distinct genetic alterations may converge on common pathways.

3.1. *Driver-Gene Networks*

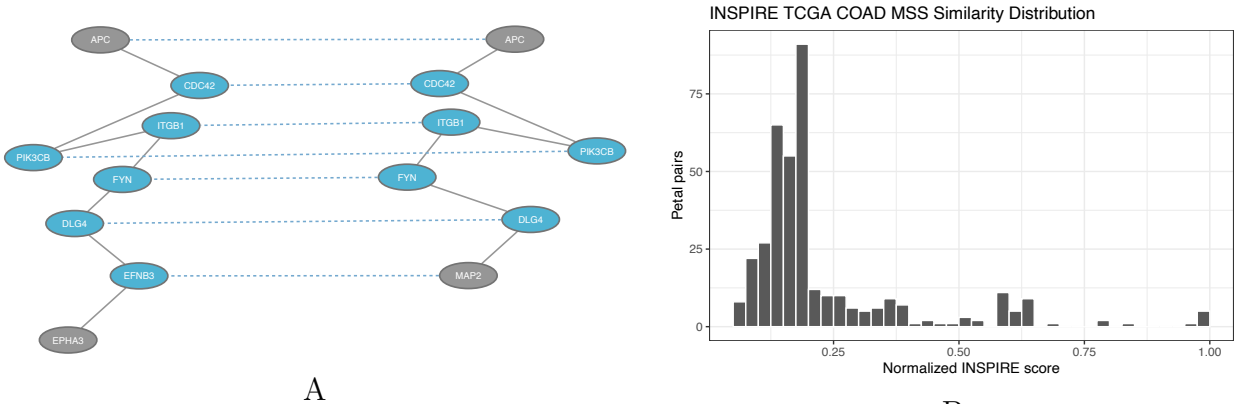
Our analysis began with 140 candidate CRC driver genes. For each gene, we constructed a signaling subnetwork connecting it to *APC*, a key initiator of colorectal tumorigenesis.[41,42] We derived these networks from a CRC-specific PPI graph built from curated databases and TCGA gene-expression data (see Methods 2.1). We identified 28 genes with high-confidence subnetworks (FDR ≤ 0.05), (Supp. Table S1). The resulting subnetworks ranged in size from 4 to 29 proteins.

3.2. *Network similarity reveals functional relationships among driver genes*

We applied INSPIRE to all pairwise comparisons among the 28 CRC driver-gene networks. Each alignment produced a normalized network similarity score reflecting similarity of signaling architecture rather than sequence homology.

The distribution of pairwise similarity scores revealed substantial heterogeneity among the driver networks, with most pairs exhibiting low similarity and a smaller subset displaying strong similarity. Using the resulting distance matrix, hierarchical clustering identified biologically coherent groups of driver genes. Members of known functional modules, including the TGF-β pathway genes *SMAD2*, *SMAD3*, and *SMAD4*, clustered closely together. Similarly, genes with established pathway-level relationships but limited sequence similarity frequently appeared within the same network-derived clusters.

To obtain a stable cluster assignment, we applied consensus clustering using Partitioning Around Medoids (PAM) with 1,000 bootstrap resamplings. Evaluation of solutions ranging from $k = 2$ to $k = 8$ identified five robust clusters (Supp. Figures S6 and S7), selected based on consensus heatmaps and biological interpretability. The final clustering exhibited positive separation in the original INSPIRE similarity space, with an average silhouette width of 0.132, indicating non-random structure among the inferred pathway relationships.



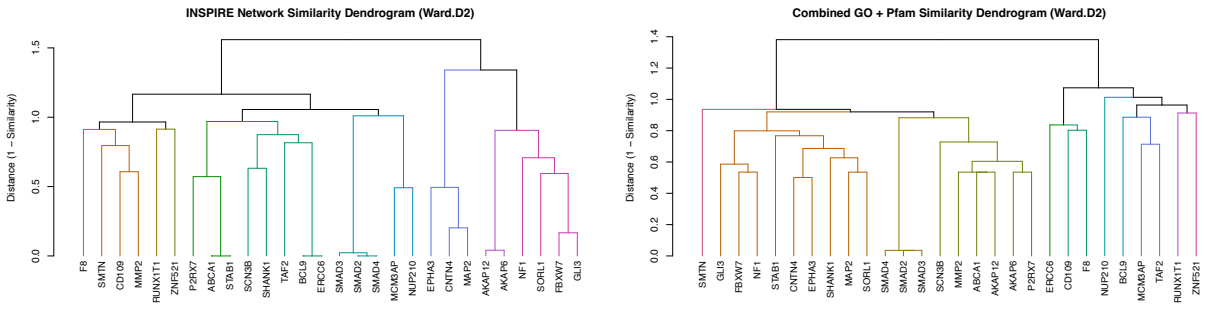
A B

Fig. 3. A. Example alignment of two CAN-gene networks (*EPHA3* (left) vs. *MAP2* (right)). CAN-genes are colored grey, and dashed (blue) edges show aligned nodes (hence edges) that returned the best score. This alignment has a normalized score of 0.69735. The alignment includes EFNB3 and MAP2. Despite distinct sequences, their network topology indicates that they occupy equivalent positions in the signaling cascade, illustrating functional convergence. B. Histogram of normalized alignment scores ($\hat{s}_{AB}$). All 28 networks are compared pairwise. While most pairs show low similarity, a subset ($\sim 10\%$) achieves high similarity scores ($\hat{s}_{AB} \geq 0.5$), suggesting functional convergence.

3.3. *Network-based clustering captures functional relationships beyond gene-level similarity*

To assess whether network alignment provides information beyond gene-level annotations, we compared INSPIRE clustering against a control approach based on Gene Ontology semantic similarity and Pfam domain similarity (Figure 4). Whereas the control approach evaluates genes individually using shared annotations and domain content, INSPIRE additionally incorporates the signaling-network context surrounding each driver gene.

Both approaches recovered the canonical TGF-β module (SMAD2, SMAD3, SMAD4). However, INSPIRE additionally identified several relationships not apparent from gene-level annotations. Most notably, MCM3AP and NUP210 formed a distinct two-gene cluster, and AKAP12, AKAP6, NF1, SORL1, FBXW7, and GLI3 were grouped within a common higher-



A B

Fig. 4. A. Hierarchical clustering of CAN-gene networks using distance scores from our INSPIRE algorithm. The clustering reveals functional relationships based on network similarity. B. Hierarchical clustering of the same CAN-genes using a gene-centric distance measure that combines GO semantic similarity[43] and sequence similarity based on Pfam domains.[32]

order clade. These findings suggest that signaling-network context captures functional relationships beyond sequence and annotation similarity.

Importantly, the biological relevance of the INSPIRE-derived clusters was supported by independent validation analyses. Genes assigned to the same network-defined clusters exhibited significantly greater co-essentiality than those in different clusters, and several candidate epistatic interactions were identified within cluster boundaries. Together, these results indicate that network topology captures biologically meaningful relationships that complement information obtained from sequence- and annotation-based similarity measures.

3.4. *Functional validation using co-essentiality analysis*

To independently evaluate whether INSPIRE clusters represent biologically coherent functional modules, we analyzed CRISPR dependency data from the Cancer Dependency Map. Co-essentiality was quantified using Chronos gene effect scores. Because CRC driver genes are enriched for tumor suppressors that are nonessential in most cell lines, we applied a selective-essentiality filtering strategy prior to analysis. Two genes (*MCM3AP* and *TAF2*) were classified as common essentials and removed, while *F8* and *SHANK1* were excluded due to insufficient signal. The final analysis therefore included 24 genes. Gene pairs assigned to the same INSPIRE cluster exhibited significantly stronger co-essentiality than pairs belonging to different clusters (one-sided Wilcoxon rank-sum test, $p = 0.038$, rank-biserial correlation $r_{rb} = 0.567$). This result indicates that genes grouped by network alignment display more similar dependency profiles across cancer cell lines, consistent with convergence on common biological processes. The same directional trend remained detectable when all 28 genes were included ($p = 0.055$), whereas the standard unfiltered analysis showed no signal ($p = 0.91$). Although cluster separation was evident in INSPIRE similarity space (average silhouette width $= 0.132$), separation in co-essentiality space was nearly absent (average silhouette width $= -0.025$).

3.5. *Epistasis proxy analysis identifies candidate functional interactions*

We next investigated whether genes grouped within the same INSPIRE cluster also exhibited evidence of functional interaction. Using 251 microsatellite-stable adenocarcinoma cell lines, we tested whether damaging mutations in one cluster member altered CRISPR dependency on another member.

Among 134 within-cluster directional gene pairs, nine showed nominal significance ($p < 0.05$), although none remained significant after multiple-testing correction. Several interactions nevertheless exhibited large effect sizes, including *SMAD2* $\rightarrow$ *SMAD4* ($\delta = -0.69$, $p = 0.004$), *EPHA3* $\rightarrow$ *CNTN4* ($\delta = -0.72$, $p = 0.014$), and *SMTN* $\rightarrow$ *MMP2* ($\delta = 0.76$, $p = 0.025$) . All three large-effect interactions occurred within INSPIRE-defined clusters (Figure 5).

3.6. *Functional Network Architecture is Independent of Global Co-occurrence but Exhibits Module-Level Mutual Exclusivity*

To test whether functional network similarity correlates with patient-level co-mutation, we performed a global Mantel test between the functional distance matrix and the mutational distance matrix (Figure S4). The test revealed no significant global correlation ($p = 0.50$). This result demonstrates that global network distance does not linearly track pairwise mutation

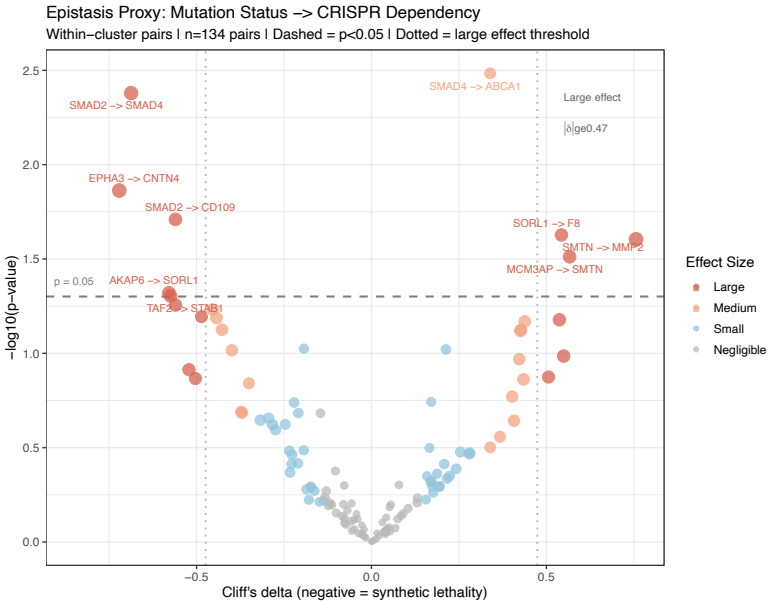


Fig. 5. Epistasis proxy analysis of within-cluster INSPIRE driver gene pairs. Volcano plot of Cliff's delta (effect size) versus -log10(p-value) for 134 directional within-cluster gene pairs tested across MSS adenocarcinoma cell lines. Points are colored by effect size magnitude.[40] A negative delta indicates synthetic lethality; a positive delta indicates epistatic buffering. Dashed line: p = 0.05. Results are exploratory given small mutant group sizes (median n=5 mutant lines per gene).

co-occurrence across the entire panel.

Pairwise single-gene mutual exclusivity tests between adjacent dendrogram leaves were statistically underpowered due to the characteristic low mutation frequencies of MSS COAD driver genes (typically 3%–20% prevalence). For instance, testing the adjacent TGF-β pathway components *SMAD2* (27 mutations) and *SMAD4* (67 mutations) yielded 4 observed co-mutated patients against an expected baseline of 5.71 patients (Odds Ratio = 0.627, $p = 0.286$, FDR = 1.00). While functionally convergent, single-gene sample sizes in MSS COAD lack the statistical power required to differentiate minor co-occurrence drops from stochastic background noise. To resolve this limitation, we aggregated individual low-frequency genes into $K = 6$ higher-order functional clades based on dendrogram topology (Supp. Table S3). Aggregating genes into functional modules elevated module mutation rates (e.g., 20%–50% prevalence), restoring statistical power. Pairwise Fisher's exact tests across clades identified significant pathway-level mutual exclusivity. Notably, Clade 2: *AKAP12, AKAP6, FBXW7, GLI3, NF1, SORL1* and Clade 6: *SMAD2, SMAD3, SMAD4* exhibited strong mutual exclusivity (Odds Ratio = 0.460, $p = 9.96 \times 10^{-4}$, FDR = 0.0149) (Table S2).

3.7. *Comparison with Existing Network Alignment Methods*

To evaluate whether standard alignment frameworks could resolve functional convergence among CRC driver networks, we benchmarked our approach against two widely used global network alignment tools: MAGNA++[36] and IsoRank[24] (Supplementary Figure S8). Both methods proved ill-suited for intra-species, pathway-level comparison. Hierarchical clustering of pairwise alignments from MAGNA++ and IsoRank produced highly degenerate, unstructured dendrograms characterized by chaining artifacts and low modular resolution. Notably, neither method recovered canonical pathway relationships—such as grouping members of the TGF-β/SMAD signaling cascade (*SMAD2*, *SMAD3*, *SMAD4*)—and frequently failed to align identical proteins shared across subnetworks. These shortcomings stem from their foundational design for cross-species orthology mapping, which assumes sequence divergence and prioritizes conserved topological isomorphism across species. In contrast, by formulating align-

ment around edge-cost transformations that explicitly enforce identity mapping and reward functional conservation, `INSPIRE` resolves distinct, biologically cohesive clusters that reflect genuine pathway-level convergence.

4. Discussion

A central premise of this study is that cancer driver genes can be understood through the signaling networks they perturb rather than solely through their individual molecular characteristics. The results support this hypothesis. INSPIRE consistently identified functionally coherent groups of CRC driver genes using network topology alone and produced clusters that were more biologically meaningful than those obtained from sequence similarity or ontology-based annotations.

The strongest independent validation came from the co-essentiality analysis. Genes assigned to the same INSPIRE clusters exhibited significantly greater similarity in CRISPR dependency profiles than genes belonging to different clusters. Importantly, this pattern emerged only after applying a selective-essentiality filter tailored to tumor suppressor genes. Conventional co-essentiality analyses did not recover a comparable structure. These findings suggest that network-derived relationships capture functional convergence reflected in cellular dependency phenotypes.

The exploratory epistasis analysis further supported the biological relevance of the network-derived clusters. Although statistical power was limited by the rarity of many driver mutations, nine within-cluster gene pairs showed nominally significant interactions, several with substantial effect sizes. While these observations require further validation, the enrichment of candidate interactions within INSPIRE-defined clusters suggests that network similarity may help prioritize functionally relevant genetic relationships.

Functional similarity was not associated with global mutational co-occurrence (Mantel test, $p = 0.50$), likely reflecting the low prevalence of MSS colorectal cancer driver mutations and the limited power of pairwise co-mutation analyses. Consistent with this interpretation, mutual exclusivity emerged only after aggregating genes into INSPIRE-derived functional clades. Whereas single-gene analyses failed to identify significant associations, module-level testing revealed robust mutually exclusive interactions, suggesting that functional redundancy operates at the level of network-defined pathways. These results demonstrate that network-based aggregation can expose mutational patterns that gene-centric analyses miss.

Several limitations should be acknowledged. First, the epistasis analysis was underpowered because most driver mutations were present in only a few cell lines. Consequently, these results should be viewed as hypothesis-generating rather than definitive evidence of genetic interaction. Second, the co-essentiality validation involved a relatively small set of driver genes following filtering, which may have reduced statistical sensitivity. Finally, the inferred signaling networks depend on available protein interaction resources and expression measurements, although restricting network construction to curated interactions increases biological confidence.

Despite these limitations, the network, co-essentiality, epistasis, and mutational analyses converge on a common conclusion that the colorectal cancer driver genes frequently partici-

pate in shared functional programs despite genetic heterogeneity and mutual exclusivity. By quantifying these relationships at the level of signaling architecture, INSPIRE provides a robust framework for uncovering pathway-level organization that conventional gene-centric approaches overlook.

While this initial validation leveraged well-characterized pathways to establish baseline accuracy, this comparative engine is naturally positioned for exploratory annotation. Future applications of INSPIRE can systematically align established reference networks against those of poorly characterized genes, providing a computational pipeline to generate targeted, testable hypotheses to guide downstream experimental design.

Acknowledgement

Research reported in this publication was supported by the National Center for Research Resources (NCRR) of the National Institutes of Health under Award Number KL2RR024990. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

1. L. D. Wood, D. W. Parsons, S. Jones, J. Lin, T. Sjoblom, R. J. Leary, D. Shen, S. M. Boca, T. Barber, J. Ptak, N. Silliman, S. Szabo, Z. Dezso, V. Ustyanksky, T. Nikolskaya, Y. Nikolsky, R. Karchin, P. A. Wilson, J. S. Kaminker, Z. Zhang, R. Croshaw, J. Willis, D. Dawson, M. Shipitsin, J. K. V. Willson, S. Sukumar, K. Polyak, B. H. Park, C. L. Pethiyagoda, P. V. K. Pant, D. G. Ballinger, A. B. Sparks, J. Hartigan, D. R. Smith, E. Suh, N. Papadopoulos, P. Buckhaults, S. D. Markowitz, G. Parmigiani, K. W. Kinzler, V. E. Velculescu and B. Vogelstein, The genomic landscapes of human breast and colorectal cancers, *Science* **318**, 1108 (nov 2007).
2. D. W. Parsons, S. Jones, X. Zhang, J. C. Lin, R. J. Leary, P. Angenendt, P. Mankoo, H. Carter, I. M. Siu, G. L. Gallia, A. Olivi, R. McLendon, B. A. Rasheed, S. Keir, T. Nikolskaya, Y. Nikolsky, D. A. Busam, H. Tekleab, J. Diaz, L. A., J. Hartigan, D. R. Smith, R. L. Strausberg, S. K. Marie, S. M. Shinjo, H. Yan, G. J. Riggins, D. D. Bigner, R. Karchin, N. Papadopoulos, G. Parmigiani, B. Vogelstein, V. E. Velculescu and K. W. Kinzler, An integrated genomic analysis of human glioblastoma multiforme, *Science* **321**, 1807 (2008).
3. M. H. Bailey, C. Tokheim, E. Porta-Pardo, S. Sengupta, D. Bertrand *et al.*, Comprehensive characterization of cancer driver genes and mutations, *Cell* **173**, 371 (apr 2018).
4. Z. Sondka, S. Bamford, C. G. Cole, S. A. Ward, I. Dunham and S. A. Forbes, The COSMIC cancer gene census: describing genetic dysfunction across all human cancers, *Nature Reviews Cancer* **18**, 696 (oct 2018).
5. A. Torkamani and N. J. Schork, Identification of rare cancer driver mutations by network reconstruction, *Genome Res* **19**, 1570 (2009).
6. G. Bebek, V. Patel and M. R. Chance, Petals: Proteomic evaluation and topological analysis of a mutated locus' signaling, *BMC Bioinformatics* **11**, p. 596 (2010).
7. N. Yosef, E. Zalckvar, A. D. Rubinstein, M. Homilius, N. Atias, L. Vardi, I. Berman, H. Zur, A. Kimchi, E. Ruppin and R. Sharan, Anat: a tool for constructing and analyzing functional protein networks, *Sci Signal* **4**, p. pl1 (2011).
8. H. Sun, B. Han, X. Cheng and K. Ma, Transcriptional regulatory network and protein–protein interaction to reveal the mechanism of pancreatic cancer, *Molecular Biology Reports* **41**, 387 (dec 2013).
9. G. Wu, X. Feng and L. Stein, A human functional protein interaction network and its application to cancer data analysis, *Genome Biology* **11**, p. R53 (2010).

10. Y. Xue, G. Cooper, C. Cai, S. Lu, B. Hu, X. Ma and X. Lu, Tumour-specific causal inference discovers distinct disease mechanisms underlying cancer subtypes, *Scientific Reports* **9** (sep 2019).
11. S. D. Markowitz and M. M. Bertagnolli, Molecular origins of cancer: Molecular basis of colorectal cancer, *N Engl J Med* **361**, 2449 (Dec 2009).
12. S. Park and B. Lehner, Cancer type-dependent genetic interactions between cancer driver alterations indicate plasticity of epistasis across cell types, *Mol Syst Biol* **11**, p. 824 (Jul 2015).
13. P. Priestley, J. Baber, M. P. Lolkema, N. Steeghs, E. de Bruijn, C. Shale, K. Duyvesteyn, S. Haidari, A. van Hoeck, W. Onstenk, P. Roepman, M. Voda, H. J. Bloemendal, V. C. G. Tjan-Heijnen, C. M. L. van Herpen, M. Labots, P. O. Witteveen, E. F. Smit, S. Sleijfer, E. E. Voest and E. Cuppen, Pan-cancer whole-genome analyses of metastatic solid tumours, *Nature* **575**, 210 (oct 2019).
14. F. Sanchez-Vega, M. Mina, J. Armenia, W. K. Chatila, A. Luna *et al.*, Oncogenic signaling pathways in the cancer genome atlas, *Cell* **173**, 321 (apr 2018).
15. M. Koyuturk, A. Grama and W. Szpankowski, An efficient algorithm for detecting frequent subgraphs in biological networks, *Bioinformatics* **20 Suppl 1**, i200 (2004).
16. M. Koyuturk, Y. Kim, S. Subramaniam, W. Szpankowski and A. Grama, Detecting conserved interaction patterns in biological networks, *J Comput Biol* **13**, 1299 (2006), Koyuturk, Mehmet Kim, Yohan Subramaniam, Shankar Szpankowski, Wojciech Grama, Ananth R01 GM068959-01/GM/NIGMS NIH HHS/ J Comput Biol. 2006 Sep;13(7):1299-322.
17. J. Flannick, A. Novak, B. S. Srinivasan, H. H. McAdams and S. Batzoglou, Graemlin: general and robust alignment of multiple large interaction networks, *Genome Res* **16**, 1169 (2006).
18. B. Dost, T. Shlomi, N. Gupta, E. Ruppin, V. Bafna and R. Sharan, Qnet: a tool for querying protein interaction networks, *J Comput Biol* **15**, 913 (2008).
19. X. Guo and A. J. Hartemink, Domain-oriented edge-based alignment of protein interaction networks, *Bioinformatics* **25**, i240 (2009).
20. C. S. Liao, K. Lu, M. Baym, R. Singh and B. Berger, Isorankn: spectral methods for global alignment of multiple protein networks, *Bioinformatics* **25**, i253 (2009).
21. U. Pieper, N. Eswar, B. M. Webb, D. Eramian, L. Kelly, D. T. Barkan, H. Carter, P. Mankoo, R. Karchin, M. A. Marti-Renom, F. P. Davis and A. Sali, Modbase, a database of annotated comparative protein structure models and associated resources, *Nucleic Acids Res* **37**, D347 (2009).
22. X. Qian and B. J. Yoon, Effective identification of conserved pathways in biological networks using hidden markov models, *PLoS One* **4**, p. e8070 (2009).
23. F. Ay, M. Kellis and T. Kahveci, Submap: aligning metabolic pathways with subnetwork mappings, *J Comput Biol* **18**, 219 (2011).
24. R. Singh, J. Xu and B. Berger, Global alignment of multiple protein interaction networks with application to functional orthology detection, *Proc Natl Acad Sci U S A* **105**, 12763 (Sep 2008).
25. J. Berg and M. Laessig, Cross-species analysis of biological networks by bayesian alignment, *Proceedings of the National Academy of Sciences* **103**, 10967 (2006).
26. M. D. M. Leiserson, F. Vandin, H. T. Wu, J. R. Dobson, J. V. Eldridge, J. L. Thomas, A. Papoutsaki, Y. Kim, B. Niu, M. McLellan, M. S. Lawrence, A. G. Perez, D. Tamborero, Y. Cheng, G. A. Ryslik, N. Lopez-Bigas, G. Getz, L. Ding and B. J. Raphael, Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes, *Nature Genetics* **47**, 106 (feb 2015).
27. A. Grover and J. Leskovec, node2vec: Scalable feature learning for networks, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (Association for Computing Machinery, New York, NY, USA, 2016).
28. D. Szklarczyk, R. Kirsch, M. Koutrouli, K. Nastou, F. Mehryary, R. Hachilif, A. L. Gable, T. Fang, N. T. Doncheva, S. Pyysalo, P. Bork, L. J. Jensen and C. von Mering, The string

database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest, *Nucleic Acids Research* **51**, D638 (2023).
29. S. Orchard, M. Ammari, B. Aranda, L. Breuza, L. Briganti, F. Broackes-Carter, N. H. Campbell, G. Chavali, C. Chen, N. Del-Toro, M. Duesbury, M. Dumousseau, E. Galeota, U. Hinz, M. Iannuccelli, S. Jagannathan, R. Jimenez, J. Khadake, A. Lagreid, L. Licata, R. C. Lovering, B. Meldal, A. N. Melidoni, M. Milagros, D. Peluso, L. Perfetto, P. Porras, A. Raghunath, S. Ricard-Blum, B. Roechert, A. Stutz, M. Tognolli, K. van Roey, G. Cesareni and H. Hermjakob, The MIntAct project—IntAct as a common curation platform for 11 molecular interaction databases, *Nucleic Acids Research* **47**, D529 (2019).
30. The Cancer Genome Atlas Network, Comprehensive molecular characterization of human colon and rectal cancer, *Nature* **487**, 330 (07 2012).
31. G. Bebek and J. Yang, Pathfinder: mining signal transduction pathway segments from protein-protein interaction networks, *BMC Bioinformatics* **8**, 335 (2007).
32. M. Punta, P. C. Coggill, R. Y. Eberhardt, J. Mistry, J. Tate, C. Boursnell, N. Pang, K. Forslund, G. Ceric, J. Clements, A. Heger, L. Holm, E. L. Sonnhammer, S. R. Eddy, A. Bateman and R. D. Finn, The pfam protein families database, *Nucleic Acids Res* **40**, D290 (2012).
33. M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin and G. Sherlock, Gene ontology: tool for the unification of biology, *Nature Genetics* **25**, 25 (may 2000).
34. J. Pandey, M. Koyutürk, S. Subramaniam and A. Grama, Functional coherence in domain interaction networks, *Bioinformatics* **24**, i28 (Aug 2008).
35. C. Y. Ma and C. S. Liao, A review of protein-protein interaction network alignment: From pathway comparison to global alignment, *Computational and Structural Biotechnology Journal* **18**, 2647 (2020).
36. V. Vijayan, V. Saraph and T. Milenkovic, Magna++: Maximizing accuracy in global network alignment via both node and edge conservation, *Bioinformatics* **31**, 2409 (07 2015).
37. S. Monti, P. Tamayo, J. Mesirov and T. Golub, Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data, *Machine Learning* **52**, 91 (2003).
38. A. Tsherniak, F. Vazquez, G. Montgomery, D. A. Weir, G. Kryukov, G. S. Cowley, S. Gill, W. F. Harrington, S. Pantel, R. Jménez *et al.*, Defining a cancer dependency map, *Cell* **170**, 564 (2017).
39. A. Walther, E. Johnstone, C. Swanton, R. Midgley, I. Tomlinson and D. Kerr, Genetic prognostic and predictive markers in colorectal cancer, *Nature Reviews Cancer* **9**, 489 (jun 2009).
40. MetricGate, Cliff's Delta Effect Size Calculator `https://metricgate.com/docs/cliffs-delta/`, (2024), Online statistical calculator.
41. B. Vogelstein, E. R. Fearon, S. R. Hamilton, S. E. Kern, A. C. Preisinger, M. Leppert, Y. Nakamura, R. White, A. M. Smits and J. L. Bos, Genetic alterations during colorectal-tumor development, *N Engl J Med* **319**, 525 (1988).
42. K. W. Kinzler and B. Vogelstein, Lessons from hereditary colorectal cancer, *Cell* **87**, 159 (1996).
43. J. Pandey, M. Koyutürk, Y. Kim, W. Szpankowski, S. Subramaniam and A. Grama, Functional annotation of regulatory pathways, *Bioinformatics* **23**, i377 (Jul 2007).
44. E. Banks, E. Nabieva, R. Peterson and M. Singh, Netgrep: fast network schema searches in interactomes, *Genome Biol* **9**, p. R138 (2008).

REFinder: Mining new enzymes from metagenomes

R. Prabakaran[1] and Yana Bromberg[1,2†],
[1]*Department of Biology, Emory University, Atlanta, GA 30322, USA*
[2]*Department of Computer Science, Emory University, Atlanta, GA 30322, USA*
Email: prabakaran@emory.edu, yana.bromberg@emory.edu

Microbial metagenomes encode vast catalytic diversity, but recovering enzymes of novel, yet-undescribed functionality typically requires whole-community assembly plus a means of identifying active proteins *ab initio*. The first step of the task is compute-intensive and misses low-abundance sequences. The second is complicated by our inability to predict novel functions. REBEAN, our DNA language model, sidesteps the latter by assigning each sequencing read a high-level Enzyme Commission (EC) class or a non-enzyme label without alignment. Here we build REFinder, a pipeline that addresses both steps by routing REBEAN-annotated reads to assemble only the putative enzymatic reads that have identified catalytic signatures.

In our evaluation of 50 microbiome metagenomes, REFinder identified 1.2 to 2.1 fold more enzymes than could be recovered via homology-based annotation of the proteins from the corresponding full assemblies. Moreover, it was as much as 6.4-fold cheaper computationally than full assembly. Across all samples, REFinder identified at least three fourths and as many as 90% of the homology-accessible enzymes identified via full assembly of the complete metagenomes. Notably, a fraction of these, 22% to 45% per EC class, carried no similarity to Swiss-Prot proteins, i.e. a set of enzymes homology cannot annotate. For roughly two fifths of the over thirteen thousand such novel oxidoreductases from two saliva samples, ESMFold predicted structures aligned with TM-score≥0.7 to a characterized enzyme structure in the PDB – a substantial structural similarity without sequence homology. These results illustrate that targeted, alignment-free assembly turns even well-mapped microbiomes into a source of thousands of previously invisible but credible novel enzymes, raising our expectations for exploration of environmental microbiomes.

Availability and implementation: https://bitbucket.org/bromberglab/NF-REFinder/

Keywords: Functional Metagenomics, Enzymes, Catalytic activity, Protein Function, Pipeline, Bioinformatics, Language model.

1. Introduction

The majority of environmental microbial taxa resist standard laboratory cultivation (Rappe & Giovannoni, 2003). Thus, metagenome sequencing, i.e. sequencing an environmental sample as a whole, is the most common path to identifying genes/proteins encoded in these unculturable lineages (Quince et al., 2017). Metagenome mining enables identification of novel biocatalysts, providing

* This work was supported by the NSF (National Science Foundation) award #2310114.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

enzymes with enhanced thermal stability, activity profiles, and, potentially, even new catalytic mechanisms (Ferrer et al., 2016; Lorenz & Eck, 2005; Robinson et al., 2021).

How do we identify novel functionality encoded in metagenomic samples? Two routes are salient – (1) assembling metagenomes to long contigs/MAGs (metagenome assembled genomes) and finding/labeling encoded genes or (2) annotating functionality directly from reads. Existing read-level annotators, e.g. MG-RAST (Meyer et al., 2008) and mi-faser (Zhu et al., 2018), label individual reads by molecular function or enzyme class via calibrated homology to known genes. Read-based profilers such as HUMAnN quantify pathways and gene families in a sample (Beghini et al., 2021). These approaches are relatively straightforward, but since a short read spans only a fragment of a gene, read-level homology annotation is less complete and less reliable than full-length-gene annotation (Wommack et al., 2008) and often cannot resolve the specific gene of read-origin or its function (Carr & Borenstein, 2014). The field as a whole, therefore, deems the first route, i.e. assemble-then-annotate, where annotation runs on longer, near-complete gene (or translated protein) sequences (Hyatt et al., 2010; Nurk et al., 2017) more reliable.

Read- and assembled gene-based annotations share a constraint in that both assign function by aligning sequences to reference databases such as UniProt, KEGG, and/or eggNOG (Cantalapiedra et al., 2021; Kanehisa et al., 2016). Multiple algorithms for alignment and scoring alignments in light of functional similarity have been developed (Altschul et al., 2009; Mahlich et al., 2018; Zhu et al., 2018). Recent advances in structure prediction have also facilitated functional annotation via structural alignments (Borujeni & Salavati, 2024; van Kempen et al., 2024). In either case, however, alignment can only recognize that which resembles a known reference. Novel or divergent enzymes, i.e. the functional dark matter that dominates many microbiomes, are therefore missed by default, whether annotation runs on reads or on contigs (Vanni et al., 2022).

We previously developed REBEAN – a model that removes the homology need at the read level. Built as a DNA language model, it assigns each read an Enzyme Commission class (EC 1–7, (Tipton, 1994)) or a non-enzyme label from sequence context alone (Prabakaran & Bromberg, 2025). Because it reads functional signal from patterns rather than reference matches, and with read embeddings largely independent of sequence identity (Prabakaran & Bromberg, 2025), REBEAN recovers enzymatic reads that alignment overlooks: three- to six-fold more enzymatic reads. We could thus choose which reads to assemble (**Fig. 1**), routing only those labeled with a target EC class and yielding an activity-focused assembly (seeding) at a fraction of the cost of assembling everything, i.e. avoiding the step that dominates metagenomic compute (Pell et al., 2012; Wang et al., 2015). An idea similar to targeted assembly of specific protein families that has been demonstrated to accelerate discovery (Boyd et al., 2018; Guo et al., 2019; Li et al., 2017; Wang et al., 2015; Zhang et al., 2014), but carried out through a DNA language model, targeting functional activities.

In this work, we use 50 metagenomic samples from two studies to evaluate our seeding pipeline's ability to recover enzymes vs. a full assembly-driven pipeline. Our results highlight the potential to

reduce computational costs of metagenomic analyses while expanding our understanding of biological catalytic mechanisms.

2. Methods

2.1. *The REFinder pipeline*

We built REBEAN_Enzyme_Finder (REFinder), a read seed-based enzyme-discovery pipeline (**Fig. 1**). REFinder is a Nextflow (Di Tommaso et al., 2017) read-to-annotation workflow built around REBEAN (Prabakaran & Bromberg, 2025), a DNA language model-based classifier that labels each sequencing read into one of seven Enzyme Commission classes (EC1–7, (Tipton, 1994)) or as non-enzyme. REFinder has five core stages and six optional annotation modules. The core stages are:
(1) Data acquisition fetching reads from ENA/SRA
(2) Quality Control (QC) - quality-trimming with fastp (Chen, 2025) and FastQC (Andrews, 2010).
(3) Enzyme-read detection, in which REBEAN scores every read; selected reads of each class form that class's seed.
(4) Seed-guided assembly, in which i) BBMap (Bushnell, 2014) recruits each seed's read neighborhood from the complete sample (minid 0.90), ii) metaSPAdes (–meta; (Nurk et al., 2017)) assembles the recruited reads, and iii) Prodigal (-p meta; (Hyatt et al., 2010)) predicts genes and translates them into proteins.
(5) Consolidation of all outputs into one per-protein table.

The six optional protein-annotation modules are:

a) Enzyme annotation of proteins through mapping of REBEAN identified enzymatic reads
b) Metal-binding prediction with MebiPred (Aptekmann et al., 2022)
c) Homology search against Pfam-A families (Mistry et al., 2021) through HMM search using HMMER (Eddy, 2011)
d) Homology search against Swiss-Prot and/or TrEMBL (UniProt Consortium, 2025) through alignment using DIAMOND (Buchfink et al., 2015) to assign EC numbers.
e) Protein structure prediction using ESMFold v1(Lin et al., 2023)
f) Reliability estimation (Prabakaran & Bromberg, 2026) of protein embedding, with an option to choose between two commonly used protein language models: ESM-2 (esm2_t33_650M_UR50D) ProtT5 (prot_t5_xl_half_uniref50-enc) (Elnaggar et al., 2022; Lin et al., 2023) .

Data: We evaluated REFinder on two publicly available metagenome sets: human saliva microbiome (BioProject PRJNA396840 (Belstrøm et al., 2017), $n = 30$) and murine antibiotic-perturbation of the gut microbiome study (BioProject PRJNA504846 (Cabral et al., 2019), $n = 20$). Note that we chose these easily accessible and well-studied environments deliberately as a conservative test for evaluation of our method vs. a traditional full-assembly homology-based pipeline. Human and mouse-associated microbiomes are less-diverse than most environmental samples and are among the most deeply sequenced microbiomes available. Thus, any enzyme that

REFinder surfaces here is unlikely to be an artifact of shallow sampling. However, even in this well-mapped setting a large fraction of enzymes per class had no close Swiss-Prot matches (**Section 3.2**). As per our workflow, metagenomic reads were quality-trimmed with fastp (-q 20, –length_required 50, adapter auto-detection; (Chen, 2025)) resulting in a mean read length of 244.9 ± 13.8 bp (saliva samples) and 149.4 ± 7.4 bp (gut samples), and then processed through two routes: (1) a full community assembly (full arm), and (2) seven REFinder runs, one per EC class (EC-$N = 1 - 7$ seed arm). The full arm assembled every trimmed read. Each seed arm, instead, assembled one EC class; i.e. it took as seeds all reads that REBEAN labeled with that class, recruiting their read neighborhood from the full sample with BBMap (minid 0.90, (Bushnell, 2014)). For both full-assembly and seeded routes, reads were assembled with metaSPAdes 4.2.0 (–meta, $k = 21{,}33{,}55$; (Nurk et al., 2017)), and gene-called/translated into protein sequences with Prodigal 2.6.3 (-p meta; (Hyatt et al., 2010)).

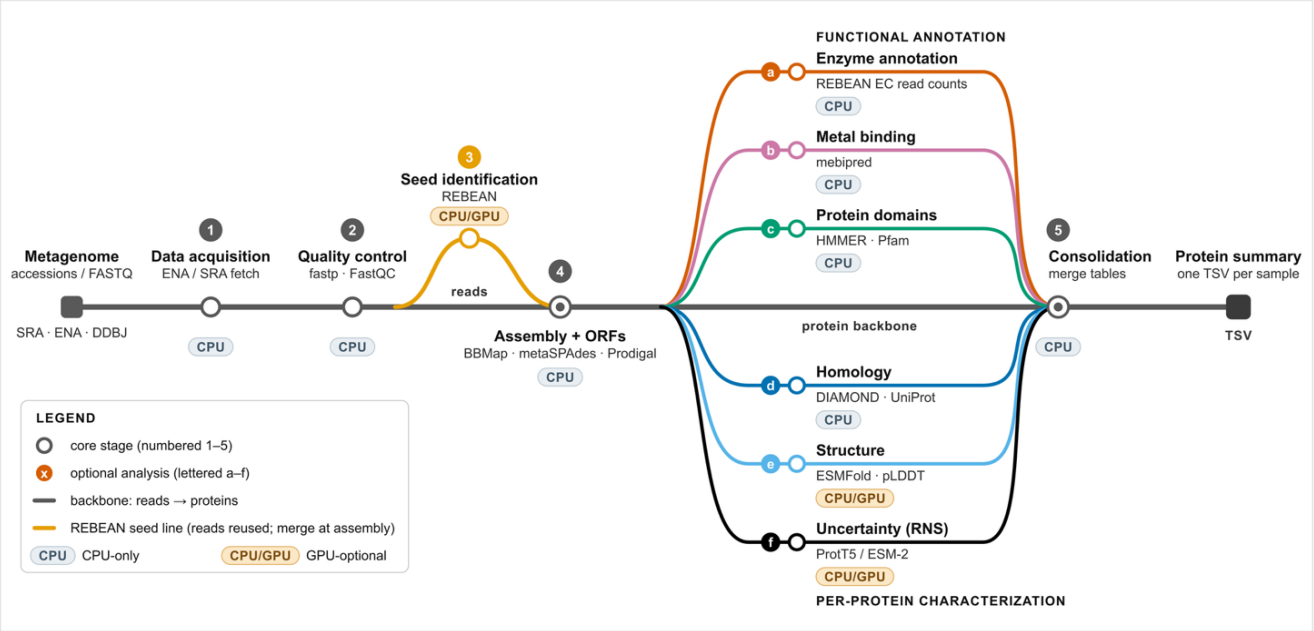


Figure 1: The REFinder pipeline. A Nextflow workflow that turns raw shotgun-metagenome reads into a functionally annotated protein table. *Inputs:* sample accessions (ENA/SRA). *Core stages:* (1) reads are fetched from source repositories, (2) quality-trimmed (fastp, FastQC); (3) REBEAN classifies every read into EC1–7 or non-enzyme, yielding per-class seeds; (4) the reads making up each seed are recruited with BBMap (minid 0.90), assembled with metaSPAdes (–meta), and gene-called with Prodigal; (5) outputs are consolidated into a protein table. *Optional annotation modules* include*:* DIAMOND homology (EC numbers), HMMER against Pfam-A, ESMFold structure prediction, mebiPred metal-binding prediction, and ESM-2/ProtT5 embedding reliability (RNS) scoring.

2.2. *Annotation of genes as enzymes*

Predicted proteins from both assembly arms were annotated as enzymes in two independent ways:

2.2.1. *Homology-based annotation*

A predicted protein was deemed a true enzyme if and only if its best Swiss-Prot hit (UniProt Consortium, 2025), assigned with DIAMOND 2.1.12 blastp (best hit, e-value ≤1E-5; (Buchfink et al., 2015)), carried exactly one EC number. Note that we performed function annotation transfer without a minimum identity or coverage threshold to maximize the recovery instead of relying on quality-centric, but low recall, approaches (e.g. (Mahlich et al., 2018)). The assigned true EC class was the first digit of that EC number, regardless of its Swiss-Prot evidence code. In addition, each read inherits the Swiss-Prot accession number of the protein's hit. We used Swiss-Prot, the manually curated subset of UniProt as reference, rather than the automatically annotated TrEMBL whose annotations are derived from a Swiss-Prot subset. This is computationally faster while retaining identical coverage of the function space. Note that Swiss-Prot is REFinder's default when sequence similarity/homology identification is requested, with TrEMBL available as an option.

2.2.2. *REBEAN-based definition*

A protein was deemed an EC-N enzyme if at least one mapped read's score reached the EC-N class-specific 90%-precision cutoff (EC1 0.944, EC2 0.971, EC3 0.947, EC4 0.829, EC5 0.888, EC6 0.863, EC7 0.698). Because classes are considered separately, proteins with reads mapping to different EC-N classes were counted multiple times. Unlike the homology-based annotations, this path can identify enzymes with no Swiss-Prot matches.

To create protein-linked identifiers, we clustered all assembly-extracted proteins (per sample, full and all seed arms separately) with MMseqs2 at 90% identity and 80% coverage (Steinegger & Soding, 2017). Each cluster was given an identifier, and we then provided each read with the cluster identifier of a protein that the read had mapped to.

2.3. *Evaluation criteria*

We compared REFinder's REBEAN-seeded assembly enzyme discovery pipeline (seeded arm) against a traditional full-assembly pipeline with homology-based annotation (full arm) on both saliva and murine samples. The comparison was done in two phases: i) evaluating recovery of known enzymes and ii) evaluating the identified novel enzymes. We also compared the computational cost of assembly in each arm.

2.3.1. *Recovery*

Comparison of known enzyme recovery between the REFinder's seeded arm and the class-matched full-assembly (full arm) was summarized by overlap of enzymes (**Eq. 1**), where A and B denote i) the distinct Swiss-Prot accession sets identified (**Section 2.2.1**) or ii) number of distinct enzyme clusters (> 90% identity and 80% coverage) (**Section 2.2.2**) assembled, from each arm computed within one EC-N class and one sample.

$$Overlap\% = \frac{A \cap B}{A \cup B} * 100 \quad \textbf{(1)}$$

2.3.2. *Novel enzymes*

A protein labelled as an enzyme via annotation transfer from a Swiss-Prot match (**Section 2.2.1**) was considered novel if the match was <30% amino-acid identity (Buchfink et al., 2015), i.e. not a reliable call (Rost, 2002). Similarly, a protein labelled as an enzyme via REBEAN assigned read labels (**Section 2.2.2**) was considered novel if it had no Swiss-Prot hit or if its best hit fell below 30% sequence identity.

2.3.3. *Computational cost*

Compute was recorded as CPU-hours for fastp, seed derivation, BBMap recruitment, metaSPAdes, Prodigal, and DIAMOND, summed over the samples of each dataset, and captured with /usr/bin/time -v and slurm accounting (sacct).

3. Results and Discussion

3.1. *Seeded assembly recovers known enzymes*

We compared the genes assembled through full-assembly arm and the seeded arm based on their ability to reconstitute homologs of Swiss-Prot enzymes. As expected due to seeding with a prediction model whose recall is below 100% (80.6% for saliva and 84.2% for murine), the seeded arm recovered somewhat fewer (~76%) homology-identifiable enzymes than the full arm (**Tables 1, S1**).

Table 1: Recovery of known enzymes from the saliva metagenomes

EC	Class	No. of distinct Swiss-Prot Enzymes identified					Overlap %
		Full arm	†Full arm w REBEAN filter	Seeded arm	Unique to full arm	Unique to seeded arm	
1.*	Oxidoreductase	2,685±451	2,341±392	2,381±403	126±31	166±45	88.5±1.3
2.*	Transferase	8,082±1462	7,132±1,308	7,242±1347	354±83	464±125	89.3±1.0
3.*	Hydrolase	5,433±931	3,850±708	3,985±749	367±75	503±116	80.1±1.2
4.*	Lyase	1,850±343	1,363±255	1,420±269	102±26	159±46	83.0±2.0
5.*	Isomerase	1,649±295	1,279±234	1,322±243	91±21	134±34	84.1±1.7
6.*	Ligase	3,114±625	2,380±469	2,518±511	190±45	328±105	81.1±2.4
7.*	Translocase	1,522±262	1,268±222	1,322±233	89±19	143±37	83.7±2.0

† To ensure a standardized efficiency comparison between the Seeded (REFinder) and Full arm assemblies, we restricted the analysis to genes containing at least one REBEAN-predicted enzymatic read. The mean and standard deviation over the 30 samples (PRJNA396840) is reported. Note counts are number of distinct Swiss-Prot accessions (**Section 2.3.1**).

To isolate the assembly stage from the seeding stage, we evaluated the seeded arm against the subset of full-assembly arm enzymes that REBEAN could detect. We required that, for each gene identified in the full arm, at least one mapped read be predicted to be of the matching EC-*N* class. On this common ground, the two arms were more closely aligned (~84%), with the seeded arm recovering slightly more enzymes than the full arm in every EC class of the saliva set (n=30) and in five of seven EC classes of the murine gut set (n=20). These results show that the two pipelines can thus be deemed comparable in recovery of known enzymes (**Tables 1, S1 & Figure 2a, S1a**).

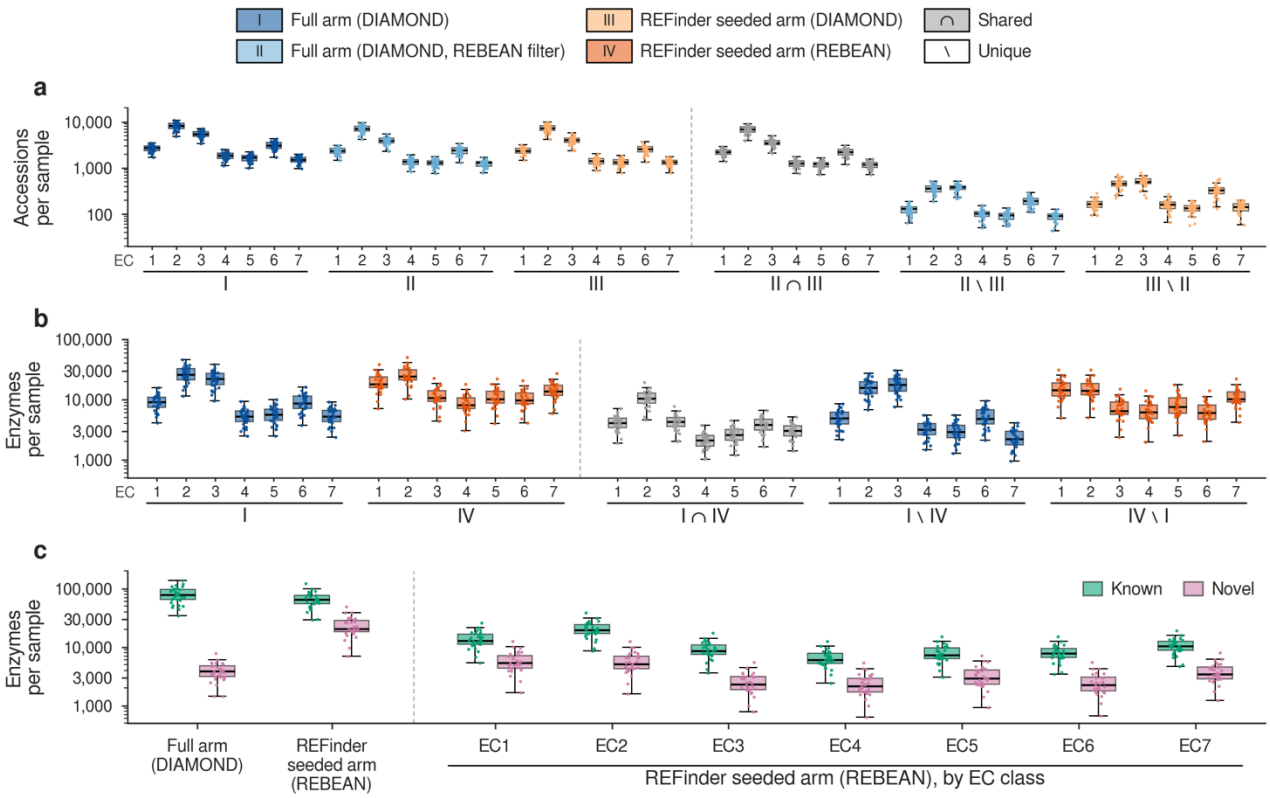


Figure 2: **Enzymes recovered from oral saliva metagenomes by full arm and REFinder's seeded arm.** Each sample was assembled in two ways: the full arm assembles all reads, and each REFinder EC-N seeded arm assembles only the reads that REBEAN classifies as EC class N (N = 1–7), plus the reads recruited to their neighborhood by BBMap. For each EC class, four sets are defined: (I) full-arm proteins whose best Swiss-Prot hit carries an EC N number (DIAMOND); (II) the enzymes of I with at least one mapped read that REBEAN classifies as EC N (REBEAN filter); (III) EC-N seeded-arm proteins whose best hit carries an EC N number (DIAMOND); (IV) EC-N seeded-arm proteins with at least one mapped read that REBEAN assigns to EC N with a probability at or above the class-specific 90%-precision cutoff (REBEAN). Comparisons are class-matched: an EC-N seeded arm is compared only with the full arm's EC-N enzymes, and an enzyme is shared (∩) only if both sets place it in the same class; \ denotes the enzymes in only one set. (a) Full versus seeded arm under the same homology annotation (II versus III), counted as distinct Swiss-Prot accessions (**Table 1, S1**) and summed over the seven classes, (b) Full arm with DIAMOND (I) versus REFinder seeded arm with REBEAN (IV), counted as enzyme clusters because REBEAN enzymes need not have a Swiss-Prot hit. Clusters are within-sample MMseqs2

clusters (90% identity, 80% coverage) pooling the full arm and all seven seeded arms, and each cluster is counted once per set. (c) Known and novel enzyme clusters in I, in IV, and in IV by EC class; a cluster is novel if no member has a Swiss-Prot hit at 30% identity or above. Boxes show the median and interquartile range (IQR), whiskers extend to the most extreme value within 1.5 × IQR, and points are individual samples; y axes in log scale.

3.2. *REFinder identifies more enzymes than sequence similarity*

Alignment-based methods can only annotate a protein when a similar sequence already exists in a reference database. Thus, they necessarily underestimate the enzymatic content of a metagenome. REFinder uses REBEAN for read annotation, i.e. prediction instead of homology. At our 90%-precision cutoff, which corresponds to approximately 30% recall in our benchmark (Prabakaran & Bromberg, 2025), REFinder identified 1.2 fold more enzymes per oral sample and 2.1 fold more per murine sample than the full-arm using alignment, on average (**Tables 2, S2 & Figure 2b, S1b**). This meant that even for some of the best studied microbiomes, we could find over twenty thousand new, sequence-unique enzymes per sample. We expect that these numbers are higher in extreme (e.g. thermophilic) or hard-to-access (e.g. ocean floor) environments. One exception to this rule is the hydrolase (EC 3), where the full arm assembles more enzymes than the seeded arm. We hypothesize that hydrolases, the largest class in Swiss-Prot with diverse sequence families, favour homology annotation over REBEAN (set at 90% precision with low recall) that is trained for functional signatures (Prabakaran & Bromberg, 2025). Note every assembled enzyme is a cluster representative of assembled genes at 90% sequence identity with 80% coverage (**Section 2.2.2**).

Table 2: Recovery of novel enzymes using REFinder seeded arm vs Full arm

EC	Class	No. of enzymes					No. of novel enzymes		Novel Overlap %†
		Full arm	Seeded arm	Overlap	Unique to full arm	Unique to seeded arm	Full arm	Seeded arm	
1.*	Oxidoreductase	9,178 ± 2,765	19,704 ± 6,777	4,165 ± 1,161	5,013 ± 1,634	15,540 ± 5,799	295 ± 104	5,941 ± 2,373	1.3 ± 0.2
2.*	Transferase	26,758 ± 8,407	26,040 ± 8,665	10,600 ± 3,154	16,157 ± 5,365	15,439 ± 5,723	1,397 ± 487	5,784 ± 2,342	4.2 ± 0.6
3.*	Hydrolase	22,261 ± 6,962	11,630 ± 3,992	4,307 ± 1,286	17,954 ± 5,750	7,323 ± 2,750	1,634 ± 569	2,571 ± 1,052	3.9 ± 0.9
4.*	Lyase	5,370 ± 1,621	8,938 ± 3,187	2,125 ± 609	3,245 ± 1,021	6,813 ± 2,697	101 ± 33	2,415 ± 1,036	0.3 ± 0.1
5.*	Isomerase	5,716 ± 1,840	11,171 ± 3,989	2,671 ± 785	3,044 ± 1,062	8,500 ± 3,370	156 ± 54	3,275 ± 1,379	0.6 ± 0.1
6.*	Ligase	9,143 ± 3,031	10,543 ± 3,570	3,898 ± 1,155	5,245 ± 1,887	6,644 ± 2,627	210 ± 81	2,519 ± 1,050	1.2 ± 0.2
7.*	Translocase	5,439 ± 1,687	14,325 ± 4,508	3,101 ± 885	2,337 ± 806	11,223 ± 3,792	152 ± 57	3,812 ± 1,430	0.7 ± 0.1

† The evaluation was carried out over 30 samples from PRJNA396840. Novel enzymes are the genes that code for proteins with less than 30% identity with Swiss-Prot database. An enzyme is a sequence representative of proteins (>90% identity), translated from each assembly that contain at least one REBEAN identified enzymatic read (seeded arm) or matches Swiss-Prot enzymes (full arm). Enzyme is considered novel if it shares less than 30% identity with proteins in Swiss-Prot database. Two enzymes from two different assemblies are said to overlap if they cluster together (≥ 90% identity with ≥ 80% of both sequences covered).

Novelty of an identified enzyme is defined by lack of homology to known protein space (**Section 2.3.2**). In this study, to be considered extant (already annotated) an enzyme needed a Swiss-Prot hit; a sequence with no hit or only with hits below 30% sequence identity counted as novel. A substantial number of REFinder-assembled enzymes are sequence-novel, ranging from 22.1 to 30.1% and 36.4 to 45.0% in oral saliva and murine gut, respectively (**Figure 2c, S1c** & **Tables 2, S2)**. In comparison, novel enzymes are a minority of what similarity searches can recover, i.e. 1.9% to 7.3% of full-assembly enzymes per EC class in saliva and 2.5% to 9.6% in the murine gut.

Using REBEAN instead of DIAMOND alignment (Buchfink et al., 2015) in the full arm resulted in a much higher novel enzyme fraction: 22.1% to 29.9% per EC class in saliva and 36.6% to 44.6% in the murine gut – similar in fraction to REFinder's seeded arm (**Table S3**), highlighting the usefulness of REBEAN, the core of REFinder pipeline, in discovery of novel enzymes.

As described above, the comparison between assembly arms was based on the number of enzyme clusters defined at 90% sequence identity, a cut-off that prevents duplicated and fragmented sequences from inflating the count. A complementary view is obtained by clustering at 30% identity, which groups related sequences together so that each distinct enzyme group is counted once. We therefore repeated the comparison at this threshold. Under this coarser definition, the seeded arm recovers 1.5-fold (saliva) and 3.1-fold (murine gut) more enzyme clusters per sample than the full arm. The difference widens because relaxing the cut-off from 90% to 30% reduces the seeded-arm cluster count by 28.5 ± 5.8% and 38.4 ± 7.2% in saliva and murine gut, respectively, whereas the corresponding reductions in the full arm are larger, at 43.2 ± 5.4% and 58.6 ± 5.6% (**Tables 2 and S2-S3; Figures 2b and S1b-S3b**). The fraction of novel enzymes in the seeded arm also increased, from 26.7% to 34.5% in saliva and from 42.9% to 51.7% in murine gut. The findings reported above are therefore robust to the definition used for enzyme distinctiveness.

3.3. *REFinder novel proteins have high structural similarity to known enzymes*

Because REFinder-only calls cannot be confirmed by sequence alignment, we asked whether they correspond to real catalytic proteins by way of structure alignment. We screened the 23,190 novel EC1 enzymes, each representative of a corresponding protein cluster and each with the best Swiss-Prot hit below 30% identity, from two saliva samples (SRR5892215 & SRR5892216) for reliable ESM2 650M (Lin et al., 2023) embeddings (RNS (Prabakaran & Bromberg, 2026). RNS is a measure of embedding uncertainty that affects downstream tasks including structure prediction, and our earlier work has shown consistent blind spots among pLM variants. We predicted structures (ESMFold average pLDDT 84.4 ± 8.9) for the 13,745 enzymes (with RNS = 0). Of these 5,180 (≥0.7 TM-score) and 772 (≥0.9 TM-score) proteins structurally aligned with EC-annotated PDB chains (Burley et al., 2025; Dana et al., 2019; van Kempen et al., 2024) despite negligible sequence identity i.e. 37.7% of proteins have considerable structural matches without sequence homology.

In addition to protein structure prediction, REFinder runs searches against Pfam HMMs that mapped 10,664 (77.5%) of the 13,745 to known (69.8%) and unknown (7.7%) domain families. Out of the 5,180 recovered enzymes that share structural similarity with PDB enzymes, 87.9% map to 1,341

distinct known and 1.6% to 101 distinct unknown Pfam domains (DUFs). The remaining 10.5% (545 of 5,180) of novel enzymes were not recoverable through hmm searches. Further illustrating the usefulness of REFinder's activity-driven targeted assembly for exploration of microbiomes.

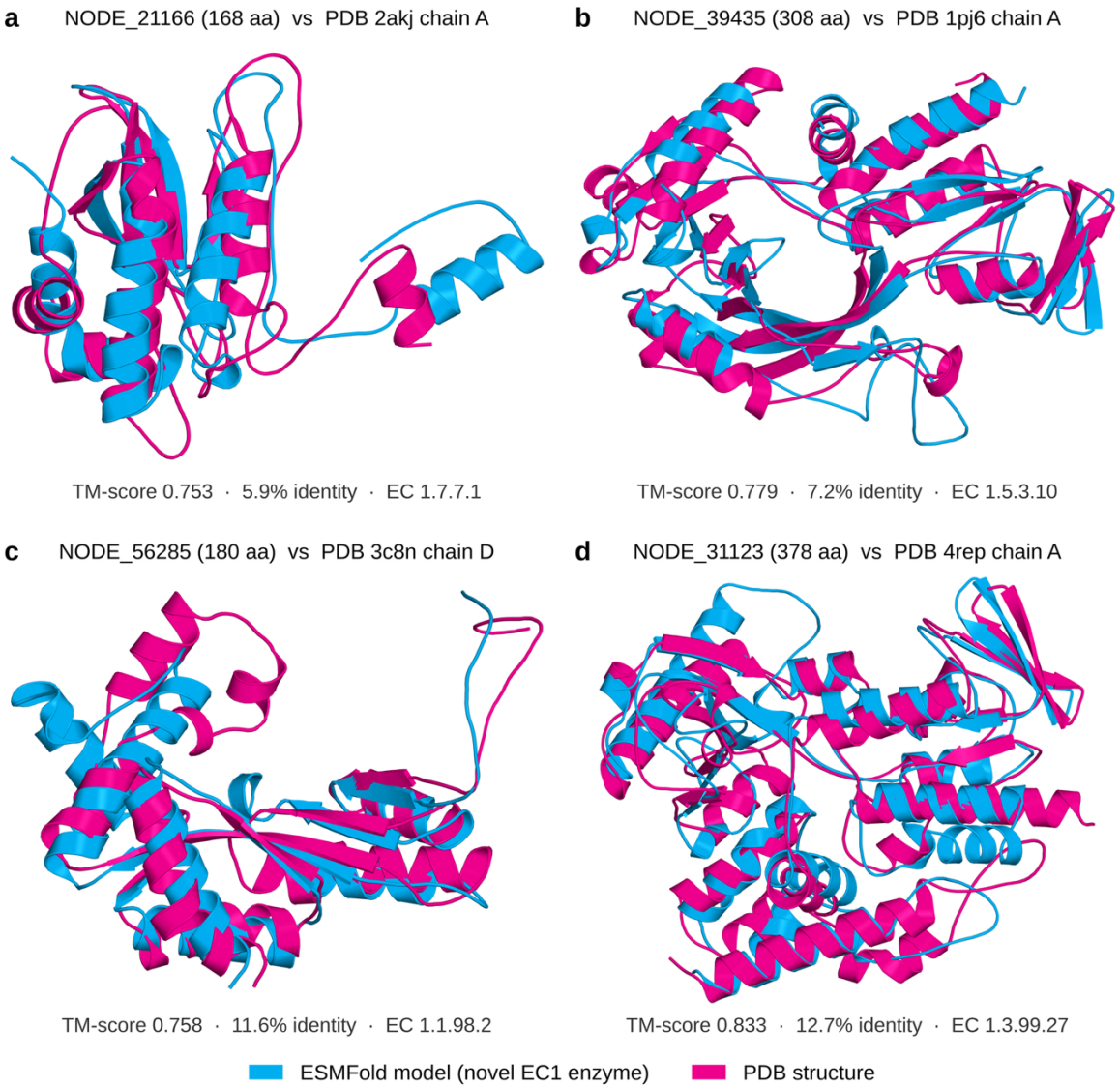


Figure 3: REFinder identified novel, homology-invisible enzymes show structural similarity to characterized oxidoreductases. ESMFold models (cyan) of four novel enzymes recovered by the seeded assembly of saliva metagenomes, superposed on their best EC-annotated PDB chains (magenta). Each protein was assigned to EC1 by REBEAN, has a reliable embedding (RNS = 0), has no Swiss-Prot hit at 30% identity or above, and has no Pfam-A match (neither at the gathering thresholds nor at E ≤ 10^{-5}); (a) NODE_21166 (168 aa; SRR5892215) against spinach ferredoxin–nitrite reductase (PDB 2akj chain A, 535 aa): TM-score 0.753, 5.9% sequence identity, RMSD 4.43 Å over 152 residues, EC 1.7.7.1. (b) NODE_39435 (308 aa; SRR5892216) against Arthrobacter globiformis N,N-dimethylglycine oxidase (PDB 1pj6 chain A, 828 aa): TM-score 0.779, 7.2% identity, RMSD 4.49 Å over 272 residues, EC 1.5.3.10. (c) NODE_56285 (180 aa; SRR5892216) against Mycobacterium tuberculosis F420-dependent glucose-6-phosphate dehydrogenase FGD1 (PDB 3c8n chain D, 330 aa): TM-score 0.758, 11.6% identity, RMSD 3.84

Å over 160 residues, EC 1.1.98.2. (d) NODE_31123 (378 aa; SRR5892216) against Nonlabens dokdonensis γ-carotene desaturase (PDB 4rep chain A, 489 aa): TM-score 0.833, 12.7% identity, RMSD 3.40 Å over 344 residues, EC 1.3.99.27. Models are confident (mean pLDDT 87–92). TM-scores and sequence identities are from Foldseek (TM-align mode) over the structural alignment.

Figure 3 shows four such proteins matched and aligned against experimentally studied PDB structures (Burley et al., 2025; Schrödinger LLC, 2024; van Kempen et al., 2024) with known oxidoreductase EC annotation (Dana et al., 2019) and no detectable matches in Swiss-Prot and Pfam-A. NODE_21166 (168 aa) matched spinach ferredoxin–nitrite reductase (PDB 2AKJ_A; TM-score 0.753, 5.9% sequence identity; EC 1.7.7.1). NODE_39435 (308 aa) matched the dimethylglycine oxidase of Arthrobacter globiformis (1PJ6_A; TM-score 0.779, 7.2% identity; EC 1.5.3.10). NODE_56285 (180 aa) matched the F(420)-dependent glucose-6-phosphate dehydrogenase (FGD1) of Mycobacterium tuberculosis (3C8N_D; TM-score 0.758, 11.6% identity; EC 1.1.98.2). NODE_31123 (378 aa) matched the γ-carotene desaturase of Nonlabens dokdonensis (4REP_A; TM-score 0.833, 12.7% identity; EC 1.3.99.27).

At 6-13% sequence identity with the corresponding PDB sequences, these matches fall far below the range in which homology transfer is reliable, so these proteins would be deemed homology-invisible, while adopting folds of well-characterized enzymes. Note that the structural correspondence establishes fold rather than activity and does not by itself demonstrate catalysis, but it shows that a substantial share of the homology-invisible set of proteins is neither a result of assembly nor an artifact of enzyme classification. The most prominent value of REFinder is that it retains these proteins at a fraction of the assembly cost, not that it proves these proteins are enzymes.

3.4. *Assembling genes at low read-depth*

A class-focused assembly should be able to reconstruct genes so sparsely covered by sequencing that they are unlikely to emerge from the whole community where they compete against the entire read pool. To test this hypothesis, we counted the trimmed sample reads that mapped to each assembled gene and compared the coverage of arm-unique enzymes. Seed-only enzymes were, on average, of lower read depth than assembly-only enzymes in all seven EC classes across both datasets; median coverage was 17-27 reads per enzyme for seed-only vs. 24-51 for assembly-only in the saliva set, and 22-49 vs. 30-95, respectively, in the murine gut, a median ratio of seed-to-full assembly proteins of 0.52-0.73 (**Figure S4, S5**). Seeding therefore uniquely recovers genes that are drawn from the sparsely covered assembly tail, i.e. the regime in which enzyme mining is hardest. We hypothesize that restricting assembly to class-routed reads may simplify the de Bruijn graph used in assembly by removing unrelated sequence and improving contiguity for the targeted class. This may also stem from low depth reads assembled into shorter contigs and incomplete genes (**Figure S5**). Though not established herein, these hypotheses are testable by controlled down-sampling and by direct measurement of graph complexity. In addition, we observed seeded assemblies were enriched with enzymatic reads. In the seeded assemblies, enzyme genes accounted

for a median of 9.2% (saliva) and 5.4% (murine gut) of the mapped reads when annotated by DIAMOND, and 33.1% and 41.1% when annotated by REBEAN. In the full assembly, the corresponding shares were 5.2% and 4.0% for DIAMOND-annotated enzyme genes (**Figure S5**) and 19.7% and 32.8% for REBEAN-annotated ones.

3.5. *Computational cost*

Assembly compute cost dominated both study arms. Per saliva sample, metaSPAdes accounted for 10.3 ± 5.6 of the 10.9 ± 5.7 CPU-hours of a full assembly; in the murine gut it accounted for 64.1 ± 13.9 of 65.5 ± 14.0 CPU-hours (**Table S5**). Because each seed arm assembled only its class-routed reads, their costs fell below the full arm in most classes: the saliva EC-*N* seed arms ranged from 1.7 ± 0.9 CPU-hours for EC7 (0.16 times the full arm) to 8.0 ± 4.6 CPU-hours for EC2 (0.73 times the full arm), for a maximum saving of 6.4-fold. The murine seed arm ranged from 18.5 ± 6.1 CPU-hours for EC7 (0.28 times full assembly) to 71.3 ± 16.7 CPU-hours for EC2 (1.09 times full assembly). Note that murine EC2, the class to which REBEAN routed the most reads, was the sole exception to cost savings – every other EC class in every sample was cheaper in seeded than in full assembly states. Consistent with prior targeted assemblers (Guo et al., 2019), restricting assembly to a single routed class reduces cost relative to whole-community assembly.

Note that recovering the complete enzyme catalogue is a different question than running seven separate seed arms, one per EC class. A single assembly of all enzymatic reads is expected to cost less than the sum across ECs or the full assembly, but this runtime was not evaluated here as it is not the scope for a seeded assembly framework. In addition, this accounting excluded the GPU usage for REBEAN enzyme annotation of saliva (6.5±3.2M reads per sample) and gut (70.1±13.3M reads) samples (0.13±0.06 h and 1.4±0.3 per sample, respectively; **Table S5)**.

4. Conclusions

This is, to our knowledge, the first implementation of an alignment-free, language model-guided, targeted, enzyme discovery pipeline. REFinder reconstructs known enzymes comparably to full assembly while recovering large numbers of sequence-novel enzymes that homology cannot reach, at lower assembly cost per targeted class.

The present system routes metagenomic reads at the top level of the EC hierarchy, which is coarse: a class such as EC7, which includes enzymes that transport specific substrates, is still an aggregate of many distinct molecular activities. Routing on finer EC sub-classes would further concentrate assembly, reducing cost and sharpening recovery of a targeted activity. While it is the natural next step for this work, we note that discovery of novel (non-EC-elucidated) enzymatic activities may be a direct product of coarse-grained classification presented here.

Three limitations bound these conclusions. First, here we evaluated two well-studied communities/datasets, so generalization to high-diversity soil and marine environments remains to be shown. Second, cost savings are class dependent. In fact, we found one outlier: for murine EC2, the class recruiting the most sample reads, seeding cost slightly more than full assembly. Most importantly, our compute accounting excludes the cost of running REBEAN over the read sets. We aim to optimize its speed in later versions of the pipeline.

5. Acknowledgments

Funding: This work was supported by the NSF (National Science Foundation) award #2310114.

6. Author contributions

Conceptualization: RP, YB; Development & Methodology: RP; Visualization: RP, YB; Supervision: YB; Writing and revisions: RP, YB;

7. Competing interests

The authors declare no competing interests.

8. Data and materials availability

The pipeline is available at https://bitbucket.org/bromberglab/NF-REFinder. The underlying standalone REBEAN package can be found at https://bitbucket.org/bromberglab/rebeanpkg.

References

Altschul, S. F., Gertz, E. M., Agarwala, R., Schaffer, A. A., & Yu, Y. K. (2009). PSI-BLAST pseudocounts and the minimum description length principle. *Nucleic Acids Res*, *37*(3), 815-824. https://doi.org/10.1093/nar/gkn981

Andrews, S. (2010). *FastQC: a quality control tool for high throughput sequence data*. In Babraham Bioinformatics, Babraham Institute. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Aptekmann, A. A., Buongiorno, J., Giovannelli, D., Glamoclija, M., Ferreiro, D. U., & Bromberg, Y. (2022). mebipred: identifying metal binding potential in protein sequence. *Bioinformatics*, *38*(14), 3532-3540. https://doi.org/10.1093/bioinformatics/btac358

Beghini, F., McIver, L. J., Blanco-Miguez, A., Dubois, L., Asnicar, F., Maharjan, S., Mailyan, A., Manghi, P., Scholz, M., Thomas, A. M., Valles-Colomer, M., Weingart, G., Zhang, Y., Zolfo, M., Huttenhower, C., Franzosa, E. A., & Segata, N. (2021). Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3. *Elife*, *10*, e65088. https://doi.org/10.7554/eLife.65088

Belstrøm, D., Constancias, F., Liu, Y., Yang, L., Drautz-Moses, D. I., Schuster, S. C., & Givskov, M. (2017). Metagenomic and metatranscriptomic analysis of saliva reveals disease-associated microbiota in patients with periodontitis and dental caries. *npj Biofilms and Microbiomes*, *3*, 23. https://doi.org/10.1038/s41522-017-0031-4

Borujeni, P. M., & Salavati, R. (2024). Functional domain annotation by structural similarity. *NAR Genom Bioinform*, *6*(1), lqae005. https://doi.org/10.1093/nargab/lqae005

Boyd, J. A., Woodcroft, B. J., & Tyson, G. W. (2018). GraftM: a tool for scalable, phylogenetically informed classification of genes within metagenomes. *Nucleic Acids Res*, *46*(10), e59. https://doi.org/10.1093/nar/gky174

Buchfink, B., Xie, C., & Huson, D. H. (2015). Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, *12*, 59-60. https://doi.org/10.1038/nmeth.3176

Burley, S. K., Bhatt, R., Bhikadiya, C., Bi, C., Biester, A., Biswas, P., Bittrich, S., Blaumann, S., Brown, R., Chao, H., Chithari, V. R., Craig, P. A., Crichlow, G. V., Duarte, J. M., Dutta, S., Feng, Z., Flatt, J. W., Ghosh, S., Goodsell, D. S., . . . Zardecki, C. (2025). Updated

resources for exploring experimentally-determined PDB structures and Computed Structure Models at the RCSB Protein Data Bank. *Nucleic Acids Research*, *53*(D1), D564-D574. https://doi.org/10.1093/nar/gkae1091

Bushnell, B. (2014). *BBMap: a fast, accurate, splice-aware aligner*. https://sourceforge.net/projects/bbmap/

Cabral, D. J., Penumutchu, S., Reinhart, E. M., Zhang, C., Korry, B. J., Wurster, J. I., Nilson, R., Guang, A., Sano, W. H., Rowan-Nash, A. D., Li, H., & Belenky, P. (2019). Microbial Metabolism Modulates Antibiotic Susceptibility within the Murine Gut Microbiome. *Cell Metab*, *30*(4), 800-823 e807. https://doi.org/10.1016/j.cmet.2019.08.020

Cantalapiedra, C. P., Hernandez-Plaza, A., Letunic, I., Bork, P., & Huerta-Cepas, J. (2021). eggNOG-mapper v2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. *Mol Biol Evol*, *38*(12), 5825-5829. https://doi.org/10.1093/molbev/msab293

Carr, R., & Borenstein, E. (2014). Comparative analysis of functional metagenomic annotation and the mappability of short reads. *PLoS One*, *9*, e105776. https://doi.org/10.1371/journal.pone.0105776

Chen, S. (2025). fastp 1.0: An ultra-fast all-round tool for FASTQ data quality control and preprocessing. *Imeta*, *4*(5), e70078. https://doi.org/10.1002/imt2.70078

Dana, J. M., Gutmanas, A., Tyagi, N., Qi, G., O'Donovan, C., Martin, M., & Velankar, S. (2019). SIFTS: updated Structure Integration with Function, Taxonomy and Sequences resource allows 40-fold increase in coverage of structure-based annotations for proteins. *Nucleic Acids Research*, *47*(D1), D482-D489. https://doi.org/10.1093/nar/gky1114

Di Tommaso, P., Chatzou, M., Floden, E. W., Barja, P. P., Palumbo, E., & Notredame, C. (2017). Nextflow enables reproducible computational workflows. *Nature Biotechnology*, *35*(4), 316-319. https://doi.org/10.1038/nbt.3820

Eddy, S. R. (2011). Accelerated Profile HMM Searches. *PLoS Comput Biol*, *7*(10), e1002195. https://doi.org/10.1371/journal.pcbi.1002195

Elnaggar, A., Heinzinger, M., Dallago, C., & Rost, B. (2022). ProtTrans: Toward Understanding the Language of Life Through Self-Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*, 7112-7127. https://doi.org/10.1109/TPAMI.2021.3095381

Ferrer, M., Martinez-Martinez, M., Bargiela, R., Streit, W. R., Golyshina, O. V., & Golyshin, P. N. (2016). Estimating the success of enzyme bioprospecting through metagenomics: current status and future trends. *Microb Biotechnol*, *9*(1), 22-34. https://doi.org/10.1111/1751-7915.12309

Guo, J., Quensen, J. F., Sun, Y., Wang, Q., Brown, C. T., Cole, J. R., & Tiedje, J. M. (2019). Review, Evaluation, and Directions for Gene-Targeted Assembly for Ecological Analyses of Metagenomes. *Front Genet*, *10*, 957. https://doi.org/10.3389/fgene.2019.00957

Hyatt, D., Chen, G. L., Locascio, P. F., Land, M. L., Larimer, F. W., & Hauser, L. J. (2010). Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics*, *11*, 119. https://doi.org/10.1186/1471-2105-11-119

Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M., & Tanabe, M. (2016). KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res*, *44*(D1), D457-462. https://doi.org/10.1093/nar/gkv1070

Li, D., Huang, Y., Leung, C. M., Luo, R., Ting, H. F., & Lam, T. W. (2017). MegaGTA: a sensitive and accurate metagenomic gene-targeted assembler using iterative de Bruijn graphs. *BMC Bioinformatics*, *18*(Suppl 12), 408. https://doi.org/10.1186/s12859-017-1825-3

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., Dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., & Rives, A. (2023). Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, *379*(6637), 1123-1130. https://doi.org/10.1126/science.ade2574

Lorenz, P., & Eck, J. (2005). Metagenomics and industrial applications. *Nat Rev Microbiol*, *3*(6), 510-516. https://doi.org/10.1038/nrmicro1161

Mahlich, Y., Steinegger, M., Rost, B., & Bromberg, Y. (2018). HFSP: high speed homology-driven function annotation of proteins. *Bioinformatics*, *34*(13), i304-i312. https://doi.org/10.1093/bioinformatics/bty262

Meyer, F., Paarmann, D., D'Souza, M., Olson, R., Glass, E. M., Kubal, M., Paczian, T., Rodriguez, A., Stevens, R., Wilke, A., Wilkening, J., & Edwards, R. A. (2008). The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes. *BMC Bioinformatics*, *9*, 386. https://doi.org/10.1186/1471-2105-9-386

Mistry, J., Chuguransky, S., Williams, L., & Bateman, A. (2021). Pfam: The protein families database in 2021. *Nucleic Acids Research*, *49*, D412-D419. https://doi.org/10.1093/nar/gkaa913

Nurk, S., Meleshko, D., Korobeynikov, A., & Pevzner, P. A. (2017). metaSPAdes: a new versatile metagenomic assembler. *Genome Research*, *27*, 824-834. https://doi.org/10.1101/gr.213959.116

Pell, J., Hintze, A., Canino-Koning, R., Howe, A., Tiedje, J. M., & Brown, C. T. (2012). Scaling metagenome sequence assembly with probabilistic de Bruijn graphs. *Proceedings of the National Academy of Sciences USA*, *109*, 13272-13277. https://doi.org/10.1073/pnas.1121464109

Prabakaran, R., & Bromberg, Y. (2025). Deciphering enzymatic potential in metagenomic reads through DNA language models. *Nucleic Acids Res*, *53*(16), gkaf836. https://doi.org/10.1093/nar/gkaf836

Prabakaran, R., & Bromberg, Y. (2026). Quantifying uncertainty in protein representations across models and tasks. *Nat Methods*, *23*(4), 796-804. https://doi.org/10.1038/s41592-026-03028-7

Quince, C., Walker, A. W., Simpson, J. T., Loman, N. J., & Segata, N. (2017). Shotgun metagenomics, from sampling to analysis. *Nat Biotechnol*, *35*(9), 833-844. https://doi.org/10.1038/nbt.3935

Rappe, M. S., & Giovannoni, S. J. (2003). The uncultured microbial majority. *Annual Review of Microbiology*, *57*, 369-394. https://doi.org/10.1146/annurev.micro.57.030502.090759

Robinson, S. L., Piel, J., & Sunagawa, S. (2021). A roadmap for metagenomic enzyme discovery. *Nat Prod Rep*, *38*(11), 1994-2023. https://doi.org/10.1039/d1np00006c

Rost, B. (2002). Enzyme function less conserved than anticipated. *J Mol Biol*, *318*(2), 595-608. https://doi.org/10.1016/S0022-2836(02)00016-5

Schrödinger LLC. (2024). *The PyMOL Molecular Graphics System, Version 3.0*. In Schrödinger, LLC. https://www.pymol.org

Steinegger, M., & Soding, J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, *35*, 1026-1028. https://doi.org/10.1038/nbt.3988

Tipton, K. F. (1994). Nomenclature Committee of the International Union of Biochemistry and Molecular Biology (NC-IUBMB). Enzyme nomenclature. Recommendations 1992. Supplement: corrections and additions. *Eur J Biochem*, *223*(1), 1-5. https://doi.org/10.1111/j.1432-1033.1994.tb18960.x

UniProt Consortium. (2025). UniProt: the Universal Protein Knowledgebase in 2025. *Nucleic Acids Res*, *53*(D1), D609-D617. https://doi.org/10.1093/nar/gkae1010

van Kempen, M., Kim, S. S., Tumescheit, C., Mirdita, M., Lee, J., Gilchrist, C. L. M., Soding, J., & Steinegger, M. (2024). Fast and accurate protein structure search with Foldseek. *Nat Biotechnol*, *42*(2), 243-246. https://doi.org/10.1038/s41587-023-01773-0

Vanni, C., Schechter, M. S., Acinas, S. G., Barberan, A., Buttigieg, P. L., Casamayor, E. O., Delmont, T. O., Duarte, C. M., Eren, A. M., Finn, R. D., Kottmann, R., Mitchell, A., Sanchez, P., Siren, K., Steinegger, M., Gloeckner, F. O., & Fernandez-Guerra, A. (2022). Unifying the known and unknown microbial coding sequence space. *Elife*, *11*, e67667. https://doi.org/10.7554/eLife.67667

Wang, Q., Fish, J. A., Gilman, M., Sun, Y., Brown, C. T., Tiedje, J. M., & Cole, J. R. (2015). Xander: employing a novel method for efficient gene-targeted metagenomic assembly. *Microbiome*, *3*, 32. https://doi.org/10.1186/s40168-015-0093-6

Wommack, K. E., Bhavsar, J., & Ravel, J. (2008). Metagenomics: read length matters. *Applied and Environmental Microbiology*, *74*, 1453-1463. https://doi.org/10.1128/AEM.02181-07

Zhang, Y., Sun, Y., & Cole, J. R. (2014). A scalable and accurate targeted gene assembly tool (SAT-Assembler) for next-generation sequencing data. *PLoS Comput Biol*, *10*(8), e1003737. https://doi.org/10.1371/journal.pcbi.1003737

Zhu, C., Miller, M., Marpaka, S., Vaysberg, P., Ruhlemann, M. C., Wu, G., Heinsen, F. A., Tempel, M., Zhao, L., Lieb, W., Franke, A., & Bromberg, Y. (2018). Functional sequencing read annotation for high precision microbiome analysis. *Nucleic Acids Res*, *46*(4), e23. https://doi.org/10.1093/nar/gkx1209

RxnPred: A Generative Framework for Enzymatic Reaction Prediction

Srikrishnan Balamuhunthan[1], Saketha Nath Jagarlapudi[2] and Karthik Raman[1*]

[1] *Department of Data Science and Artificial Intelligence, Indian Institute of Technology Madras, Chennai, India*

[2] *Department of Computer Science and Engineering, Indian Institute of Technology Hyderabad, Hyderabad, India*

[*] *Corresponding author: kraman@dsai.iitm.ac.in*

Enzyme function annotation remains a challenging problem, even with many advancements in machine learning and deep learning. Most computational tools predict Enzyme Commission numbers, which are occasionally reassigned, or retrieve reactions from a candidate pool. We introduce RxnPred, which instead generates reaction SMILES from protein sequence and structure inputs, using an encoder-decoder transformer model trained on protein-reaction pairs. We evaluate RxnPred on three data splits of increasing difficulty in terms of sequence identity, and on three external test sets, comparing against a homology baseline, a contrastive learning reaction screening tool, and four EC prediction tools. On reactions seen during training, RxnPred is competitive with these tools and achieves higher recall under beam search, though at lower precision; the homology baseline nevertheless remains the strongest method. Performance drops when test sequences become more dissimilar from training data, and on unseen reactions, the model does not recover the correct answer, whereas a retrieval framework can rank such reactions because they are supplied as input. Its predictions for these entries are nonetheless closer to the ground truth by fingerprint similarity than those of the homology baseline and an EC prediction tool. The model does generate reaction SMILES absent from its training data, a small fraction of which map to existing reactions, but assigns them to the wrong enzymes. We identify metabolite-level tokenization and protein-reaction alignment, rather than reaction generation, as the main bottleneck, and granular tokenization, chemical validity constraints, and strong alignment as directions for improvement.

Keywords: Enzyme function prediction; reaction SMILES generation; reaction prediction

1. Introduction

Protein function prediction is an active area of research due to the task's challenging nature. As the available corpus of genome sequences grows ever faster, so does the number of proteins, but the functions of many of them are still unknown. Protein functions are commonly defined using Gene Ontology terms with three subcategories: Molecular Function, Cellular Component, and Biological Process. For enzymes, the common way to describe their function is through Enzyme Commission (EC) numbers, which are expert-defined four-digit codes, with each digit having a specific meaning. Apart from experimentally determining what reaction an enzyme catalyses, computational methods have been developed to expedite the process of

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

functional characterisation of new proteins.[1] Traditionally, EC numbers are assigned either by homology search using tools like blastp[2] or by using Hidden Markov Model profiles.[3] Recently, deep learning tools have been developed for function prediction. For example, CLEAN[4] is a contrastive learning tool for predicting EC numbers from protein sequences. CLEAN-Contact[5] utilises protein structure in addition to sequence for EC prediction. HIT-EC[6] is a transformer-based EC tool that predicts EC number hierarchically. While the EC numbers provide a standard nomenclature, they are occasionally reassigned or deprecated, making it difficult to have a consistent representation of the enzyme function.

A more specific way to describe enzyme function is through the exact biochemical reaction(s) it catalyses. Unlike EC numbers, the exact reactions are given by various identifiers across different databases. However, representing the metabolites of the reaction as SMILES (Simplified Molecular Input Line Entry System) strings, and thus the reaction as a reaction SMILES string, will yield a consistent representation. While most existing tools focus on the EC number, very few recent methods have shifted their focus to exact reactions. CLIPZyme[7] is an enzyme screening tool, where enzyme structures and reactions are encoded using Graph Neural Networks. Horizyn-1[8] encodes reactions using reaction fingerprints and encodes proteins using protein language models for reaction or enzyme retrieval. ReactZyme[9] benchmarks different architectures on enzyme-reaction prediction tasks, across diverse challenging data splits. While these reaction-centric tools rank reactions from a given pool, a generative framework generates them from scratch, which none of these tools does. This opens the label space, since the model is not restricted to reactions present in the training, at the risk of generating chemically invalid ones. Improving the accuracy of enzyme function annotation has the potential to accelerate genome-scale metabolic model reconstruction and annotation of metagenomes.

Pretrained and fine-tuned generative models exist for organic chemistry reactions, which are used for forward reaction prediction or retrosynthesis.[10–12] However, biochemical reactions differ from those reactions because they depend on the input enzyme. The amount of data available for organic chemistry reactions is also disproportionately larger than that available for enzymatic reactions. For example, Molecular Transformer was trained on hundreds of thousands of reactions, while the total number of reactions present in Rhea curated database is 18,558.[10,13] These factors likely explain why, to our knowledge, no existing method addresses this problem as a generative task.

In this work, we introduce RxnPred, a generative framework that predicts reaction SMILES directly from protein sequences and structures (Figure 1A), and evaluate it across three data splits of increasing sequence dissimilarity and three external test sets against a homology baseline, a retrieval based tool, and four EC prediction tools, characterizing where the generative framing helps and where it does not. A generative framework, especially with an open SMILES vocabulary, is not restricted to generating reactions only from the training data. However, we used a full metabolite vocabulary, which shortens the tokenized sequences considerably but is closed: the model cannot predict any metabolite out of the vocabulary. By simulating a test setting in which a model is prompted to predict the functions of enzymes whose associated reactions it has not seen, we assess its ability to generalise in biochemistry. RxnPred

is competitive with a homology baseline on reactions seen during training and outperforms a contrastive retrieval tool, with beam search raising recall at the cost of precision, though the homology baseline remains the strongest method in this setting. It does generate reaction SMILES absent from its training set, but assigns them to the wrong enzymes, so the generative advantage remains unrealized.

2. Methods

2.1. *Training data and preprocessing*

RxnPred was developed using protein and reaction data from SwissProt, a curated subset of UniProt[14] and Rhea.[13] We downloaded the protein sequences, along with their EC numbers and Rhea annotations, from SwissProt (as on 15th Sep 2025). We removed protein sequences shorter than 30 amino acids or longer than 1024 amino acids, retaining only the entries that have at least one Rhea association and not more than 10. We considered only the forward (left-to-right) versions of all the reactions and we filtered out reactions that lacked a valid SMILES string from Rhea. Each metabolite's SMILES string was canonicalised using RDKit.[15] We evaluated our model on three data splits: one random split and two sequence-similarity-based splits. For the sequence similarity-based splits, we used MMseqs2[16] to cluster sequences at two sequence identity thresholds: 30% and 70%. A query coverage of 0.5 was considered in all clustering experiments. Train-validation-test splits were made in such a way that sequences belonging to the same cluster are present within the same split, and there is no leakage across the splits, with an approximate ratio of 0.8/0.1/0.1.

Since our method's main focus is on monofunctional enzymes, validation and test splits contain only enzymes with a single reaction association. However, multifunctional enzymes increase reaction coverage and were therefore included in the training set. Our model being a sequence-to-sequence transformer, each training item is expected to be a pair of an enzyme and a reaction. For these multifunctional enzymes, the sequences were duplicated and differentially masked to create augmented sequences, following CLEAN[4]. We similarly augmented reactions associated with fewer than 10 enzymes. Based on metabolite frequencies in the training split, metabolites in a reaction were ordered from least to most frequent on both the reactant and product sides. We obtained the structures of all the proteins from the AlphaFold structure database v6,[17] using LocalColabFold(`https://github.com/YoshitakaMo/localcolabfold`) for structures absent from AlphaFold database (Supplementary material Section S1.1). The resulting training sets are comparable in size across splits: 241,820 entries (132,565 proteins, 9,643 reactions) for the random split, 238,230 (132,578 proteins, 8,994 reactions) for MMseqs30, and 235,992 (132,580 proteins, 8,827 reactions) for MMseqs70. Data, code, and all supplementary materials are available at `https://github.com/srikrishnan-b/ReactionPred`.

2.2. *Protein and reaction representation*

We used the protein language model ESM2[18] (`esm2_t12_35M_UR50D`) to generate embeddings for the proteins. The embeddings are of shape $L \times d$, where L is the length of the sequence, and d is the embedding dimension, which is 480 for this model. For proteins longer than 512 residues, the first 512 residues were taken. All embeddings were precomputed and stored on

the disk in half-precision (FP16) format. From the protein structures, we used ESM-IF1[19] to generate embeddings of shape $L \times d$, where L is the length of the sequence, and d is the embedding dimension, which is 512 in this case. The final protein representation is obtained by concatenating the structure embedding with the sequence embedding along the embedding dimension, yielding a representation of shape $L \times 992$. Reactions are represented as reaction SMILES, where the metabolites are separated by a "." token and the reactants and products are separated by a ">>" token. ChemBERTa-10M-MLM[20] was used to obtain 384-dimensional embeddings for all the metabolites from their SMILES strings.

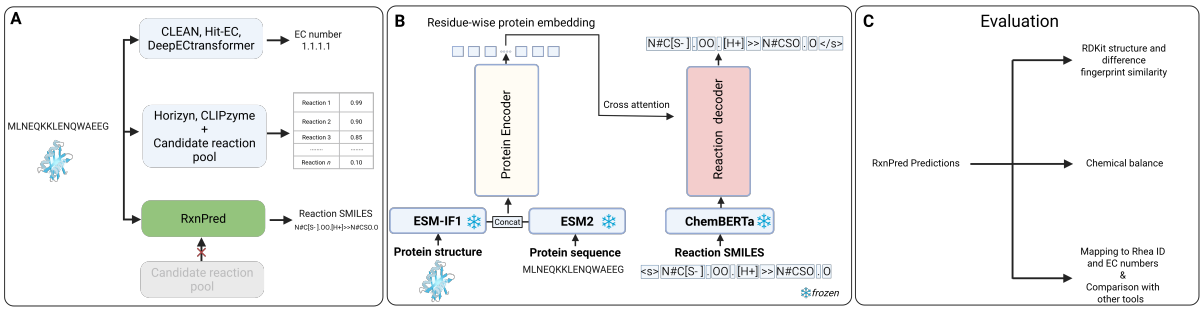


Fig. 1. **RxnPred Overview:** A. Existing tools predict a fixed EC label or rank a supplied candidate pool; RxnPred directly generates reaction SMILES. B. Architecture: fused ESM-2 and ESM-IF1 embeddings feed a protein encoder; the reaction decoder cross-attends and generates metabolite tokens autoregressively. C. Predictions are evaluated by fingerprint similarity, chemical balance, and mapping to Rhea and EC. (Created in BioRender. Raman, K. (2026) https://BioRender.com/rqkj8j8)

2.3. *RxnPred implementation*

RxnPred is a sequence-to-sequence transformer model adopting the BART[21] skeleton, which consists of a bidirectional encoder and an autoregressive decoder. We adopted the underlying encoder and decoder architectures, but the model is initialized randomly and trained from scratch. RxnPred has two transformer modules: a protein encoder and a reaction decoder (Figure 1B). We chose this architecture because the task involves mapping a variable-length input to a variable-length output, which is the standard setting for an encoder-decoder with cross attention. The protein embeddings are processed by the encoder, which consists of self-attention layers, and its last hidden state $H \in \mathbb{R}^{L \times d}$ is used by the decoder. The reaction decoder also has self-attention and cross-attention layers, with the reaction SMILES string as the input during training, and H used for cross-attention. For the decoder, in the input, a `<START>` token acts as the first token and the original tokens are shifted to the right by one position, while the output to be predicted ends with a `<STOP>` token. The reaction SMILES strings are tokenized into individual metabolites and special reaction tokens, making each individual metabolite a token. We chose this tokenization scheme to reduce the complexity arising from the longer input sequences produced by atom-wise tokenizers. For the reaction decoder, we constructed the vocabulary from all the metabolites available in our dataset. The

embedding layer for this was initialized from the precomputed ChemBERTa embeddings and remains frozen throughout the training. The non-metabolite special token (eg., "." and ">>") embeddings are randomly initialized and updated throughout the training.

The model is trained with teacher forcing using cross-entropy loss. The final model uses four encoder layers and six reaction decoder layers, with four attention heads and $d_{\text{model}} = 320$. The feed-forward dimension is 4096 in the encoder and 2048 in the decoder. Models were trained up to 25 epochs with early stopping on validation loss, using AdamW optimizer with a peak learning rate of 4.4×10^{-4}, cosine annealing, and a linear warmup over the first 10% of steps. We used a per-device batch size of 64 with two gradient accumulation steps, giving an effective batch size of 128. Training was implemented with the HuggingFace Transformers `Trainer`[22] (v4.57.1) on PyTorch[23] (v2.13.0) and Python 3.10.0, under Ubuntu 22.04.5 LTS. Experiments were run on NVIDIA RTX PRO 2000 Blackwell and NVIDIA A100 GPUs. Training took approximately 10.4 hours for the longest run on NVIDIA RTX PRO 2000 Blackwell GPU.

We evaluated two modes of inference: greedy decoding and beam search. In beam search, instead of a single output, k (beam width) best partial outputs are maintained at each decoding step, ranked by cumulative log probability.[24] We used a beam width of five for the internal test splits and ten for the external test sets. All k completed beams are retained, ranked by final log probability without length normalisation. This rank is the prediction rank in all top-k evaluations. During inference, the protein sequence and structure embeddings are given as input to the encoder, and the decoder is given a `<START>` token.

2.4. *Evaluation*

For evaluation, we use the internal test set of each split. In addition, we also evaluated the models on three external test sets: Price-149, Halogenase, and New-477. Price is a set of 149 proteins curated by ProteInfer,[25] from a previous work,[26] where they experimentally verified the functions of poorly annotated bacterial proteins, and is used as a benchmark set commonly.[4,5,27] Halogenase is a set of 37 halogenase enzymes, whose functions were assigned experimentally or through expert curation in CLEAN.[4] We constructed a new test set of 477 proteins released after our model's training cutoff date or excluded from our training data, providing a real-world scenario for evaluation. This set contains sequences released on SwissProt from 01/01/2025 to 27/03/2026. We made sure that none of the entries from these internal and external tests was used in the training process. In the external sets, only the entries with a Rhea annotation and that are monofunctional are considered, leaving Halogenase with 30 entries covering 12 Rhea reactions, Price with 110 entries covering 40 reactions, and New with 288 entries covering 194 reactions. We evaluate the model on two tasks: reaction prediction performance primarily, and EC number prediction performance as a proxy for reaction prediction.

The predicted reaction sequence is compared with the true reaction sequence using structure fingerprint Tanimoto similarity (RDKit CreateStructuralFingerprintForReaction) and difference reaction fingerprint Tanimoto similarity (RDKit CreateDifferenceFingerprintForReaction) (Figure 1C). Structure fingerprint encodes substructures present in reactants and products, while the difference fingerprint encodes the changes between them. Tanimoto simi-

larity is defined in Supplementary material Section S1.2. We check what fraction of predictions have similarity equal to or above a given threshold. We also assess whether the predicted reaction is mass- and charge-balanced. Further, all predictions are mapped to any of the existing Rhea IDs and from each Rhea ID to EC numbers. For comparison with other tools, we consider this as multi-label prediction and compute Precision, Recall and F1 scores for both Rhea and EC labels. To compare with a reaction screening tool, we calculate the top-k hit rate at different k values, which computes the fraction of predictions for which at least one of the ground truth reactions is predicted within the top-k ranks. In our model predictions, beams are considered as ranks. Multilabel EC and Rhea metrics are computed per label across the test set and averaged with weighting by label support. Results are reported separately for each internal and external test set.

For comparison with other tools, we used their released versions, except for CLEAN, which we also retrained on our datasets. The full details about the implementation can be found in the Supplementary material Section S1.4. We also performed experiments with a variant in which the structure was removed from the input, and another in which an additional EC decoder was used to predict EC up to 3 digits (Supplementary material Section S1.3).

3. Results

3.1. *Reaction generation performance of RxnPred*

RxnPred was trained and evaluated on three data splits: random, MMseqs30, and MMseqs70. In MMseqs30, no test sequence shares more than 30% identity with any training sequence at 0.5 query coverage, and in MMseqs70, the corresponding threshold is 70%. First, we evaluated the reaction prediction performance of our model on all the test splits. We compared the reaction SMILES generated by our model, obtained via either greedy decoding or beam search, with the ground truth using structure- and difference-fingerprint similarities. A beam width of 5 was used for the test sets, and the maximum similarity score for each entry was taken.

Figure 2 shows the fraction of predictions with similarity equal to or more than a given threshold for six thresholds from 1.0 to 0.5. With beam search, structure fingerprint exact matches fall from 93% on the random split to 83% on MMseqs70 and 56% on MMseqs30; difference fingerprint exact matches were 90% on the random split. Average similarities follow the same ordering (Figure 2).

In general, the structure fingerprint similarity scores are higher than the difference fingerprint scores, indicating that the model predicts similar metabolites even when the captured reaction transformation is incorrect. The reduction in exact matches and average similarity scores in MMseqs splits indicates that the model struggles to maintain accuracy as test set entries become more dissimilar to the training distribution. We assessed how the chemical balance of predicted reactions varies across beams and found that the first-ranked beam was balanced in 99% of predictions in the random test split (Supplementary material Section S2.1). The proportion of balanced predictions decreases as the beam rank increases, pointing towards the need for inference time steering to ensure chemical balance.

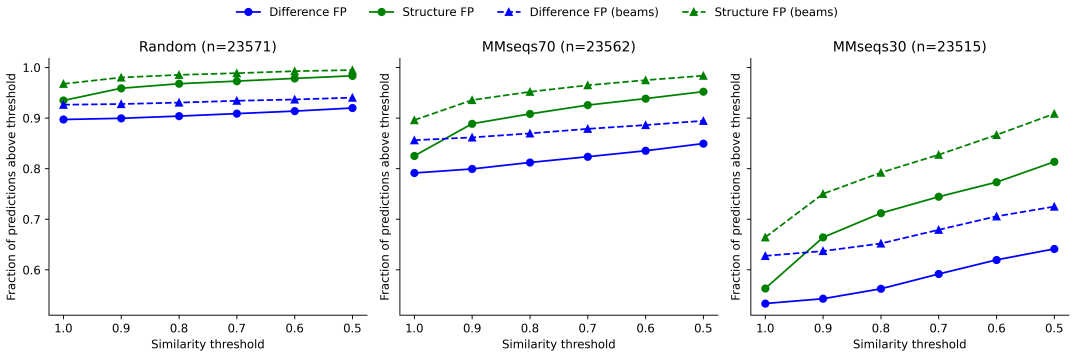


Fig. 2. Cumulative fraction of predictions from greedy decoding (solid lines) and beam search with width 5 (dashed lines) whose difference or structure fingerprint (FP) similarity to the ground truth is at or above the threshold on x-axis, running from 1.0 (exact match) to 0.5; blue – difference FP; green – structure FP. Panels: random (n=23,571), MMseqs70 (n=23,562) and MMseqs30 (n=23,515).

3.2. *Comparison with a homology baseline*

We investigated how our model performs in predicting the reaction compared with DIAMOND[28](v2.1.11) blastp (hereafter blastp), a homology baseline, on the internal test splits and the external test sets. Using blastp, the test set entries were queried against the training data as the target database, and either the top hit or the top-10 hits (top-5 for the random, MMseqs70, and MMseqs30 splits) were considered. The Rhea annotation of the hits is transferred to the test queries. Using our model, we predicted reactions for the test sets using greedy decoding and beam search with a width of 10 (beam width of 5 for the random, MMseqs70, and MMseqs30 splits). Then we computed multilabel classification metrics, precision, recall and F1 scores in the weighted average setting.

Table 1 shows the scores achieved by blastp and RxnPred across all test sets at the top-1 and top-k, where k is 5 for internal test splits and 10 for external test sets. When only the top hit is considered, RxnPred has a higher precision than blastp on Price and leads on all three metrics on the random split, while blastp lies ahead in the other test sets. The recall is also better for blastp across all test sets except the random test at the top-1 level. With beam search, our model's recall increases significantly as we consider more retrieved reactions. With top-k hits, though blastp's recall increases in the external test sets, it hardly increases in the internal test sets. The reason is that the training set contains many sequences similar to the queries and that share the same annotation, so they are retrieved as top hits. For the same reason, precision is not affected by using top-k hits with blastp. Conversely, beam search generates k candidate reactions, most of which are not the ground truth. On the internal splits this raises recall but sharply reduces precision. On the external sets, precision instead increases, because these sets contain far fewer distinct ground-truth reactions and the metric is weighted by label support, so false positives on rare labels contribute little to the average. In general, our model performs poorly on the Halogenase set but better on the Price dataset. Given that generation is a harder task than retrieving similar sequences from training data, matching or exceeding blastp's recall on five of six test sets at top-k is a reasonable outcome, though it comes at lower precision on the internal splits.

Table 1. Reaction prediction as multilabel classification: RxnPred (RxP) compared with blastp (Dmd), using weighted precision, recall, and F1. *Top-5 for internal test sets, Top-10 for external test sets.

	Top-1						Top-5/10*					
	Precision		Recall		F1		Precision		Recall		F1	
Test set	Dmd	RxP	Dmd	RxP	Dmd	RxP	Dmd	RxP	Dmd	RxP	Dmd	RxP
Price	0.52	**0.57**	**0.34**	0.32	**0.38**	**0.38**	0.62	**0.75**	0.46	**0.61**	0.49	**0.64**
Halogenase	**0.42**	0.13	**0.20**	0.13	**0.24**	0.12	**0.48**	0.45	**0.43**	0.33	**0.45**	0.30
New	**0.55**	0.46	**0.51**	0.42	**0.52**	0.43	**0.55**	0.48	0.54	**0.57**	**0.54**	0.50
Random split	0.91	**0.93**	0.90	**0.93**	0.90	**0.93**	**0.90**	0.67	0.90	**0.97**	**0.89**	0.76
MMseqs70	**0.86**	0.84	**0.84**	0.82	**0.85**	0.82	**0.84**	0.62	0.85	**0.89**	**0.84**	0.70
MMseqs30	**0.73**	0.66	**0.65**	0.54	**0.67**	0.55	**0.73**	0.57	**0.65**	**0.65**	**0.67**	0.56

3.3. *Comparison with a retrieval-based framework*

As an alternative to traditional homology-based methods, the problem of reaction assignment has recently been approached as a retrieval task, in which a protein is screened against a set of reactions, and the reactions are ranked, or vice versa. Horizyn-1[8] is a recent work that uses a dual encoder contrastive learning framework to screen enzymes for a given reaction, utilising cosine similarity scores between reaction and enzyme representations. Since this computation is symmetric, we used this framework to evaluate the model's ability to predict reactions for given enzymes. We used both released Horizyn-1 checkpoints, the development checkpoint used in the paper and the inference checkpoint trained on a larger dataset (Supplementary material Section S1.4).

We also report the top-k hit rate obtained using blastp. We used beam search of width 10 to generate results for these datasets from our model. Hence, we considered the Top-k hit rate for k values of 1, 3, 5, 7, and 10. In the combined dataset of size 657 (including multifunctional enzymes, excluding reactions with no SMILES), RxnPred matches or exceeds both Horizyn-1 checkpoints at every k (Figure 3, left).

Interestingly, the Horizyn-1 inference checkpoint was trained on a larger dataset, and we do not know which proteins and reactions were included in it or whether they are present in the test sets. Since the number of unseen reactions for Horizyn-1 and our model differs, for a fair comparison, we evaluated only top-k hit rates on a subset of 426 entries whose reactions were commonly seen by both models during training (Figure 3, right). In this, both blastp and RxnPred outperformed Horizyn-1 across all k levels. Although blastp leads at top-1 and top-3, RxnPred matches or exceeds it from top-5 onward, in both the full set and the subset of commonly seen reactions (Figure 3, left and right). Horizyn-1, however, performed better on a subset of common, unseen reactions for both models, in which it was able to rank unseen reactions in the top 10 at a rate of 0.15, while both blastp and RxnPred failed to predict any unseen reaction. This is possible because of the nature of the screening procedure, in which all possible reactions are presented to the model during inference for ranking, even if they were unseen during training. Blastp does not inherently predict unseen labels, as it just

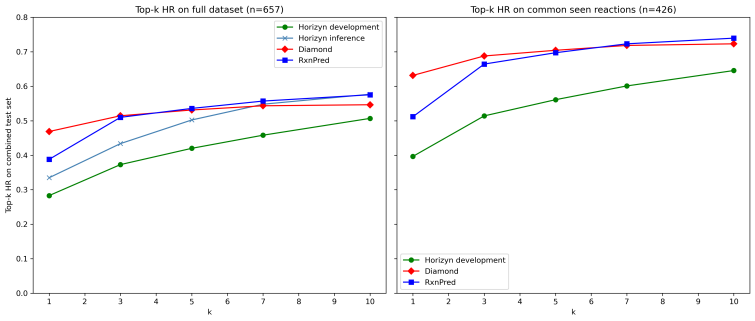


Fig. 3. Top-k hit rates for RxnPred compared with Horizyn-1 and blastp for the combined external test sets ($n = 657$) and the subset of $n = 426$ entries whose reactions were seen during training by both models.

picks candidates from the training data. RxnPred, on the other hand, is prompted to generate reactions for a protein from scratch. Although the generative framework can generate new reactions, our current model could not produce the correct reactions for these unseen entries. The model could, in fact, generate new reactions but for the wrong proteins, as explained in Section 3.5. The generative capabilities can be further improved by experimenting with different tokenization strategies that are more granular, as the notion of unseen metabolites vanishes in those cases, which is part of our future work.

3.4. *Comparison with EC prediction tools*

Next, we evaluated how our model compares to existing EC number prediction tools. For this, we mapped the predicted reaction SMILES to a Rhea identifier, then to an EC number associated with that Rhea ID. We use this comparison as a proxy for reaction prediction and compare it with existing EC tools. The tools used for comparison were DeepECtransformer,[29] CLEAN, CLEAN-Contact, HIT-EC, and blastp. The original released models were used for inference. We also checked the results after retraining CLEAN on our splits. For our model and blastp, results from two versions are reported: greedy and beam search for RxnPred, top-1 and top-10 for blastp. For the released models, we evaluated only on the external test sets, as our test set split might include proteins used in their training. We used our own splits to evaluate blastp and retrained CLEAN. The results for the external sets are presented in Table 2. On Price, as with the Rhea multilabel metrics, RxnPred with beam search showed the best recall and F1 scores. In the Halogenase dataset, CLEAN showed the best precision, recall, and F1 scores.

In the New dataset, recall was highest for RxnPred, while blastp showed higher precision and F1. The pattern of beam search increasing recall continues here as well, suggesting that RxnPred could predict the correct reactions given the chance to generate more candidates, even when its most confident prediction is incorrect. While CLEAN's released model performs well on the Halogenase dataset, its performance declined sharply when retrained on our dataset, suggesting the gap is attributable to training set composition rather than architecture. On the internal splits, blastp and retrained CLEAN lead on precision, while RxnPred with beam search remains competitive on recall (Supplementary material Section S2.3).

Table 2. Multilabel EC label metrics Precision, Recall, and F1 are computed from different models on the three external tests, and compared with RxnPred. Best result is written in bold.

	Price			Halogenase			New		
Method	P	R	F1	P	R	F1	P	R	F1
blastp (top-1)	0.42	0.28	0.31	0.37	0.18	0.21	0.63	0.61	0.62
blastp (top-10)	0.56	0.47	0.47	0.43	0.38	0.40	**0.64**	0.65	**0.64**
DeepECtransformer	0.53	0.31	0.36	0.13	0.15	0.14	0.52	0.47	0.49
CLEAN	0.61	0.49	0.52	**0.76**	**0.71**	**0.72**	0.56	0.54	0.55
CLEAN-Contact	**0.63**	0.53	0.54	0.00	0.00	0.00	0.58	0.56	0.56
HIT-EC	0.41	0.23	0.28	0.41	0.06	0.10	0.52	0.46	0.48
CLEAN (retrained)	0.40	0.33	0.34	0.52	0.41	0.43	0.62	0.59	0.60
RxnPred (greedy)	0.46	0.27	0.31	0.12	0.12	0.11	0.49	0.52	0.49
RxnPred (beam)	0.60	**0.57**	**0.55**	0.40	0.29	0.27	0.49	**0.69**	0.54

3.5. *Generalization to unseen reactions*

We wanted to assess the generalization ability of our model on unseen reactions. We investigated two aspects: 1) Can the model correctly predict an unseen reaction? 2) Can the model predict any unseen reaction at all, whether it is correct or not, being irrelevant. For the second question, the answer is no by design for CLEAN and blastp, as they cannot generate a new reaction that the model has not been trained on or is not present in the training data. To answer the first question, we evaluated RxnPred, blastp, and CLEAN on random test split entries with unseen reactions. We computed the structure and difference fingerprint similarity between the predicted and actual reactions. For CLEAN, we mapped the EC number to all possible reactions and selected the one that showed the maximum similarity. For RxnPred, we used beam search with a width of 5 on our test set and selected the beam with the highest score. For blastp, top-5 hits were considered. For the 226 entries with unseen reactions, RxnPred's predictions were closer to the ground truth than those of blastp and CLEAN on both fingerprints (Figure 4).

Next, we checked whether the model had generated any new reaction that was completely unseen during training, irrespective of whether it was the correct reaction for the given protein. This ability is unique to a generative framework, and we wanted to verify whether the model has learned the general chemistry and can generate new reactions. Using all five beams per test entry, the model generated 17,192, 17,516 and 12,779 distinct reactions for the random, MMseqs70 and MMseqs30 splits, of which 12,200, 12,333 and 9,247 were absent from the training data. We then checked how many of them are actual, valid reactions by mapping them to a Rhea identifier (Table 3). Only 68, 57 and 47 matched an actual Rhea reaction, and under 10% of the novel SMILES were chemically balanced. Therefore, many of the predictions are chemically invalid reactions that the model hallucinated. This also suggests that the model is not merely regurgitating reactions from the training set. In future work, chemistry-guided inference is needed to make sure the model generates only valid reactions. This could also be approached by incorporating validity constraints during training.

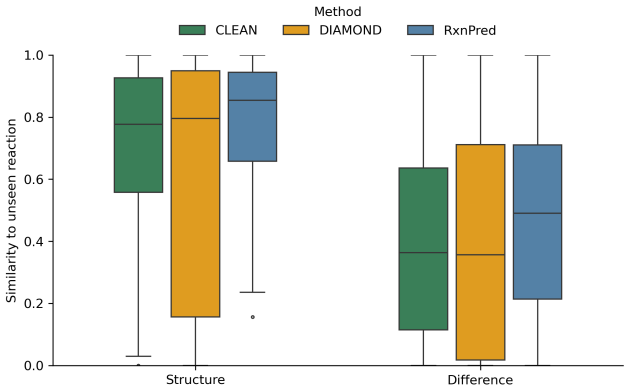


Fig. 4. Similarity distributions of predictions from RxnPred, blastp and CLEAN for 226 entries in the random test split whose reactions were not present in the training set, shown for structure fingerprint (left) and difference fingerprint (right) similarity.

Furthermore, we computed the difference fingerprint similarity between the novel SMILES corresponding to actual Rhea reactions and all SMILES in the training data to assess how different they are from the training distribution. The actual Rhea reactions generated by the model had a mean fingerprint similarity of 0.68 in the random split, 0.74 in the MMseqs70 split and 0.61 in the MMseqs30 split. We also found that one newly generated reaction was correct in each of the MMseqs70 and MMseqs30 splits (Rhea 33023 and 18369); both are similar to reactions present in the training data (Supplementary material Section S2.5). Most of the newly generated reactions were either transport reactions or seen transformations with different metabolites.

Table 3. "Novel SMILES" counts distinct predicted reaction SMILES not present in training, across all beams; the total number of distinct reactions predicted per split is given in the text. "Actual Rhea reactions" counts those matching an actual Rhea entry; "Balanced" gives the count and fraction of novel predictions that are chemically balanced.

Split	Novel SMILES	Actual Rhea reactions	Balanced (n)	Balanced (frac.)
Random	12,200	68	1,157	0.09
MMseqs70	12,333	57	1,199	0.10
MMseqs30	9,247	47	779	0.08

4. Discussion

In our task, there are two alignments to be learned: reaction generation and protein-to-reaction mapping. Our model has learned to generate reactions reasonably well, with a few unseen reactions being generated, but it does not learn the correct mapping to proteins as reliably. The bottleneck, therefore, lies in protein-reaction alignment rather than in reaction generation, and the metabolite-level tokenization does not yield generalization; we regard these as the main findings of this work. In future work, we will focus on improving the protein representation

and on better interaction between the protein and reaction representations, which currently occurs only through cross-attention.

Traditional bioinformatics tools like blastp are highly effective at assigning functions to proteins through homology, as demonstrated in this work as well. However, they operate only at the protein level, not at the reaction level; they are not limited to enzyme functions, can identify similar proteins across any group, and are not specific to biochemistry. Deep learning tools like Horizyn-1 and CLIPZyme, on the other hand, process both protein and reaction information to find similarities. These tools can generalise to new reactions in the sense that, when unseen reactions are screened, they can process them and give relevance scores — but only when the unseen reaction is already known to us. A generative model, in contrast, has the potential to unearth unknown biochemical reaction chemistry. We are not able to achieve this with our current model, which may be due to a lack of diverse data compared with organic chemistry reactions. Despite this, we observe positive signs: new reactions, in the context of the model, are being generated. Although these reactions are not predicted for the right enzyme, this suggests that the identification of new reaction chemistry is not out of reach.

The mapping from proteins to seen reactions is better in our model than in the screening tool Horizyn-1, which in turn has the edge on unseen reactions. Tools like blastp and CLEAN show a superior ability to map proteins to seen reactions compared with our model, but fail on unseen reactions by design. RxnPred is therefore stronger than Horizyn-1 on seen reactions, but has not yet realized the generative advantage on unseen ones. Horizyn-1 can only retrieve reactions already known, whereas our model's limit is the closed metabolite vocabulary, which granular tokenization would remove. With current evidence, blastp and CLEAN remain the better choice for routine annotation, given blastp's higher precision at lower computational cost. RxnPred is preferable where multiple candidate reactions are acceptable, and downstream filtering is available. The advantage gained by including protein structures is marginal, but more experiments are needed. We also found that the addition of an EC number decoder to condition the reaction decoder did not help (Supplementary material Section S2.4).

Finally, we note three limitations of this work. First, for simplicity, all reactions are considered only in the forward direction, which may lead to false metabolic capabilities, especially when the predictions are used to construct metabolic networks. Second, the weighted multilabel metric may inflate the apparent capability of the model by diluting false positives. Although the metrics may appear favourable, a tool that returns the correct function alongside many false positives is less useful in practice than tools like blastp or CLEAN, which achieve better precision. Third, even where the model performs well on one subset of proteins, performance drops on another, which holds for the other tools as well. Developing more specialised models may address this going forward.

Acknowledgments

SB acknowledges funding from the Ministry of Education, Government of India.

References

1. L. Kennedy, M.-E. Harper and M. Cuperlovic-Culf, Computational approaches to enzymatic reaction assignment: a review of methods, validations, and future directions, *Briefings in Bioinformatics* **26**, p. bbaf635 (2025).
2. S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, Basic local alignment search tool, *Journal of molecular biology* **215**, 403 (1990).
3. S. R. Eddy, Profile hidden markov models., *Bioinformatics (Oxford, England)* **14**, 755 (1998).
4. T. Yu, H. Cui, J. C. Li, Y. Luo, G. Jiang and H. Zhao, Enzyme function prediction using contrastive learning, *Science* **379**, 1358 (2023).
5. Y. Yang, A. Jerger, S. Feng, Z. Wang, C. Brasfield, M. S. Cheung, J. Zucker and Q. Guan, Improved enzyme functional annotation prediction using contrastive learning with structural inference, *Communications Biology* **7**, p. 1690 (2024).
6. L. Dumontet, S.-R. Han, J. H. Lee, T.-J. Oh and M. Kang, Trustworthy prediction of enzyme commission numbers using a hierarchical interpretable transformer, *Nature Communications* **17**, p. 1146 (2026).
7. P. G. Mikhael, I. Chinn and R. Barzilay, Clipzyme: Reaction-conditioned virtual screening of enzymes, *arXiv preprint arXiv:2402.06748* (2024).
8. J. W. Rocks, D. P. Truong, D. Rappoport, S. Maddrell-Mander, D. A. Martin-Alarcon, T. M. Lee, S. Crossan and J. E. Goldford, Dual-encoder contrastive learning accelerates enzyme discovery, *Proceedings of the National Academy of Sciences* **123**, p. e2520070123 (2026).
9. C. Hua, B. Zhong, S. Luan, L. Hong, G. Wolf, D. Precup and S. Zheng, Reactzyme: A benchmark for enzyme-reaction prediction, *Advances in Neural Information Processing Systems* **37**, 26415 (2024).
10. P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. A. Hunter, C. Bekas and A. A. Lee, Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction, *ACS central science* **5**, p. 1572 (2019).
11. R. Irwin, S. Dimitriadis, J. He and E. J. Bjerrum, Chemformer: a pre-trained transformer for computational chemistry, *Machine Learning: Science and Technology* **3**, p. 015022 (2022).
12. T. Sagawa and R. Kojima, Reactiont5: a pre-trained transformer model for accurate chemical reaction prediction with limited data (2025).
13. P. Bansal, A. Morgat, K. B. Axelsen, V. Muthukrishnan, E. Coudert, L. Aimo, N. Hyka-Nouspikel, E. Gasteiger, A. Kerhornou, T. B. Neto *et al.*, Rhea, the reaction knowledgebase in 2022, *Nucleic acids research* **50**, D693 (2022).
14. Uniprot: the universal protein knowledgebase in 2023, *Nucleic acids research* **51**, D523 (2023).
15. G. Landrum *et al.*, RDKit: Open-source cheminformatics `https://www.rdkit.org`, (2026), Version 2026.03.3, `https://doi.org/10.5281/zenodo.20446949`.
16. M. Steinegger and J. Söding, Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets, *Nature biotechnology* **35**, 1026 (2017).
17. D. Bertoni, M. Tsenkov, P. Magana, S. Nair, I. Pidruchna, M. Querino Lima Afonso, A. Midlik, U. Paramval, D. Lawal, A. Tanweer *et al.*, Alphafold protein structure database 2025: a redesigned interface and updated structural coverage, *Nucleic Acids Research* **54**, D358 (2026).
18. Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli *et al.*, Evolutionary-scale prediction of atomic-level protein structure with a language model, *Science* **379**, 1123 (2023).
19. C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie, T. Sercu, A. Lerer and A. Rives, Learning inverse folding from millions of predicted structures, in *International conference on machine learning*, 2022.
20. W. Ahmad, E. Simon, S. Chithrananda, G. Grand and B. Ramsundar, Chemberta-2: Towards

chemical foundation models, *arXiv preprint arXiv:2209.01712* (2022).
21. M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov and L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, in *Proceedings of the 58th annual meeting of the association for computational linguistics*, 2020.
22. T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz *et al.*, Transformers: State-of-the-art natural language processing, in *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, 2020.
23. J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski *et al.*, Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation, in *Proceedings of the 29th ACM international conference on architectural support for programming languages and operating systems, volume 2*, 2024.
24. M. Freitag and Y. Al-Onaizan, Beam search strategies for neural machine translation, in *Proceedings of the First Workshop on Neural Machine Translation*, 2017.
25. T. Sanderson, M. L. Bileschi, D. Belanger and L. J. Colwell, Proteinfer, deep neural networks for protein functional inference, *elife* **12**, p. e80942 (2023).
26. M. N. Price, K. M. Wetmore, R. J. Waters, M. Callaghan, J. Ray, H. Liu, J. V. Kuehl, R. A. Melnyk, J. S. Lamson, Y. Suh *et al.*, Mutant phenotypes for thousands of bacterial genes of unknown function, *Nature* **557**, 503 (2018).
27. Z. Yang, B. Su, J. Chen and J.-R. Wen, Interpretable enzyme function prediction via residue-level detection, *arXiv preprint arXiv:2501.05644* (2025).
28. B. Buchfink, K. Reuter and H.-G. Drost, Sensitive protein alignments at tree-of-life scale using diamond, *Nature methods* **18**, 366 (2021).
29. G. B. Kim, J. Y. Kim, J. A. Lee, C. J. Norsigian, B. O. Palsson and S. Y. Lee, Functional annotation of enzyme-encoding genes using deep learning with transformer layers, *Nature Communications* **14**, p. 7370 (2023).

METEOR: Evidence-weighted reaction selection reconciles enzyme prediction with growth feasibility

Kexin Niu[1] and Robert Hoehndorf[1,2,3,4,†]

[1]*Biological and Environmental Sciences & Engineering (BESE) Division,* [2]*KAUST Center of Excellence for Smart Health (KCSH),* [3]*KAUST Center of Excellence for Generative AI,* [4]*Computer, Electrical and Mathematical Sciences and Engineering (CEMSE) Division, King Abdullah University of Science and Technology, 4700 KAUST, Thuwal 23955, Kingdom of Saudi Arabia*

[†]*E-mail: robert.hoehndorf@kaust.edu.sa*

Enzyme function predictors score each protein in isolation and never ask whether the resulting assignments describe an organism that can grow. Metabolism supplies a different kind of constraint: steady-state fluxes must satisfy mass balance and can be required to support biomass production, regardless of whether a protein resembles anything already characterised. Here we present METEOR, which aggregates predicted per-protein Enzyme Commission (EC) scores into reaction-level scores and solves a mixed-integer linear program (MILP) over the SEED universal reaction database to return a growth-feasible reaction set. A penalty $\mu(1 - w_j)^p$ charges each reaction by the confidence behind it, a post-solve check repairs solutions that do not grow on their own, and a bounded per-EC update maps the selected context back to protein scores. On six curated models METEOR recovers 60.4% of curated ECs that DeepProZyme scores below 0.5, against 48.0% for thresholding followed by parsimonious repair and 48.3% for a size-matched top-scoring baseline, while EC precision falls from 45.5% to 42.2%. When predictor scores are deleted, the confidence-derived cost admits 4–58× fewer off-reference reactions than confidence-independent or shuffled costs. Across three predictors and 108 bacterial genomes, METEOR models grow, unlike thresholded networks; a candidate-mask ablation attributes growth to the shared growth-feasible scaffold and the choice of reactions that join it to evidence weighting. Network constraints operate at the organism level: METEOR returns a growth-feasible reaction set but improves neither size-matched pathway coverage nor individual per-protein accuracy.

Keywords: Enzyme function prediction; Genome-scale metabolic reconstruction; Constraint-based modelling; Mixed-integer linear programming; Parsimony.

1. Introduction

Enzyme function annotation has moved from homology transfer[1–3] to sequence-based deep learning, including contrastive,[4] CNN,[5,6] and transformer[7] architectures. Accuracy has improved substantially, but the unit of inference has not: these methods score each protein on its own sequence against what the training set contains, treating two proteins from the same genome independently and never asking whether the resulting assignments describe an organism that could exist.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

Metabolism supplies a constraint that these predictors ignore. A feasible steady state must balance the production and consumption of every internal metabolite,[8] and EC assignments made one protein at a time need not satisfy it. A growth requirement can also favour a weakly scored reaction when that reaction completes a feasible route. Such a constraint informs the organism-level reaction set. Raising or lowering a confidence threshold does not impose it, and GLOBUS,[9] which adds genomic context, does not test whether the resulting assignment is metabolically consistent.

Two families of methods already couple reaction evidence with stoichiometric constraints. Context-specific extraction methods start from a generic model and reaction evidence that is already classified: FASTCORE extracts a flux-consistent subnetwork around a predefined core,[10] and CORDA retains reactions by confidence class together with the support they need.[11] Reconstruction tools such as CarveMe, gapseq and Reconstructor[12–14] commonly build a draft model and gap-fill it, adding reactions until the model grows;[15] ProbAnno ranks gap-fill candidates by homology-derived likelihood,[16] and other methods by subsystem, pathway or genomic context.[17–22] In these two-stage workflows the draft is fixed before repair: accepted reactions are never revisited, and rejected ones return only when growth demands them. Where likelihoods do become costs, as in ProbAnno, they apply only to reactions outside a fixed draft; none of these methods selects the whole reaction set under one confidence-derived cost or returns its result as protein scores.

METEOR makes the two decisions at once. It aggregates continuous per-protein EC scores into reaction-level costs and selects a growth-feasible reaction set from the SEED universal reaction database[23,24] in a single biomass-constrained optimisation, in which candidates compete under a cost that penalises weak support more than strong support (Section 2.2). The selected set is then written back to the protein–EC score matrix through a bounded update.

On 108 bacterial genomes, METEOR's selected sets grow where thresholded networks do not, a property the shared scaffold supplies (Section 3.1). METEOR recovers curated ECs that a fixed threshold discards, with higher recovery than a size-matched baseline and a measured precision cost (Section 3.2), and among the many growth-feasible sets the sequence-derived cost determines which is selected (Section 3.3). It improves neither size-matched pathway coverage nor per-protein accuracy (Sections 3.2 and 3.4). METEOR is predictor-agnostic and is freely available at `https://github.com/bio-ontology-research-group/Meteor/tree/v8.3-psb2027`.

2. Methods

2.1. *Datasets and benchmark construction*

Protein-level holdout. We built a temporal UniProtKB/Swiss-Prot[25] holdout of 227 bacterial proteins created between 2025-01-01 and 2026-01-28 (UniProt 2026_01), all postdating the evaluated predictors' training data. Among them, 125 have complete four-digit EC annotations across 92 classes. We linked proteins to genomes by strain-level `Proteomes` cross-reference, then manifest assembly, then exact sequence match within that proteome, preventing cross-organism matches from identical sequences. This yielded 66 proteins across 53 genomes for protein-level evaluation (metrics in Supplementary Table S2); we excluded 59 proteins for

missing strain proteomes, off-manifest assemblies, or absence from the genome panel.

Organism panel. Of the 227 holdout proteins, 208 map to an NCBI RefSeq assembly, covering 112 distinct assemblies. We excluded four assemblies with incomplete baseline coverage, yielding a panel of 108 bacterial genomes from 69 genera (63 Gram-negative, 45 Gram-positive). All organism-level analyses use this 108-genome panel; the 53 genomes containing a genome-resolved holdout protein also support the protein-level evaluation. Baseline predictions were matched to each proteome by exact sequence hash. Because this panel is derived from the holdout rather than sampled to represent bacterial diversity, organism-level claims are restricted to these 108 genomes.

METEOR uses the ModelSEED `Prokaryote_Universal` reaction database distributed with Reconstructor v1.1.0 (47,880 reactions, 23,013 metabolites).[14,23,24] We evaluated canonical pathway completeness with KEGG pathway-to-EC mappings[26] accessed on 2026-06-30. Of the 172 KEGG Metabolism pathways we retained the 151 with at least five SEED-representable four-digit ECs, the smallest size at which our three stringencies (≥1 EC, ≥25%, ≥50%) require distinct EC counts.

Baseline predictions enter METEOR in each method's native score space, except for DeepProZyme,[6] whose EC-specific decision boundaries are mapped to τ=0.5 by per-EC piecewise-linear calibration. CLEAN[4] and EnzBERT[7] use a single native threshold. We use these bounded scores as confidence inputs, although they are not calibrated as comparable probabilities; reaction confidence therefore inherits each predictor's score scale. Supplementary Section S2 describes sequence-identity-filtered retraining.

2.2. *METEOR framework*

METEOR (Figure 1b) takes the score matrix $E \in [0,1]^{N\times C}$ of one organism's N proteins over C ECs and returns a selected reaction set and a refined matrix E^{new} of the same shape. It also exports the EC annotations of selected reactions as a separate organism-level set, which may extend beyond the input vocabulary. *Evidence* means model-derived predictor confidence: a protein–EC score or its reaction-level aggregate w_j.

(1) *Noisy-OR aggregation.* For each of the J = 47,880 SEED reactions j, let $M(j)$ denote its possibly empty set of EC annotations. Protein-level evidence is aggregated as

$$w_j = 1 - \prod_{p\in\mathcal{P},\, e\in M(j)} (1 - E_{p,e}). \tag{1}$$

We keep only each protein's top $k = 5$ EC scores before aggregation and set the rest to zero, limiting inflation by small score tails (sensitivity in Supplementary Section S5).

(2) *Evidence-weighted cost mapping.* Each reaction confidence is converted to a linear cost

$$c_j = -\log \frac{w_j + \epsilon}{1 - w_j + \epsilon} + \mu(1 - w_j)^p, \tag{2}$$

a log-odds evidence term plus an evidence-weighted parsimony penalty; at the defaults the cost is negative above $w_j \approx 0.61$ (Figure 1b, step 2). The exponent p controls how sharply the penalty concentrates on weak evidence; $p = 0$ charges every reaction the same μ, the

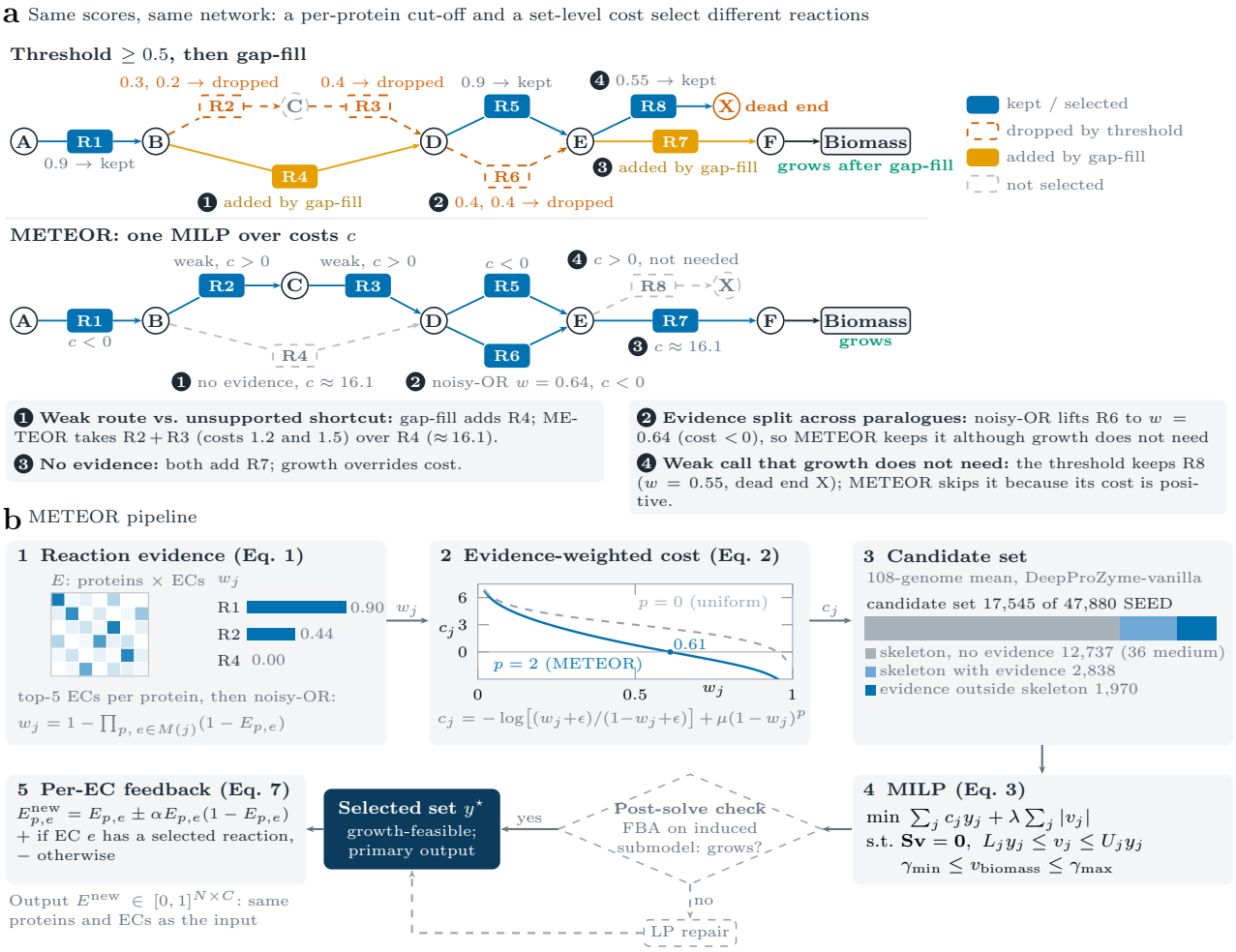


Fig. 1. METEOR vs. threshold-then-gap-fill. (a) Toy network with invented scores: from the same evidence, a per-protein cut-off and a set-level cost select different reactions (cases 1–4). (b) The pipeline; step 3 shows per-genome means on the DeepProZyme-vanilla panel. The selected set $y^\star$ is the primary output and is written back to E^{new}.

confidence-independent control of Section 3.3. The cost can retain evidence-supported alternative routes while discouraging unsupported ones (Figure 1a; Section 3.1). It cannot preserve or distinguish isozymes because the optimisation contains no protein-level variables. A reaction without evidence costs about 16.1 at the defaults (implementation details in Supplementary Section S1.1).

(3) *Metabolic network optimisation.* The MILP runs over a per-genome candidate set rather than all 47,880 SEED reactions. It contains the reactions with evidence $w_j \geq 0.01$ in that genome, the *medium reactions* (exchange reactions that supply the defined growth medium across the system boundary), and a biomass-feasible skeleton. The skeleton is computed once per Gram group by maximising biomass flux over the universal model and is shared by every genome in the group, so its reactions are selectable without evidence in a given organism; about 12,700 of the 17,545 candidates per genome are of this kind (Figure 1b, step 3; Supplementary Section S1.1).

Let $y_j \in \{0,1\}$ indicate reaction inclusion, $v_j \in \mathbb{R}$ its steady-state flux, $\mathbf{v}$ the flux vector over all candidate reactions, and $\mathbf{S}$ the stoichiometric matrix (metabolites by reactions). METEOR selects the reaction set by solving

$$\min_{y,v} \quad \sum_j c_j y_j + \lambda \sum_j |v_j| \tag{3}$$

$$\text{s.t.} \quad \mathbf{Sv} = \mathbf{0}, \tag{4}$$

$$L_j y_j \le v_j \le U_j y_j \quad \forall j, \tag{5}$$

$$\gamma_{\min} \le v_{\text{biomass}} \le \gamma_{\max}. \tag{6}$$

The objective minimises evidence-derived reaction cost together with a flux parsimony term, subject to mass balance (each internal metabolite is produced as fast as it is consumed), inclusion-gated flux bounds and a biomass-flux interval ($\gamma_{\max} = 2.5$). Medium reactions have $y_j = 1$; non-candidate reactions have $y_j = 0$. With CBC[27] at a 5% relative gap and a 600 s limit, 97 of 108 DeepProZyme-vanilla solve calls took less than 600 s and 11 took at least 600 s, consistent with time-limited runs. The solve call (including solver input and output) took a median of 137 s per genome (range 72–738 s) on four cores. Across all twelve configurations 624 of 1,296 solve calls took at least 600 s, most for CLEAN (Supplementary Table S1); settings, media and fixed reactions are in Supplementary Section S1.1.

(4) *Verify-and-repair.* The flux-selection coupling constraints $L_j y_j \le v_j \le U_j y_j$ operate as standard big-M bounds.[28] Because parsimony penalises skeleton reactions without evidence, the solver can leave them at $y_j \approx 0$ within its integrality tolerance while $U_j \approx 1000$ still passes the small flux the biomass floor needs. Consequently, the induced submodel (only the selected reactions and exchange reactions) may fail independent growth verification despite formal MILP feasibility. After each solve we run flux balance analysis (FBA) on the induced submodel and require $\max v_{\text{biomass}} \ge 0.05$. Where this fails, we add the support (the reactions carrying nonzero flux) of a linear program (LP) that minimises flux through reactions not already selected while satisfying the biomass floor. Every growth count in this paper refers to the repaired set (repair frequency in Section 3.1).

(5) *Constraint feedback propagation.* For each EC e, let $A_e = \{j : e \in M(j) \text{ and } y_j^\star = 1\}$ be the selected reactions associated with e. We update per-protein scores

$$E_{p,e}^{\text{new}} = \begin{cases} E_{p,e} + \alpha\, E_{p,e}(1 - E_{p,e}) & \text{if } A_e \neq \emptyset, \\ E_{p,e} - \alpha\, E_{p,e}(1 - E_{p,e}) & \text{if } A_e = \emptyset, \end{cases} \tag{7}$$

with step size α; the $E_{p,e}(1-E_{p,e})$ factor keeps updates within $[0,1]$. All proteins sharing an EC receive the same monotone transformation, preserving their ordering within that EC column but not necessarily the ordering of ECs within a protein. A reaction with several ECs (8.2% of EC-annotated SEED reactions) boosts all of them when selected; the method attributes nothing to a specific protein or paralogue.

We fixed ϵ=10^{-6}, $\gamma_{\min}$=0.1, λ=10^{-4} and α=0.4 before holdout evaluation. We chose p=2 and μ=3 on panel and holdout metrics, so the holdout is not an independent test of these two (sensitivity analyses in Supplementary Section S5).

Structural metrics. Dead-end metabolites (produced but never consumed, or the reverse) and mass-imbalanced reactions are counted with MEMOTE[29] on each selected submodel, identically for both methods. The dead-end count also includes metabolites an organism legitimately only secretes or takes up, so it is a structural diagnostic whose biological interpretation depends on exchange and secretion.

3. Results

3.1. *Growth feasibility comes from the shared scaffold*

For each predictor we compared a baseline network, containing every reaction with at least one protein score above $\tau = 0.5$, against METEOR's selected set across the 108-genome panel (Figure 2a; per-predictor selected-reaction counts and mass-imbalance fraction in Supplementary Section S7.7; Figure 2 also previews additional recovery and stability results from Sections 3.2–3.3).

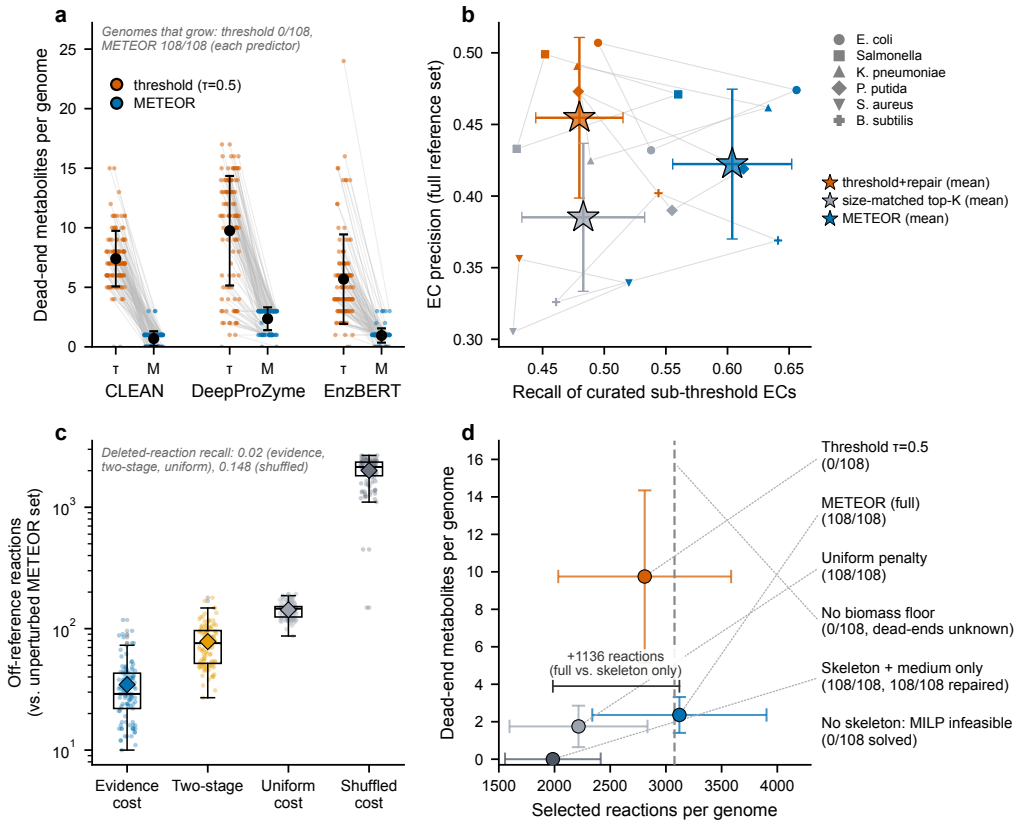


Fig. 2. Properties of METEOR-selected networks (mean ± s.d.; small points are genomes or organisms). **(a)** Dead ends per genome, threshold ($\tau = 0.5$, does not grow) vs. METEOR; a threshold network repaired to grow has a similar count (Section 3.1). **(b)** Recall of curated sub-threshold ECs vs. EC precision, six curated models; stars are method means. **(c)** Off-reference reactions under four deleted-evidence cost regimes (log scale; boxes: IQR; diamonds: mean). **(d)** Selected reactions vs. dead ends for five DeepProZyme ablations; open markers do not grow.

No thresholded network grows even with every exchange open: FBA on the submodel returns zero biomass flux for every predictor, and about half of the reactions in each submodel fail elemental-balance checks (Figure 2a; Supplementary Section S7.7).

Relative to the raw thresholded networks, mean dead-end counts fall by 76–91% (paired Wilcoxon $p < 10^{-18}$), with fewer dead ends in 104–106 of 108 genomes and ties in the rest. The candidate-mask ablation attributes this reduction to growth-feasible construction: the skeleton alone yields growing sets with no dead ends, and METEOR's 2.4 dead ends per genome (DeepProZyme) arrive with the evidence-supported reactions added to the skeleton (Section 3.3). A threshold network repaired to grow with the same LP (Section 3.2) has slightly fewer dead ends (2.23 against 2.36 per genome; equal in 95 of 108, METEOR higher in 11); METEOR's lower dead-end fraction (0.84 against 0.90 per 1,000 metabolites) reflects only its larger network (2,723 against 2,381 metabolites). The parsimony penalty controls size: without the $\mu(1 - w_j)^p$ term, log-odds evidence alone selects a mean of 5,966 reactions per genome against 3,121. About half of the reactions in every submodel fail elemental balance whichever method selects them, and the imbalanced fraction moves only from 0.49–0.51 to 0.46–0.50 (0.478 for the repaired threshold network against METEOR's 0.464). METEOR retains the SEED stoichiometry of each selected reaction; the small shift reflects which reactions are selected. Reaction selection ($y_j = 1$) permits both zero and nonzero flux. Flux variability analysis shows that 54.5% of METEOR's selected reactions can carry nonzero flux on average, compared with 40.5% in thresholded networks. METEOR improves both the flux-consistent fraction and absolute count in all 108 genomes (paired Wilcoxon $p = 1.9 \times 10^{-19}$ for both comparisons). In a single biomass-maximising flux solution only 13.3% of selected reactions carry flux, so the selected set describes candidate reactions across possible flux states.

Growth feasibility is imposed by the MILP but counted only after post-solve verify-and-repair (Section 2.2), which 77 of 1,296 raw solutions (5.9%) across the twelve configurations require and which adds a median of 98 reactions (range 1–230; Supplementary Table S1). After repair, all 108 METEOR submodels grow under the defined medium. Exported in the Systems Biology Markup Language (SBML) and re-tested with MEMOTE under a medium it infers per model, the repaired models grow in 105–108 of 108 genomes in the vanilla configurations and 83–108 across all twelve. MEMOTE's unbounded-flux fraction (lower is better) is 0.73–0.94 for METEOR against 0.545 for CarveMe on the same panel, a consequence of the universal template that METEOR does not correct (Supplementary Sections S7.1 and S7.2).

Thresholded networks are repairable by the same flux-parsimony LP, which restores growth in every genome (Section 3.3). The sequence-derived cost determines which feasible selection is returned. We next assess those selections against curated reference models.

3.2. *METEOR recovers curated sub-threshold ECs at a measured precision cost*

A hard threshold at τ=0.5 discards every EC scored below it, including ECs the organism carries. We measure that loss against curated reference models. For each of six curated bacterial genome-scale metabolic models (GEMs),[30–35] we collect the ECs annotated by the reference model that DeepProZyme scores between 0 and 0.5, then measure the recovery rate for each

Table 1. Recovery of sub-threshold ECs across six curated bacterial GEMs (*E. coli*, *Salmonella*, *K. pneumoniae*, *P. putida*, *S. aureus*, *B. subtilis*); denominator: reference ECs that DeepProZyme scores in $(0, 0.5)$, 102 ± 16 per organism, 612 in total. Mean ± s.d.; precision against the full reference EC set.

	Threshold + parsimonious repair	METEOR
Recall, curated sub-threshold ECs (n=612)	0.480 ± 0.039	$\mathbf{0.604 \pm 0.053}$
Precision, full EC set	0.455 ± 0.061	0.422 ± 0.057

method. This is the set that standard thresholding drops despite nonzero predictive evidence. The baseline thresholds the same predictor at 0.5 and repairs the draft model by minimising flux through external reactions, using the identical reaction database, medium, and biomass constraints; it differs from METEOR's binary selection in its two-stage objective and candidate pool (Supplementary Section S7.4).

METEOR recovers more of these ECs at a moderate precision cost (Table 1). The recall gain holds across all six organisms ($+0.124 \pm 0.030$), while the precision loss is smaller and equally consistent (-0.032 ± 0.012). Pooling across organisms rather than averaging yields the same trend, 368/612 against 293/612. Measured on DeepProZyme alone and on six genomes, these data apply only to this comparison.

The additional output contains a small fraction of reference-supported ECs. Across the six models METEOR emits 773 more ECs than the baseline (1,402 against 1,274 per model). Of these, 77 match curated annotation, 75 of those being curated sub-threshold ECs, yielding a marginal precision of 0.10 for the extra calls. Many remaining predictions fall outside the reference EC vocabulary (Section 3.4), where the comparison can neither confirm nor refute them. In a size-matched comparison, a baseline of the top-K ECs by score, with K equal to METEOR's EC count, recovers 0.483 of the sub-threshold set at precision 0.385, compared with METEOR's 0.604 at 0.422, and is lower in all six organisms (Supplementary Section S7.5).

The recovered ECs are specific capabilities such as dTDP-4-dehydrorhamnose reductase (EC 1.1.1.133, O-antigen assembly; checked against ENZYME[36] and KEGG[26]); every recovered EC is listed in Supplementary Section S7.5.

Pathway coverage depends on the number of ECs selected. Against the raw threshold METEOR gains 4.7–9.1 percentage points of mean KEGG pathway coverage, but against the size-matched top-K baseline it loses 2.7–2.8 points, in 107 of 108 genomes for CLEAN and in all 108 for DeepProZyme and EnzBERT (Supplementary Section S3.1). This comparison attributes the threshold-relative gain to output size.

3.3. *The sequence-derived cost chooses among feasible sets*

Many reaction sets satisfy the growth constraint equally well; when several reactions could supply the same biomass precursor, only the evidence separates them. To quantify this reliance on evidence, we remove predictive evidence and measure the resulting shift in the selected reaction set. For each genome we zero the EC scores behind 60 randomly selected, evidence-bearing, non-skeleton reactions (fixed seed; this may also strip evidence from co-annotated reactions) and re-solve over the same candidate set under four regimes: the evidence-derived

cost, a uniform cost ($p = 0$), shuffled evidence-derived costs, and a two-stage threshold-then-weighted-gap-fill of the kind ProbAnno[16] uses, run on METEOR's own costs and candidates. We count *off-reference* reactions, those absent from the unperturbed METEOR set. The reference is METEOR's own output, so this experiment measures selected-set stability under evidence deletion.

All four regimes grow, but they select very different sets (Figure 2c). The evidence-derived cost admits 34.6 off-reference reactions per genome, against 78.2 for the two-stage arm (2.3×), 142.9 for uniform cost (4.1×) and 2,006.6 for shuffled costs (58×), with the same ordering in both Gram groups (Supplementary Section S9). The uniform and shuffled arms keep METEOR's binary objective and change only the costs, so their gap isolates the evidence-to-cost mapping. The two-stage arm also changes the objective (costs per unit of flux outside a fixed draft), so its count is descriptive. Recovery of deleted reactions is low: shuffled costs reach 0.148 while adding 2,015 reactions. The comparison quantifies changes in the selected set.

Ablations on DeepProZyme (108 genomes; Figure 2d) show which component supplies what. Feasibility comes from the biomass floor and the skeleton: without the floor the set barely changes in size (3,077 vs. 3,121 reactions) but no submodel grows (0/108), and without the skeleton the MILP is infeasible in every genome. The skeleton with the medium alone already yields growing sets of 1,985 reactions with no dead ends, all needing repair (Supplementary Table S7). Evidence weighting decides what is added on top: the full cost selects 1,136 more reactions (3,121 against 1,985) at 2.4 dead ends, while a uniform penalty in place of $\mu(1 - w_j)^2$ keeps growth but shrinks the network to 2,215 reactions and retains less of the nonzero sub-threshold evidence (10.8% against 19.1%).

3.4. *Protein-level interface and evaluation on six curated organisms*

The confidence update step projects reaction-level selections back onto a protein–EC score matrix sharing the original vocabulary. Because its monotone per-EC transformation can still reorder ECs within a protein, we test protein-level predictions, mapping baseline and METEOR through the identical EC vocabulary (*net-only*). On the 66 temporal-holdout proteins, METEOR alters no top-1 call for any predictor and $F_{\max}$ shifts by at most 0.012; the same holds in all twelve baseline×cut-off configurations (Supplementary Table S2). For DeepProZyme on 69,666 NCBI-annotated proteins, top-1 shifts from 90.2% to 90.1%. On the temporal holdout, micro-F_1@0.5 changes by +0.015 for CLEAN, +0.008 for DeepProZyme and 0.000 for EnzBERT. Full-proteome results are stratified by training-set identity in Supplementary Section S8.

Against published BiGG GEMs for the six curated organisms of Section 3.2, METEOR matches CarveMe[12] on recall of the ECs annotated to selected reactions (0.884 vs. 0.893) while returning nearly twice as many ECs (1,521 vs. 818), and trails on every set-sensitive metric: precision 0.438 vs. 0.822, F1 0.585 vs. 0.854 (Supplementary Section S7.1). Much of the gap is vocabulary mismatch. METEOR emits ECs from the 6,598-EC ModelSEED mapping; the curated models and CarveMe use a 1,234-EC BiGG-derived[37] subset of it, and 76% of METEOR's apparent false positives fall outside that subset; their correctness requires a broader reference or targeted validation. Restricting both methods to the reference vocabulary raises

METEOR's precision to 0.761 against CarveMe's 0.822, narrowing the precision gap from 38 points to 6 (F1 0.817 against 0.854). CarveMe's BiGG template shares a lineage with this restricted evaluation standard; the residual gap is largest for the two Gram-positive species. Both vocabularies and every false positive are deposited.

Against a *same-predictor* baseline, constructed by thresholding at 0.5 and then minimising flux through reactions outside the draft (Supplementary Section S7.4), METEOR recovers marginally more (0.784 vs. 0.767, +0.017 mean; 6/6, p=0.031); both models are growth-feasible. This recall is lower than the 0.884 above because it scores only ECs in the predictor's vocabulary (1,402 per model), whereas the full reaction-associated EC set (1,521) also carries superseded EC numbers that SEED and the curated models still use (108–124 per organism). The reference sets are identical; the score matrix retains its input columns. The corresponding precision and sub-threshold results are reported in Section 3.2, with controls in Supplementary Section S3.

Gene-essentiality prediction shows no general advantage. Against experimentally determined essential-gene sets for the same six organisms, in eight organism-medium-reference combinations (Supplementary Section S7.6), METEOR beats the threshold-and-repair baseline only on *B. subtilis* (both media tested; mean Matthews correlation coefficient, MCC, 0.323 vs. 0.247 in rich medium), ties it on *E. coli* in minimal medium, is inconclusive on *S. aureus*, and loses on *E. coli* in rich medium and on *Salmonella*, *K. pneumoniae* and *P. putida*. The two *E. coli* cases are scored against different reference tables, and a 2×2 decomposition shows that the table, not the medium, drives the reversal: against the Keio reference[38] METEOR trails in both media (MCC 0.052 vs. 0.105 in minimal medium), and switching the medium alone moves scores by only 0.01–0.05. At the gene level, METEOR's wrong-direction calls on the medium switch under DeepProZyme are all five de novo purine genes, consistent with parsimony removing salvage routes; the effect is small. For the other three organisms, tested only under a rich medium, most of the gap has a different cause: the shared candidate pool contains reactions annotated as eukaryotic (e.g. mitochondrial nucleotide carriers) that the baseline's gap-fill admits regardless of evidence, and removing them roughly halves the MCC gap, reversing it for one organism-predictor pair. Of the reactions METEOR still declines with no alternative route in its own network, only 19% are in that organism's curated model, against 74% for reactions it keeps, which descriptively points to candidate-pool quality rather than a parsimony failure.

4. Discussion

4.1. *Reaction-level scope and protein attribution*

Two limits belong to the formulation. Evidence cannot reach ECs outside the predictor's vocabulary or lift a zero score, and unannotated proteins contribute none; only the candidate set can supply reactions for such ECs. Stratifying curated-EC recovery by vocabulary on the six curated organisms measures how much the candidate set supplies: of the 7–12% of curated ECs outside a predictor's vocabulary, METEOR's reaction-associated EC set recovers 55–71% against 31–40% for the skeleton alone (6/6 organisms for each predictor), and 2.5–4.4% of curated ECs have no candidate reaction at all (Supplementary Section S7.8). Selection also

runs post hoc as a discrete MILP, so training is separate from selection. A differentiable relaxation[39,40] with protein–reaction variables could couple these steps, at up to $O(NJ)$ variables; testing protein attribution would also require suitable labels and a defined vocabulary.

METEOR's evidence-weighted parsimony, which favours the fewest supported reactions, controls network size (Section 3.1) but has a cost for essentiality prediction: by removing redundant routes it can make a retained gene look essential when it is not, as for five purine genes in *E. coli* (Section 3.4).

4.2. *Limitations*

Several limitations bound what this paper claims. *Lack of protein attribution.* METEOR evaluates reactions at the organism level without resolving specific protein-to-reaction assignments. When a selected reaction maps to multiple ECs, all associated proteins receive uniform feedback updates, leaving paralogues indistinguishable and unannotated zero-score proteins unrefined.

Protein attribution and validation. Top-1 calls remain unchanged in every holdout configuration (Section 3.4). The evaluations use curated models, KEGG definitions, published experimental essential-gene sets, and the method's own solutions. Prospective enzyme assays and growth experiments would directly test the recovered functions.

Evidence aggregation and truncation. The noisy-OR formulation assumes independent, probability-like confidence inputs, whereas raw predictor scores are uncalibrated and biological functional annotations often exhibit inter-protein correlations. Reaction confidence can consequently increase with proteome or paralogue count. Although top-k ($k = 5$) truncation limits this effect, the available k sweep predates the final evidence-weighted formulation and shows an approximately twofold change in selected network size (Supplementary Section S5).

Reaction mappings and skeleton. EC-to-reaction mappings in the SEED universal contain errors that propagate into the MILP, and SEED sustains thermodynamically infeasible cycles relative to curated references.[41] The biomass-feasible skeleton is the support of one biomass-maximising LP solution per Gram group, not per genome, with alternative optima unsampled. Every genome in a Gram group shares this scaffold, which supplies most of the candidate set without organism evidence and, by the candidate-mask ablation (Section 3.3), both growth and the absence of dead ends.

Fixed medium and biomass. All genomes solve under a single minimal medium with oxygen, so aerobic pathways are available to every organism (including obligate anaerobes) and growth-coupled pathways are detected more reliably than non-coupled ones (secondary metabolism, denitrification). METEOR also presupposes a defined medium and a biomass reaction in the universal, of which SEED's prokaryote universal carries two Gram-specific variants. Eukaryotes would need a compartmentalised universal and their own biomass composition, and unculturable taxa an inferred medium that is itself a source of error, so we claim nothing beyond culturable bacteria.

Optimisation and evaluation. Four-threaded CBC with a 5% gap and no fixed seed can return different near-optimal sets between runs (re-solving the DeepProZyme-vanilla panel changed a mean of 1.6 reactions per genome, Supplementary Section S1.1). Nearly half of

the solve calls exceed the configured time limit; these wall-time counts are proxies for time-limited runs. Certification of termination reasons and achieved gaps requires solver logs and bounds. Solve time does not rise with candidate-set size (Spearman $\rho = -0.34$ over a narrow 16,000–17,900 range; Supplementary Section S7.3), so scaling to larger universals is untested; construction time scales roughly linearly with the universal. The 108 genomes include related taxa, so genome-wise tests are not 108 independent samples, and the full-proteome comparison is descriptive rather than a holdout test.

Scope. The sub-threshold comparison covers six curated models and DeepProZyme alone; the deletion experiment uses one perturbation per genome against METEOR's own output. Broader claims require additional predictors, organisms and perturbations.

5. Conclusion

METEOR converts per-protein EC scores into reaction costs, selects a growth-feasible reaction set in one optimisation, and writes the result back through a bounded per-EC update at the organism level. The shared scaffold makes the selected sets grow; evidence weighting decides which reactions join it, recovering 60.4% of curated sub-threshold ECs against 48.0% for threshold-and-repair and 48.3% for a size-matched baseline, at a marginal precision of 0.10 on the extra calls. METEOR shows no pathway-inference or per-protein gain, and no general gene-essentiality advantage across six organisms; the losses follow the reference table and candidate-pool quality more than parsimony (Section 3.4).

Availability. Code, results and supplement:
`https://github.com/bio-ontology-research-group/Meteor` (tag `v8.3-psb2027`; Zenodo `https://doi.org/10.5281/zenodo.21716682`). Baseline prediction matrices must be regenerated. The repository lists NCBI RefSeq accessions; Supplementary Section S1 gives solver settings and parameters.

Acknowledgments

This work was supported by KAUST baseline funding and by grants FCC/1/5932-11-01 and FCC/1/5940-07-02. We thank the KAUST Ibex HPC team for computational support. Large language models were used for language and structural editing, formatting and consistency checks; the authors developed all scientific content and remain responsible for the text.

References

1. S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, Basic local alignment search tool, *Journal of molecular biology* **215**, 403 (1990).
2. A. Bateman, L. Coin, R. Durbin, R. D. Finn, V. Hollich, S. Griffiths-Jones, A. Khanna, M. Marshall, S. Moxon, E. L. Sonnhammer *et al.*, The pfam protein families database, *Nucleic acids research* **32**, D138 (2004).
3. C. P. Cantalapiedra, A. Hernández-Plaza, I. Letunic, P. Bork and J. Huerta-Cepas, eggnog-mapper v2: functional annotation, orthology assignments, and domain prediction at the metagenomic scale, *Molecular biology and evolution* **38**, 5825 (2021).

4. T. Yu, H. Cui, J. C. Li, Y. Luo, G. Jiang and H. Zhao, Enzyme function prediction using contrastive learning, *Science* **379**, 1358 (2023).
5. J. Y. Ryu, H. U. Kim and S. Y. Lee, Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers, *Proceedings of the National Academy of Sciences* **116**, 13996 (2019).
6. G. B. Kim, J. Y. Kim, J. A. Lee, C. J. Norsigian, B. O. Palsson and S. Y. Lee, Functional annotation of enzyme-encoding genes using deep learning with transformer layers, *Nature Communications* **14**, p. 7370 (2023).
7. N. Buton, F. Coste and Y. Le Cunff, Predicting enzymatic function of protein sequences with attention, *Bioinformatics* **39**, p. btad620 (2023).
8. J. D. Orth, I. Thiele and B. Ø. Palsson, What is flux balance analysis?, *Nature biotechnology* **28**, 245 (2010).
9. G. Plata, T. Fuhrer, T.-L. Hsiao, U. Sauer and D. Vitkup, Global probabilistic annotation of metabolic networks enables enzyme discovery, *Nature Chemical Biology* **8**, 848 (2012).
10. N. Vlassis, M. P. Pacheco and T. Sauter, Fast reconstruction of compact context-specific metabolic network models, *PLoS Computational Biology* **10**, p. e1003424 (2014).
11. A. Schultz and A. A. Qutub, Reconstruction of tissue-specific metabolic networks using CORDA, *PLoS Computational Biology* **12**, p. e1004808 (2016).
12. D. Machado, S. Andrejev, M. Tramontano and K. R. Patil, Fast automated reconstruction of genome-scale metabolic models for microbial species and communities, *Nucleic Acids Research* **46**, 7542 (September 2018).
13. J. Zimmermann, C. Kaleta and S. Waschina, gapseq: informed prediction of bacterial metabolic pathways and reconstruction of accurate metabolic models, *Genome Biology* **22**, p. 81 (December 2021).
14. M. L. Jenior, E. M. Glass and J. A. Papin, Reconstructor: a COBRApy compatible tool for automated genome-scale metabolic network reconstruction with parsimonious flux-based gap-filling, *Bioinformatics* **39**, p. btad367 (June 2023).
15. I. Thiele, N. Vlassis and R. M. Fleming, fastgapfill: efficient gap filling in metabolic networks, *Bioinformatics* **30**, 2529 (2014).
16. M. N. Benedict, M. B. Mundy, C. S. Henry, N. Chia and N. D. Price, Likelihood-based gene annotations for gap filling and quality assessment in genome-scale metabolic models, *PLoS Computational Biology* **10**, p. e1003882 (2014).
17. A. Osterman and R. Overbeek, Missing genes in metabolic pathways: a comparative genomics approach, *Current Opinion in Chemical Biology* **7**, 238 (2003).
18. R. Overbeek, T. Begley, R. M. Butler, J. V. Choudhuri, H.-Y. Chuang, M. Cohoon, V. de Crécy-Lagard, N. Diaz, T. Disz, R. Edwards, M. Fonstein, E. D. Frank, S. Gerdes, E. M. Glass, A. Goesmann, A. Hanson, D. Iwata-Reuyl, R. Jensen, N. Jamshidi, L. Krause, M. Kubal, N. Larsen, B. Linke, A. C. McHardy, F. Meyer, H. Neuweger, G. Olsen, R. Olson, A. Osterman, V. Portnoy, G. D. Pusch, D. A. Rodionov, C. Rückert, J. Steiner, R. Stevens, I. Thiele, O. Vassieva, Y. Ye, O. Zagnitko and V. Vonstein, The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes, *Nucleic Acids Research* **33**, 5691 (2005).
19. M. L. Green and P. D. Karp, A Bayesian method for identifying missing enzymes in predicted metabolic pathway databases, *BMC Bioinformatics* **5**, p. 76 (2004).
20. M. N. Price, A. M. Deutschbauer and A. P. Arkin, GapMind: Automated annotation of amino acid biosynthesis, *mSystems* **5**, e00291 (2020).
21. C. Chen, C. Liao and Y.-Y. Liu, Teasing out missing reactions in genome-scale metabolic networks through hypergraph learning, *Nature Communications* **14**, p. 2375 (2023).
22. M. D. Boer, C. Melkonian, H. Zafeiropoulos, A. F. Haas, D. R. Garza and B. E. Dutilh, Improving genome-scale metabolic models of incomplete genomes with deep learning, *iScience* **27**, p. 111349

(2024).
23. C. S. Henry, M. DeJongh, A. A. Best, P. M. Frybarger, B. Linsay and R. L. Stevens, High-throughput generation, optimization and analysis of genome-scale metabolic models, *Nature Biotechnology* **28**, 977 (2010).
24. S. M. Seaver, F. Liu, Q. Zhang, J. Jeffryes, J. P. Faria, J. N. Edirisinghe, M. Mundy, N. Chia, E. Noor, M. E. Beber *et al.*, The modelseed biochemistry database for the integration of metabolic annotations and the reconstruction, comparison and analysis of metabolic models for plants, fungi and microbes, *Nucleic acids research* **49**, D575 (2021).
25. B. Boeckmann, A. Bairoch, R. Apweiler, M.-C. Blatter, A. Estreicher, E. Gasteiger, M. J. Martin, K. Michoud, C. O'Donovan, I. Phan *et al.*, The swiss-prot protein knowledgebase and its supplement trembl in 2003, *Nucleic acids research* **31**, 365 (2003).
26. M. Kanehisa and S. Goto, KEGG: kyoto encyclopedia of genes and genomes, *Nucleic acids research* **28**, 27 (2000).
27. J. Forrest, T. Ralphs, S. Vigerske, H. G. Santos, L. Hafer, B. Kristjansson, M. Lubin, M. Saltzman *et al.*, coin-or/cbc: Release 2.10.12 (August 2024).
28. J. P. Vielma, Mixed integer linear programming formulation techniques, *SIAM Review* **57**, 3 (2015).
29. C. Lieven, M. E. Beber, B. G. Olivier, F. T. Bergmann, M. Ataman, P. Babaei, J. A. Bartell, L. M. Blank, S. Chauhan, K. Correia, C. Diener, A. Dräger, B. E. Ebert, J. N. Edirisinghe, J. P. Faria, A. M. Feist, G. Fengos, R. M. T. Fleming, B. García-Jiménez, V. Hatzimanikatis, W. Van Helvoirt, C. S. Henry, H. Hermjakob, M. J. Herrgård, A. Kaafarani, H. U. Kim, Z. King, S. Klamt, E. Klipp, J. J. Koehorst, M. König, M. Lakshmanan, D.-Y. Lee, S. Y. Lee, S. Lee, N. E. Lewis, F. Liu, H. Ma, D. Machado, R. Mahadevan, P. Maia, A. Mardinoglu, G. L. Medlock, J. M. Monk, J. Nielsen, L. K. Nielsen, J. Nogales, I. Nookaew, B. O. Palsson, J. A. Papin, K. R. Patil, M. Poolman, N. D. Price, O. Resendis-Antonio, A. Richelle, I. Rocha, B. J. Sánchez, P. J. Schaap, R. S. Malik Sheriff, S. Shoaie, N. Sonnenschein, B. Teusink, P. Vilaça, J. O. Vik, J. A. H. Wodke, J. C. Xavier, Q. Yuan, M. Zakhartsev and C. Zhang, MEMOTE for standardized genome-scale metabolic model testing, *Nature Biotechnology* **38**, 272 (March 2020).
30. J. M. Monk, C. J. Lloyd, E. Brunk, N. Mih, A. Sastry, Z. King, R. Takeuchi, W. Nomura, Z. Zhang, H. Mori, A. M. Feist and B. O. Palsson, iML1515, a knowledgebase that computes escherichia coli traits, *Nature Biotechnology* **35**, 904 (2017).
31. I. Thiele, D. R. Hyduke, B. Steeb, G. Fankam, D. K. Allen, S. Bazzani, P. Charusanti, F.-C. Chen, R. M. Fleming, C. A. Hsiung, S. C. De Keersmaecker, Y.-C. Liao, K. Marchal, M. L. Mo, E. Özdemir, A. Raghunathan, J. L. Reed, S.-I. Shin, S. Sigurbjörnsdóttir, J. Steinmann, S. Sudarsan, N. Swainston, I. M. Thijs, K. Zengler, B. O. Palsson, J. N. Adkins and D. Bumann, A community effort towards a knowledge-base and mathematical model of the human pathogen Salmonella Typhimurium LT2, *BMC Systems Biology* **5**, p. 8 (2011).
32. Y.-C. Liao, T.-W. Huang, F.-C. Chen, P. Charusanti, J. S. J. Hong, H.-Y. Chang, S.-F. Tsai, B. O. Palsson and C. A. Hsiung, An experimentally validated genome-scale metabolic reconstruction of Klebsiella pneumoniae MGH 78578, iYL1228, *Journal of Bacteriology* **193**, 1710 (2011).
33. J. Nogales, J. Mueller, S. Gudmundsson, F. J. Canalejo, E. Duque, J. Monk, A. M. Feist, J. L. Ramos, W. Niu and B. O. Palsson, High-quality genome-scale metabolic modelling of Pseudomonas putida highlights its broad metabolic capabilities, *Environmental Microbiology* **22**, 255 (2020).
34. Y. Seif, J. M. Monk, N. Mih, H. Tsunemoto, S. Poudel, C. Zuniga, J. Broddrick, K. Zengler and B. O. Palsson, A computational knowledge-base elucidates the response of Staphylococcus aureus to different media types, *PLOS Computational Biology* **15**, p. e1006644 (2019).
35. Y.-K. Oh, B. O. Palsson, S. M. Park, C. H. Schilling and R. Mahadevan, Genome-scale reconstruction of metabolic network in Bacillus subtilis based on high-throughput phenotyping and

gene essentiality data, *Journal of Biological Chemistry* **282**, 28791 (2007).
36. A. Bairoch, The ENZYME database in 2000, *Nucleic Acids Research* **28**, 304 (2000).
37. J. Schellenberger, J. O. Park, T. M. Conrad and B. Ø. Palsson, Bigg: a biochemical genetic and genomic knowledgebase of large scale metabolic reconstructions, *BMC bioinformatics* **11**, 1 (2010).
38. T. Baba, T. Ara, M. Hasegawa, Y. Takai, Y. Okumura, M. Baba, K. A. Datsenko, M. Tomita, B. L. Wanner and H. Mori, Construction of escherichia coli k-12 in-frame, single-gene knockout mutants: the keio collection, *Molecular Systems Biology* **2**, p. 2006.0008 (2006).
39. B. Amos and J. Z. Kolter, OptNet: Differentiable optimization as a layer in neural networks, in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, , Proceedings of Machine Learning Research Vol. 70 (PMLR, 2017).
40. A. Agrawal, B. Amos, S. Barratt, S. Boyd, S. Diamond and J. Z. Kolter, Differentiable convex optimization layers, in *Advances in Neural Information Processing Systems*, (Curran Associates, Inc., 2019).
41. C. J. Fritzemeier, D. Hartleb, B. Szappanos, B. Papp and M. J. Lercher, Erroneous energy-generating cycles in published genome scale metabolic networks: Identification and removal, *PLoS computational biology* **13**, p. e1005494 (2017).

A SIGNED ELEMENT-CONTRIBUTION FRAMEWORK FOR DESCRIBING THE FUNCTION OF FUSION PROTEINS

Katsuhiko Murakami
Fujitsu Research, Fujitsu Ltd.
Kawasaki, 211-8588, Japan
E-mail: murakami.kt@fujitsu.com

A substantial fraction of cancers are driven by fusion genes, and interpreting a newly observed fusion candidate remains a practical problem. What a fusion protein does is decided by which functional elements survive the breakpoint, so its function is not determined by the presence or absence of domains, and existing annotation resources are insufficient for inference. What they omit is direction: whether keeping a given element moves the phenotype toward disease or away from it. We formalize the classes of oncogenic mechanism as a sum of context-dependent signed element responses, extend the annotation of fusion protein elements with that direction, and catalogue the conditions under which it reverses. Applying the framework to solid-tumour fusions from COSMIC selected for breakpoint diversity, we build a tool that matches an unknown fusion against known ones on a role composition expressing the role of each element in a pathogenicity context, and that declines to answer, with a coverage guarantee, when no known mechanism is close enough. Identification of the mechanism class was not achieved, and we locate the reason in the representation itself: between breakpoints of one gene pair what varies is how much protein remains, not which functional roles remain. We report term by term which parts of the framework current knowledge and current annotation can support.

Keywords: Gene fusion; functional annotation; conformal prediction; breakpoint; oncogenic mechanism.

1. Introduction

Computational prediction of protein function has long relied on homology transfer. Recent AI methods have greatly improved the detection of distant homologues, yet determining protein function remains difficult when the functional elements are known but their organization is altered.

Fusion proteins illustrate this limitation, and the problem is a practical one: sequencing now reports new fusion candidates whose consequence must be judged. Similarity search usually identifies their retained domains correctly, but function depends on which elements survive the breakpoint and how they act after rearrangement. Breakpoint variants of the same gene pair can differ in disease and in drug response, as BCR-ABL p190 and p210 do, and as EML4-ALK variants do under tyrosine kinase inhibitors. Homology provides a parts list but not the consequence of the rearrangement.

Existing resources describe the parts but not their direction. Pfam, InterPro and Gene Ontology annotate domains and molecular functions; fusion-specific resources record domain retention [1-3] or predict pathogenicity for a gene pair or a transcript [4-10]; none states whether keeping a given element pushes the phenotype toward disease or away from it, which we call the *sign* of the element. Gene Ontology cannot supply it, because it annotates gene products rather than elements [11] and its negative-regulation terms relate processes rather than the consequence of losing an element.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

Reviews of fusion activation describe the mechanisms [12], and one already links the intron in which the break falls to the retained domains and to disease in ABL1 fusions [13], but neither assigns a direction to each element.

Our own earlier work [10] predicted the pathogenicity of a fusion as a whole and could not say which retained element carries the effect; the present framework is a response to that limitation, on an analysis set rebuilt independently from COSMIC that does not overlap the earlier data. Since no published method returns a signed description at the level of the element, we state these differences rather than report a head-to-head comparison.

A description is worth having if it separates what differs, so writing down the sign raises two questions. Does the set of signed elements that survive a breakpoint distinguish the mechanism by which a fusion is oncogenic? And when the breakpoint moves within one gene pair, as between the variants above, what changes: which roles are kept, or only how much of the protein? The answer to the second question turns out to explain the answer to the first.

We formulate the contribution of each element as a context-dependent signed response, catalogue the conditions under which the sign reverses, and build a precedent search on the resulting role composition, calibrated by conformal prediction so that it declines to answer when nothing known is close enough. We then report, term by term, which parts of the framework current evidence supports, which it rejects, which cannot be tested for want of annotation, and which are defined but not yet built.

2. Theory: signed description under a structural event

The framework below does not aim to exceed existing methods in predictive accuracy but to bring observations reported separately into a common descriptive form. We therefore keep the components deliberately simple: role assignment is rule-based without learning, the precedent search uses one-nearest-neighbour with Euclidean distance, and its reliability rests on the distribution-free guarantee of conformal prediction (Section 3.6). Keeping the prediction simple is what allows the location of the discriminating information to be traced by hand.

2.1. *Objects and operations*

We use *functional elements* to mean annotation units that contribute to protein function, including domains, regions and motifs. A fusion joins two genes, which we call the *gene pair*; the position at which the two proteins are joined is the *junction*, and two junctions of the same gene pair are *breakpoint variants*. The *mechanism class* is the type of oncogenic mechanism a fusion belongs to, and we use three: kinase activation, transcription-factor chimera, and other. Let a molecule M consist of a set of annotated functional elements $E = \{e_1, \ldots, e_n\}$. A structural event τ transforms M into M'. For each element we define a *functional integrity* $x_i \geq 0$, where $x = 1$ means intact, $x = 0$ means lost, $0 < x < 1$ means partial retention, and $x > 1$ means duplication. The retained fraction in a fusion is a special case of x. We do not cap x at 1 because events such as internal tandem duplication increase the number of copies of an element.

2.2. *Signed response model*

We define the annotation as three maps. A *role map* ρ assigns each element a role, written r_i for element i, among effector, autoinhibitory, oligomerization and neutral. A *sign map* s returns -1, 0 or +1 given a role and a context c. A *response function* g converts functional integrity into the magnitude of the contribution. The estimated contribution of role r to the phenotype of M', written $\hat{y}_r(M')$, is

$$\hat{y}_r(M') = \frac{1}{|E_r|} \sum_{i:r_i=r} s(r,c)\, w_i\, g_r(x_i)\,, \qquad E_r = \{i : r_i = r\} \tag{1}$$

where E_r is the set of elements with role r and $w_i \geq 0$ is the weight of the element. In this paper all weights are equal ($w_i = 1$) and g is the identity, so the estimator reduces to the mean retained fraction within each role; Eq. (1) is a definition, and no parameter is fitted to data. An effector acts toward pathogenicity when retained and takes sign +; an autoinhibitory element acts toward pathogenicity when lost and takes sign -; a neutral element takes 0. What existing fusion annotation resources do is the case in which s is always +1 and g is one if any part of the element remains and zero otherwise: an unsigned, all-or-nothing judgment, a degenerate case of this framework. Our generalization has three parts: introducing the sign s, making the response g continuous, and letting the sign depend on context.

The advantage of this framework is that each term can be tested separately as a comparison of nested models: whether the first-order term is supported, whether a non-monotone term adds explanatory power, and whether separating the sign by role is worthwhile. The same framework applies to other events that change the composition of elements, such as exon skipping (MET exon 14), in-frame deletion (EGFRvIII) and internal tandem duplication (FLT3-ITD).

2.3. *Two-stage resolution of the sign and reversal conditions*

The sign is resolved in two stages. The default is a map at the level of the domain family; if an entry exists in the catalogue of reversal conditions at the level of the gene or of the gene pair, that entry takes precedence. The structure is that of a position-specific scoring matrix overriding a general substitution matrix at particular positions.

Besides the family default, reversal conditions arise at the level of the gene, such as the PKC catalytic domain or the kinase domains of tumour-suppressive kinases, and of the gene pair, such as CLTC; Section 4.3 lists them.

If no entry applies and there is no assurance that the family default holds for the gene in question, no sign is assigned and the element is left unresolved, as the precedent search returns the empty set when nothing known is close enough.

The catalogue of reversal conditions is keyed on genes and gene pairs, not on fusions. A new combination of known genes can therefore receive the sign override even though the fusion itself is unknown. In the current implementation, however, gene-level resolution is not connected, so this route is defined in the framework but not reached (Section 4.10).

3. Materials and methods

3.1. *Data and analysis set*

We used COSMIC Fusion v100 (GRCh37) [14] with gene and transcript models from Ensembl GRCh37 release 73 [15] accessed through pyensembl. We restricted the data to solid tumours and, where both orientations of a fusion were reported in the same specimen, normalized to the driver orientation, obtaining 224 pairs for which the reading frame of the fusion protein could be reconstructed. We then restricted these to the 82 pairs in which two or more distinct predicted protein sequences were observed. Fifteen of those pairs are promoter-hijacking pairs, in which the 5' gene supplies only regulatory sequence and no protein; two of them (CDH11-USP6 and COL1A1-USP6) were removed before annotation and the remaining thirteen at the mechanism-class step, so the counts pass through 80 annotated pairs with 457 junctions and end at 67 pairs and 373 junctions, the main analysis set. This selection is deliberate: a change between breakpoints of one pair is defined only when the pair has more than one junction, so the 142 of 224 pairs (63%) seen with a single junction are excluded and the results do not generalize to them.

3.2. *Definition of the junction*

The proteins encoded by the two fusion partner genes are called the 5' and 3' partner proteins, and a junction is the position at which they are joined. Because the count depends on what is treated as the same junction, we compared three definitions (experiment E-1): agreement of the text representation, agreement of the pair of genomic coordinates, and agreement of a hash of the predicted protein sequence. We adopt the third. Since the question is the function of the fusion protein, coordinate differences that give the same protein are functionally equivalent. For the same reason, fusions with different 5' partners that yield the same protein are merged; among the 80 annotated pairs this happens for three junctions, all of the promoter-hijacking type, and it does not occur in the 67-pair analysis set.

3.3. *Role annotation, retained fraction, and the catalogue of reversal conditions*

For the 5' and 3' partner proteins of the 80 annotated pairs, the feature table of UniProt [16] lists 9,081 functional elements over 108 proteins. To assign a role and an expected sign to these 9,081 elements we constructed rules over element names (experiment E-2). Two element counts therefore appear in this paper: the 9,081 elements of the 80 annotated pairs, used for the rule table and for the null comparison of Section 4.4, and the 8,131 of them that belong to the 90 proteins of the 67-pair analysis set, used for the GO survey, the precedent search, the within-pair tests and the tables of Sections 4.8 and 4.9. The complete set of rules is given in Table 1.

We mapped the genomic coordinate of the breakpoint onto the protein coordinate of the representative transcript and computed, for each functional element (domain, region, motif and the like), the fraction retained in the fusion protein, which we call the *retained fraction*. This computation depends only on coordinates and not on the reading frame or the position of the stop codon. The *role composition* of a junction is a four-dimensional vector of the mean retained fraction for each of the four roles.

The catalogue of reversal conditions was assembled from primary literature describing mechanisms in which the expected sign does not hold. Each row records the target element, the default interpretation, the condition under which reversal occurs, and the mechanism.

3.4. *Survey of existing resources and of GO*

To ask whether existing resources can supply a sign at junction resolution, we tabulated the main fusion-related resources by whether they reach breakpoint resolution, whether they carry a pathogenicity label, and how they treat fusions (experiment E-3).

To test whether GO can supply the sign, we retrieved GO annotations for the 90 partner proteins from UniProt. GO annotations were retrieved with and without electronically inferred (IEA) annotations, to test whether the conclusions depended on automatically assigned annotations. We extracted every term whose name contains "negative regulation" and classified whether the target is the activity of the annotated protein itself or another pathway. Classification was not automated: the rule was fixed before the results were seen and every term was read.

3.5. *Null comparison for the distribution of breakpoints*

To ask whether breakpoint positions are random, we constructed a null model in which breakpoints fall uniformly on the protein coordinate (experiment E-4) and compared, against the observed data, the probability that an element is completely retained and the probability that it is completely lost. Confidence intervals were obtained by bootstrap resampling at the level of the gene pair and by the Clopper-Pearson method [17]. Because real breakpoints fall in introns rather than at arbitrary amino-acid positions, we repeated the comparison with two further null models in which the cut is placed at an intron of the representative transcript, either uniformly over the introns or with probability proportional to intron length, the latter being the same structural assumption used for the selection strength. Point estimates are means over elements, and it is the bootstrap that resamples gene pairs.

3.6. *Calibrated precedent search*

The purpose of experiment E-5 is to return known fusions whose role composition is close to a new junction, and to state in a calibrated form how close is close enough to be trusted. The E-value of BLAST gives the probability of such similarity arising by chance; it does not give the probability that a hit carries the correct function. What is needed is a guarantee of the latter kind.

We therefore used conformal prediction [18,19]. It takes any score as a *nonconformity score*, a measure of poorness of fit, and returns a candidate set of mechanism classes. For a query x and a candidate class y, a conformal p-value is defined against the nonconformity scores $A_1, \ldots, A_n$ of a calibration set as

$$p_y(x) = \frac{|\{i: A_i \geq A(x,y)\}| + 1}{n+1}, \qquad \Gamma_\alpha(x) = \{y \in Y: p_y(x) > \alpha\} \tag{2}$$

where Y is the set of mechanism classes, n is the size of the calibration set and α the target error rate, so that, as far as calibration data and query data are exchangeable, $P(y_{\text{true}} \in \Gamma_\alpha(x)) \geq 1 - \alpha$ holds. The guarantee depends neither on the quality of the score nor on the learning model. Unlike models that produce uncalibrated point predictions and offer no guarantee on future data, conformal

prediction returns statistically backed sets for a target error rate without retraining, which matches our motivation: evidence with a mathematical basis rather than fluctuating uncertainty. A larger set means greater uncertainty, and an empty set states that nothing known is close enough.

We used, as the nonconformity score, the minimum Euclidean distance from the role composition of the query junction to known junctions belonging to each mechanism class, and returned a candidate set of mechanism classes ($\alpha = 0.2$, nominal coverage 80%). If the query retrieves another junction of the same gene pair, the match is made by the shortcut of the gene name rather than by role composition; to avoid this we removed the entire pair to which the query belongs from the search set (leave-one-pair-out, 20 random seeds). Coverage is the fraction of queries for which the returned set contained the true mechanism class.

3.7. *Test of the within-pair effect and variance partition of the role composition*

To test whether the strength of selection changes when the breakpoint moves within a pair, we used a mixed-effects model absorbing the gene pair as a random effect (experiment E-6), comparing by likelihood-ratio test a model with only first-order monotone terms against one adding non-monotone terms. The response variable is the selection strength of a junction on the log scale, log(O/E), where O is the number of specimens in which the junction was observed and E is the number expected under a structural null in which a breakpoint falls in an intron with probability proportional to intron length. The monotone model has four predictors, the mean retained fraction of the domain and region layers on each side; the role-split model also has four, the retained fraction of the four roles. The two are not nested in one another and each is compared against the same null model.

If nothing is detected in that test, the question remains whether there was any variation within pairs to predict. To measure this without using the response variable, we partitioned the variance using the predictors alone. For each role we computed the fraction of pairs in which the retained fraction crosses the functional boundary of 0.5 between junctions of the same pair, and call this the *crossing rate*; 0.5 marks whether an element is functionally preserved or lost. The crossing rate of the neutral component was taken as the baseline and the decision rule was fixed before the analysis was run. A high crossing rate means that the role varies within a pair, so the annotation resolves differences between breakpoints of the same gene pair; a low crossing rate means the role is effectively a property of the pair and the annotation cannot express such differences. To check whether the crossing rate depends on the number of junctions, we computed crossing rates stratified by junction count, applied direct standardization to the junction-count distribution of the neutral component, and used promoter-hijacking pairs as a negative control.

3.8. *Ceiling of annotation and matching of clinically distinguished variant pairs*

To measure how far annotation could reach if it were extended, we scanned the UniProt feature table of the 90 partner proteins and counted whether a description corresponding to each role exists (experiment E-7), comparing the current number of annotations with the number of proteins for which such a description exists, which we take as the ceiling. For three variant pairs reported to show different clinical outcomes, we computed the mean retained fraction by role and asked whether the two members can be distinguished at the level of roles.

Data and code availability. The analysis set is derived from COSMIC, whose licence does not permit redistribution, so the junction-level data cannot be released; the rules that assign roles are listed in full in Table 1 and the catalogue of reversal conditions in Section 4.3, and the steps above are described in enough detail to be reimplemented from COSMIC and UniProt obtained from their providers. Analyses use the releases given in Section 3.1 and UniProtKB retrieved in July 2026. Analyses were performed with Python 3.14, SciPy 1.18 and statsmodels 0.14.

4. Results

4.1. *Neither existing resources nor GO supplies the sign*

The survey showed that although resources reach breakpoint resolution, none annotates pathogenicity experimentally at that resolution. COSMIC reaches the junction but its label is only the observed case count; FusionGDB2 annotates breakpoints and domain retention but only computationally. OncoKB [20], CIViC [21] and ChimerDB carry pathogenicity labels but at the level of the gene pair, without distinguishing junctions. The ClinGen/CGC/VICC standard operating procedure, the community rule set for calling a somatic variant oncogenic [22], places fusions out of scope. Resources that reach the breakpoint carry no label, and resources that carry a label do not reach the breakpoint.

In the GO survey, of the 118 "negative regulation" terms carried by the 90 proteins, one (0.8%) had the activity of the annotated protein itself as its target and 115 (97.5%) targeted other pathways or processes; the remaining two could not be decided from the term name alone (negative regulation of protein-containing complex assembly on RAF1, and negative regulation of catalytic activity on NOTCH1), and rather than force the binary rule we left them unassigned. The term on BRAF, "negative regulation of apoptotic process", is of the latter kind; suppression of apoptosis is oncogenic, so the direction of the string and the direction of the contribution to the phenotype are opposite. GO's negative regulation describes a relation between molecules, a quantity distinct from the direction an element carries under a structural event.

4.2. *Content and coverage of the role annotation*

Table 1 shows the result of applying the rules to the 9,081 elements (experiment E-2). A clearly non-neutral role could be assigned with confidence to about 9% of the total; the remainder fell to Neutral (default). The mean retained fraction of 0.000 for autoinhibitory elements means that every observed fusion has lost autoinhibition. The neutral component is in effect a proxy for the fraction of sequence retained in the fusion protein.

4.3. *Catalogue of conditions under which the sign reverses*

The catalogue starts from the two processes that reviews of fusion activation agree on, enforced oligomerization and inactivation of inhibitory domains [12], asks of each where the expected sign can fail, and anchors every type to a primary report; it is a curated list of counterexamples rather than a systematic review, and claims neither completeness nor frequency. We collected seven types of mechanism in which the expected sign does not hold; expanded to concrete fusion instances these

become eleven, three of which can be confirmed in our analysis set and the others rest on the literature. Against a default of "should be lost", the N-terminal autoinhibition of BRAF can be retained and the protein still be oncogenic, because the 5' partner tethers the near-full-length kinase to the membrane system; an intact dimer interface is required, but it is not what the partner supplies [23]. The PKC regulatory domain gives dominant-negative suppression when the kinase is lost [24], and the ABL cap is layered, so the myristoyl cap is lost while SH3 and SH2 survive and impose a second level of autoinhibition [25,26]. Against a default of "should be retained", the PKC catalytic domain does not confer stable activity [24], the beta-propeller of EML4 breaks folding when partially retained but is stable when completely lost [27], and in CLTC the region annotated as the trimerization domain is retained in CLTC-ALK but lost in CLTC-TFE3, whose partner carries a dimerization motif of its own [28,29]; we read the shift of oligomerization from the 5' to the 3' side off the retained regions, as neither fusion has been assayed for oligomerization, and it is a pair-level condition. Two entries fall outside these defaults: the HMGA2 3'UTR, whose loss escapes let-7 repression [30] and which lies outside the protein role vocabulary, and kinase domains in general, since in tumour-suppressive kinases such as STK11, CSK, MATK and DAPK1 the family does not determine the sign. All four BRAF fusions in our set had lost the autoinhibitory region, consistent with the known mechanism; the reversal cases reported in the literature are not present in our set.

Table 1. Rules for role assignment and their distribution in the data.

Role (expected sign)	Elements	Junctions	Mean retained fraction	Matching rule (applied in this order)
Effector (+)	418	319	0.636	name contains kinase; or ETS, basic-leucine, bHLH, homeobox, zinc finger, DNA, RanBP2, bromo, SET, PHD, chromo, Myb
Autoinhibitory (-)	44	22	0.000	name contains RBD, Ras-binding, phorbol or autoinhibitory, or the word C1
Oligomerization (+)	389	149	0.629	coiled-coil layer; or name contains SAM, Pointed, PB1, SCAN, trimerization, oligomerization, zipper, or begins with WD
Neutral (explicit)	2,481	451	0.547	disordered or motif layer; region not matched above
Neutral (default)	5,749	454	0.702	domain not matched by any rule above (default)

4.4. *Breakpoints converge on effector retention and autoinhibition loss*

To ask whether breakpoint positions are non-random, experiment E-4 compared a uniform-cut null model against the observations. The retained fraction by role is bimodal (Fig. 1A): 61% of effector elements are completely retained and 36% fall below 0.5, with almost nothing in between. Against the null, complete retention of the effector is 1.35-fold (95% CI 1.21-1.51) and complete loss of autoinhibition is 1.47-fold (1.41-1.49), both with lower bounds above one; autoinhibition is lost completely in all five pairs where it is defined, so that interval reflects only the uncertainty of the null. The probability that both hold simultaneously is 4.29-fold that of the null. The median case count of the group not retaining the effector is 2, outside the typical composition of recurrent fusions. Placing the cut at an intron instead makes the departure smaller but does not remove it: complete retention of the effector is 1.27-fold (1.11-1.42) when cuts fall uniformly over introns and 1.20-fold (1.10-1.28) when they are weighted by intron length, and complete loss of autoinhibition is 1.35-

fold (1.32-1.36) and 2.79-fold (2.74-2.97). Exon structure thus accounts for part of the departure, and the lower bound stays above one under every null.

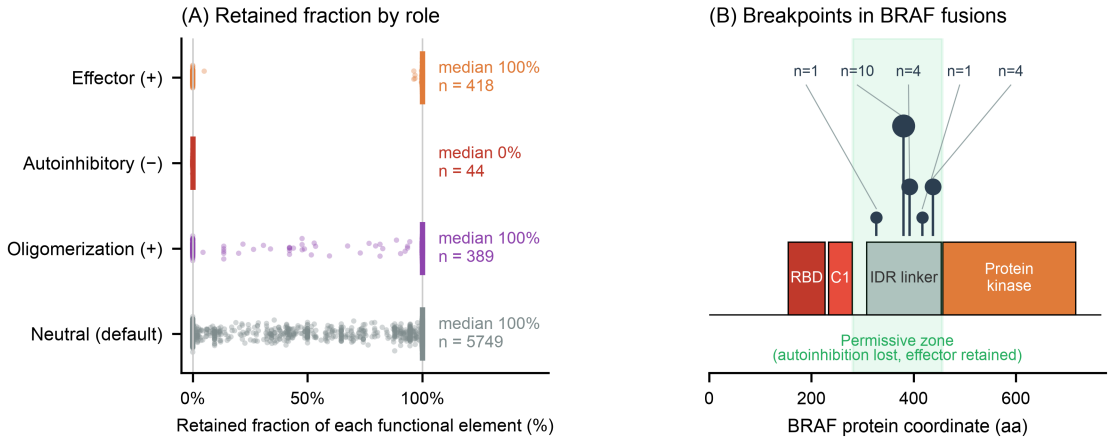


Fig. 1. (A) Retained fraction of each functional element by role, from 0% (lost entirely) to 100% (kept entirely). Each point is one element and the thick bar marks the median. Effector and autoinhibitory elements are almost always kept or lost whole, oligomerization elements mostly so, and partial cuts fall mainly on neutral elements. (B) Breakpoints in BRAF fusions on the amino-acid coordinate of BRAF. Boxes are domains (red, autoinhibitory; orange, effector kinase; grey, intrinsically disordered linker); the green band is the permissive zone, where autoinhibition is lost and the effector retained. Each stem marks an observed breakpoint, with circle size and the number along the top giving its junction count. In every fusion observed here BRAF is the 3' partner (20 junctions in four pairs), so the N-terminal autoinhibitory region is the part that can be lost; fusions with BRAF as the 5' partner are known but do not pass the filters used here.

All 20 junctions of BRAF fusions fall inside the region that loses autoinhibition and retains the effector kinase (Fig. 1B). This is a case study confined to BRAF rather than a general validation, since every pair for which autoinhibition is defined belongs to the BRAF/RAF1 family.

4.5. *Precedent search returns a guaranteed set*

In experiment E-5 (Section 3.6), observed coverage was 0.778, close to the nominal 0.80; at coverage matched to 80%, the mean size of the non-empty sets was 1.57, below the frequency-ordered baseline of 2.00, with an information gain of 0.80 bits. Removing the query's own gene pair is necessary: without it, the nearest neighbour would come from the same pair for 202 of 373 junctions (54%). Other pairs that share one gene with the query do remain in the search set, however, and coverage is higher for queries with such a pair (0.812) than for those without (0.566), so part of the search's performance may rest on a shared partner gene rather than on the role composition alone. For 15.9% of queries the set was empty: nothing known was close enough. To ask whether the roles carry information beyond the retained fraction, we repeated the search with the role composition replaced by the total retained fraction alone. Coverage is unchanged, since conformal prediction calibrates it by construction, but the returned sets are larger: 2.04 classes at matched coverage against 1.57 for the role composition, which is the size of the frequency-ordered baseline of 2.00, and the information gain falls from 0.80 to 0.50 bits. Permuting the role labels among elements gives 2.17 classes and 0.39 bits, so the gain is not an effect of the number of dimensions. Replacing each junction by the mean role composition of its gene pair left the search essentially unchanged (mean

set size 1.53 versus 1.57), consistent with mechanism class being a pair-level label in the present evaluation. The discriminating quantity is an amount, but averaging that amount over all elements destroys it; the roles are what make it informative.

4.6. *Mechanism-class identification does not succeed, but the cause is located*

The nearest neighbour had the same mechanism class as the query more often than always answering the most frequent class would (0.68 against 0.59), but not consistently: it did better in 15 of 20 splits and worse in 5, and the difference is not distinguishable from zero. Coverage was uneven across classes, at 0.593 for kinase activation.

The cause can be identified. Discrimination by the role composition is carried in effect by the neutral component, the total amount of sequence retained in the fusion protein. In oncogenic fusions the three components effector, autoinhibitory and oligomerization converge on approximately (1, 0, 0) and produce no difference between junctions. In the three cases KIAA1549-BRAF, SND1-BRAF and MYB-NFIB these three components are identical, and the ranking is determined only by small differences in the neutral component (distances 0.0006 and 0.0045, Fig. 2). In Fig. 2 these distances appear as 0.00: the compositions differ only beyond the second decimal, in the neutral component. The role space is not completely collapsed only because the neutral component is continuous; on the three functional components alone, many junctions coincide at the same point. The information the current role vocabulary retains is therefore how much was retained, and not which effector, which is what distinguishing mechanism classes would require.

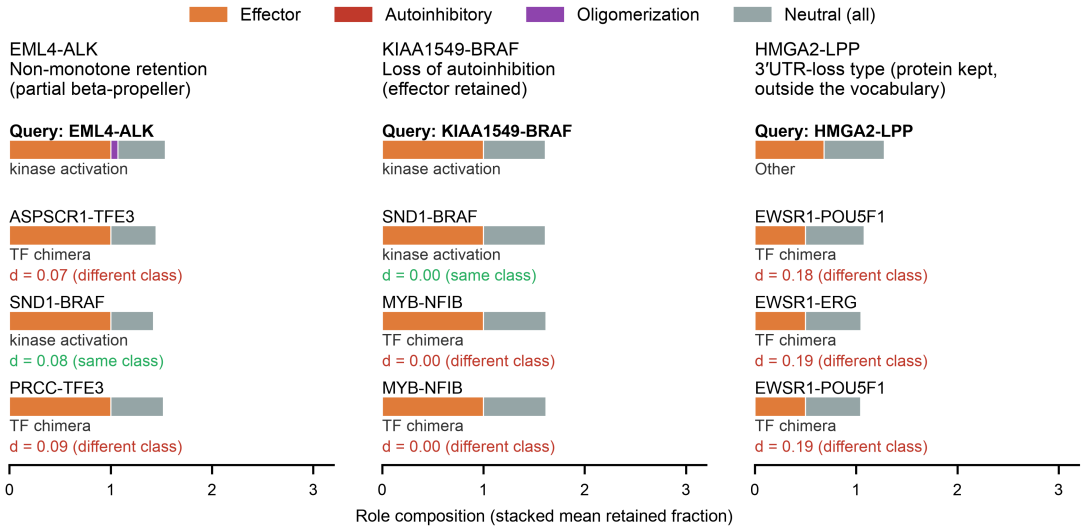


Fig. 2. Precedent search for three query junctions. The heading of each panel names the query and the type of case it illustrates. Each row is one junction, and the bar shows its role composition: the four segments are the mean retained fraction of the effector (orange), autoinhibitory (red), oligomerization (purple) and neutral (grey) elements, drawn end to end. The first row of each panel, in bold, is the query; the three rows below it are the nearest known junctions returned by the search, ordered by Euclidean distance in that four-dimensional space. Under each row are its mechanism class and its distance d from the query, printed in green when the class matches the query's and in red when it does not; d is rounded to two decimals, so 0.00 means a distance below 0.005. Junctions of the query's own gene pair are excluded from the search, so the neighbours always come from other pairs.

4.7. *A monotone trend exists within pairs, but no information beyond it*

In experiment E-6, adding the retained fraction as an explanatory variable improved the fit significantly ($\chi^2 = 46.8$, $p = 5.0 \times 10^{-6}$), and the first-order monotone term alone was significant ($\chi^2 = 35.3$, 4 df, $p = 4.1 \times 10^{-7}$). Within a pair, the more functional elements are retained, the more likely the junction is to be selected. Splitting the terms by role reaches a smaller statistic than the averaged monotone term at the same degrees of freedom ($\chi^2 = 33.7$ against 35.3); the two models are not nested in one another and each is compared against the same null, so the role split adds no explanatory power over the average.

Information beyond the monotone trend, that is, the non-monotone information carried by the breakpoint being at an intermediate position, was not detected ($\chi^2 = 11.5$, 8 df, $p = 0.17$). This is not for want of instances: 44 junctions across 14 pairs carry partially retained oligomerization elements.

4.8. *The role composition is a pair-level property on the decisive axes*

The variance partition of experiment E-6 (Table 2) asks whether there was within-pair variation to predict at all. The two mechanistically decisive axes, effector retention and autoinhibition loss, are almost constant within a pair. In 51 of 55 pairs (92.7%) the effector shows no difference crossing the functional boundary between junctions of the same pair, and the median within-pair standard deviation is 0.000. Autoinhibition shows no crossing in any of the five testable pairs; with 0 of 5 the exact binomial (Clopper-Pearson) two-sided 95% interval is [0, 0.522], so a true crossing rate as high as 0.522 cannot be rejected and this axis has no power at the present sample size.

The only functional axis that varies within a pair is oligomerization, whose crossing rate of 0.385 exceeds the neutral baseline of 0.149. We confirmed that this comparison is not distorted by sensitivity to junction count. The baseline increases monotonically with junction count, but the junction-count distributions of the three roles almost coincide and direct standardization leaves the crossing rates essentially unchanged (effector 0.073 to 0.074; oligomerization 0.385 to 0.390). The ordering of effector below neutral holds in all four strata, and the negative control gave 0.000 for both effector and autoinhibition.

Table 2. Variance partition of the role composition (67 pairs, 373 junctions).

Role	Pairs testable	Between-pair SD	Median within-pair SD	Crossing rate	Standardized
Effector	55	0.388	0.000	0.073 (4/55)	0.074
Autoinhibitory	5 (no power)	0.000	0.000	0.000 (0/5)	-
Oligomerization	26	0.304	0.011	0.385 (10/26)	0.390
Neutral (baseline)	67	0.245	0.040	0.149 (10/67)	0.149

4.9. *The ceiling of annotation lies on the resource side*

Against the UniProt ceiling (experiment E-7), the effector is at 55 pairs against a ceiling of 55, or 100%, and oligomerization at 26 against 28, or 93% (Table 3). Only autoinhibition is below, at 5 against 8, or 63%, and for the remaining 59 pairs there is no description in UniProt at all. None of ALK, RET, ROS1 or NTRK1 carries a description corresponding to autoinhibition. Juxtamembrane

autoinhibition in receptor tyrosine kinases is established biology, but it does not exist as a machine-readable annotation.

Table 3. Current role annotation against the UniProt ceiling.

Role	Current annotation	UniProt ceiling	Current/ceiling	No description in UniProt
Effector	55 pairs	55 pairs	100%	12 pairs
Oligomerization	26 pairs	28 pairs	93%	39 pairs
Autoinhibitory	5 pairs	8 pairs	63%	59 pairs

Of the three variant pairs reported to differ clinicopathologically, only EML4-ALK differed at the level of roles, and the difference appears on the oligomerization axis, the only axis that varies within pairs (Section 4.8). Variant 1 (E13;A20) keeps part of the EML4 beta-propeller, which leaves the fold incomplete and the protein unstable, whereas variant 3a/b (E6;A20) loses that region entirely and is more stable; the stability difference is reported to track the response to ALK inhibitors [31], and our retained fractions on this axis, 0.429 against 0.071, follow it. The region of EML4 that drives oligomerization is retained in every variant [32], so the difference along this axis reflects how much of the rest of EML4 survives, which bears on the stability of the protein rather than on oligomerization itself. The role-level difference appears in this pair because an element with a non-monotone response is partially retained. In KIAA1549-BRAF the roles on the BRAF side are identical between variants (effector 1.0, autoinhibitory 0.0) and the difference appears only in the retained length on the 5' side (neutral 0.75 against 0.65); in NAB2-STAT6 no functional role is assigned at all (neutral 0.724 against 0.835), because the DNA binding and transcriptional activation of STAT6 have no corresponding concept in the current vocabulary. Both of these pairs are reported to differ in tumour location or clinicopathological phenotype, but neither difference is an established independent predictor of survival or recurrence [33-36] and the vocabulary cannot express either of them.

4.10. *How far each term of the framework is supported*

Table 4 organizes how far each term of the framework is supported by current knowledge and resources. The reasons for lack of support fall into three kinds. Two further terms are stated here rather than in the table: gene-pair-level context is partly implemented (Section 4.3), and the neutral term is a diagnosis rather than a test (Section 4.6).

Table 4. Audit of the model terms.

Term	Verdict	Evidence and caveats	What is needed
First-order term (effector)	Supported	1.35-fold departure from the uniform null (1.21-1.51) (Section 4.4); likelihood-ratio test chi-square 46.8 (Section 4.7)	-
First-order term (autoinhibitory)	Partly supported	Complete loss 1.47-fold (1.41-1.49) (Section 4.4); no power within pairs at five pairs (Section 4.8)	Annotation of autoinhibition; absent from UniProt for 59 pairs (Section 4.9)
Non-monotone term	Tested, not supported	chi-square 11.5, 8 df, p = 0.17; 44 junctions across 14 pairs available (Section 4.7)	A fifth role (structural integrity), or subdivision of the effector
Sign split by role	Tested, not supported	chi-square 33.7 against 35.3, no gain at equal df (Section 4.7)	As above
Context at gene level	Not implemented	0 of 8,131 elements resolved (0.0%); a prototype exists but is not connected	Implementation only; no additional knowledge required

5. Discussion

Three independent analyses (Sections 4.6-4.8) agree: breakpoint variants of the same gene pair differ mainly in the amount of protein retained, not in the role composition. Effector retention and autoinhibition loss are largely fixed within a pair, and oligomerization is the only functional axis that often varies, which is why variants of EML4-ALK can be told apart [31].

The negative result is informative. Mechanism-class identification did not exceed the baseline because the present vocabulary records whether functional roles are retained, but not which specific effector performs the function. The representation therefore preserves quantitative variation in retained protein length more effectively than qualitative differences among effectors. This identifies the principal direction for extending the framework: subdividing broad functional roles, particularly the effector category, into biologically distinct subclasses.

The framework also records where current knowledge stops: autoinhibitory regions in particular remain under-annotated in UniProt, which limits the coverage of role annotation (Section 4.9).

Clinical variant interpretation separates evidence in a similar way: the ACMG/AMP framework assigns direction and strength to individual variants [37], and the present framework applies the same principle to functional elements under structural alteration, including alterations other than fusions, such as protein-truncating variants.

The present study has several limitations. The autoinhibitory axis is supported by only five informative gene pairs, convergence on effector retention may partly reflect ascertainment bias in reported oncogenic fusions, protein coordinates differ between UniProt and Ensembl for a subset of genes, and the evaluation is restricted to recurrent gene pairs. A benchmark against experimentally characterized fusion proteins would need pathogenicity labels at breakpoint resolution, which no current resource carries (Section 4.1); assembling them junction by junction is the work this framework points to.

6. Conclusion

Whether a fusion protein drives cancer depends on which of its parts survive the breakpoint and on the direction in which each surviving part pushes the protein. We wrote that direction down as a context-dependent sign on each element; annotation that records only whether a domain is present is the special case in which every sign is positive. Applied to 67 gene pairs and 373 junctions, the framework returns known fusions with a stated error rate and declines to answer when nothing known is close enough.

It does not identify the mechanism class, and it shows why: variants of one gene pair keep the same effector and the same loss of autoinhibition and differ mainly in how much protein remains, so the class is a property of the gene pair and the breakpoint acts below it. Each term of the framework is marked as supported, rejected, untestable for want of annotation, or not yet built, and the next step this points to is functional labels at junction resolution rather than another predictor.

References

1. C. Murphy and O. Elemento, AGFusion: annotate and visualize gene fusions, https://github.com/murphycj/AGFusion (accessed 2026).

2. P. Kim, H. Tan, J. Liu, H. Lee et al., FusionGDB 2.0: fusion gene annotation updates aided by deep learning, Nucleic Acids Res. 50, D1221 (2022).
3. H. Kumar, L.-Y. Tang, C.-C. Yang and P. Kim, FusionPDB: a knowledgebase of human fusion proteins, Nucleic Acids Res. 52, D1289 (2024).
4. M. Shugay, I. Ortiz de Mendíbil, J. L. Vizmanos et al., Oncofuse: a computational framework for the prediction of the oncogenic potential of gene fusions, Bioinformatics 29, 2539 (2013).
5. F. Abate, S. Zairis, E. Ficarra et al., Pegasus: a comprehensive annotation and prediction tool for detection of driver gene fusions in cancer, BMC Syst. Biol. 8, 97 (2014).
6. M. Lovino, M. Montemurro, V. S. Barrese and E. Ficarra, Identifying the oncogenic potential of gene fusions exploiting miRNAs, J. Biomed. Inform. 129, 104057 (2022).
7. M. Lovino, M. S. Ciaburri, G. Urgese et al., DEEPrior: a deep learning tool for the prioritization of gene fusions, Bioinformatics 36, 3248 (2020).
8. S. Vincoff, S. Goel, K. Kholina et al., FusOn-pLM: a fusion oncoprotein-specific language model via adjusted rate masking, Nat. Commun. 16, 1436 (2025).
9. C.-C. Wu, H. C. Beird, J. Zhang and P. A. Futreal, FusionPathway: prediction of pathways and therapeutic targets associated with gene fusions in cancer, PLoS Comput. Biol. 14, e1006266 (2018).
10. K. Murakami, S.-I. Tago, S. Takishita et al., Pathogenicity prediction of gene fusion in structural variations: a knowledge graph-infused explainable artificial intelligence (XAI) framework, Cancers 16, 1915 (2024).
11. D. Cho, H.-M. Lee, J.-A. Kim et al., Autoinhibited protein database (AiPD): a curated database of autoinhibitory domains and their autoinhibition mechanisms, Database (Oxford) 2024, baae085 (2024).
12. S. Medves and J. B. Demoulin, Tyrosine kinase gene fusions in cancer: translating mechanisms into targeted therapies, J. Cell. Mol. Med. 16, 237 (2012).
13. E. De Braekeleer, N. Douet-Guilbert, D. Rowe et al., ABL1 fusion genes in hematological malignancies: a review, Eur. J. Haematol. 86, 361 (2011).
14. Z. Sondka, N. B. Dhir, D. Carvalho-Silva et al., COSMIC: a curated database of somatic variants and clinical data for cancer, Nucleic Acids Res. 52, D1210 (2024).
15. P. Flicek, M. R. Amode, D. Barrell et al., Ensembl 2014, Nucleic Acids Res. 42, D749 (2014).
16. The UniProt Consortium, UniProt: the Universal Protein Knowledgebase in 2023, Nucleic Acids Res. 51, D523 (2023).
17. C. J. Clopper and E. S. Pearson, The use of confidence or fiducial limits illustrated in the case of the binomial, Biometrika 26, 404 (1934).
18. V. Vovk, A. Gammerman and G. Shafer, Algorithmic Learning in a Random World, 2nd edn. (Springer, Cham, 2022).
19. A. N. Angelopoulos and S. Bates, A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification (2021).
20. D. Chakravarty, J. Gao, S. M. Phillips et al., OncoKB: a precision oncology knowledge base, JCO Precis. Oncol. 2017, PO.17.00011 (2017).
21. M. Griffith, N. C. Spies, K. Krysiak et al., CIViC is a community knowledgebase for expert crowdsourcing the clinical interpretation of variants in cancer, Nat. Genet. 49, 170 (2017).

22. P. Horak, M. Griffith, A. M. Danos et al., Standards for the classification of pathogenicity of somatic variants in cancer (oncogenicity): joint recommendations of the Clinical Genome Resource (ClinGen), Cancer Genomics Consortium (CGC) and Variant Interpretation for Cancer Consortium (VICC), Genet. Med. 24, 986 (2022).
23. F. Weinberg, R. Griffin, M. Fröhlich et al., Identification and characterization of a BRAF fusion oncoprotein with retained autoinhibitory domains, Oncogene 39, 814 (2020).
24. A. N. Van, M. T. Kunkel, T. R. Baffi et al., Protein kinase C fusion proteins are paradoxically loss of function in cancer, J. Biol. Chem. 296, 100445 (2021).
25. K. M. Smith, R. Yacobi and R. A. Van Etten, Autoinhibition of Bcr-Abl through its SH3 domain, Mol. Cell 12, 27 (2003).
26. B. Nagar, O. Hantschel, M. A. Young et al., Structural basis for the autoinhibition of c-Abl tyrosine kinase, Cell 112, 859 (2003).
27. M. W. Richards, E. W. Law, L. P. Rennalls et al., Crystal structure of EML1 reveals the basis for Hsp90 dependence of oncogenic EML4-ALK by disruption of an atypical beta-propeller domain, Proc. Natl. Acad. Sci. U.S.A. 111, 5195 (2014).
28. J. A. Bridge, M. Kanamori, Z. Ma et al., Fusion of the ALK gene to the clathrin heavy chain gene, CLTC, in inflammatory myofibroblastic tumor, Am. J. Pathol. 159, 411 (2001).
29. P. Argani, M. Y. Lui, J. Couturier et al., A novel CLTC-TFE3 gene fusion in pediatric renal adenocarcinoma with t(X;17)(p11.2;q23), Oncogene 22, 5374 (2003).
30. C. Mayr, M. T. Hemann and D. P. Bartel, Disrupting the pairing between let-7 and Hmga2 enhances oncogenic transformation, Science 315, 1576 (2007).
31. C. G. Woo, S. Seo, S. W. Kim et al., Differential protein stability and clinical responses of EML4-ALK fusion variants to various ALK inhibitors in advanced ALK-rearranged non-small cell lung cancer, Ann. Oncol. 28, 791 (2017).
32. M. W. Richards, L. O'Regan, D. Roth et al., Microtubule association of EML proteins and the EML4-ALK variant 3 oncoprotein require an N-terminal trimerization domain, Biochem. J. 467, 529 (2015).
33. H. Kurani, M. Gurav, O. Shetty et al., Pilocytic astrocytomas: BRAF V600E and BRAF fusion expression patterns in pediatric and adult age groups, Childs Nerv. Syst. 35, 1525 (2019).
34. C. Faulkner, H. P. Ellis, A. Shaw et al., BRAF fusion analysis in pilocytic astrocytomas: KIAA1549-BRAF 15-9 fusions are more frequent in the midline than within the cerebellum, J. Neuropathol. Exp. Neurol. 74, 867 (2015).
35. S. Barthelmess, H. Geddert, C. Boltze et al., Solitary fibrous tumors/hemangiopericytomas with different variants of the NAB2-STAT6 gene fusion are characterized by specific histomorphology and distinct clinicopathological features, Am. J. Pathol. 184, 1209 (2014).
36. K. Akaike, A. Kurisaki-Arakawa, K. Hara et al., Distinct clinicopathological features of NAB2-STAT6 fusion gene variants in solitary fibrous tumor with emphasis on the acquisition of highly malignant potential, Hum. Pathol. 46, 347 (2015).
37. S. V. Tavtigian, M. S. Greenblatt, S. M. Harrison et al., Modeling the ACMG/AMP variant classification guidelines as a Bayesian classification framework, Genet. Med. 20, 1054 (2018).

RNA-Protein Interaction Prediction Using Hybrid Quantum-Classical Fusion

A K M Muhitul Islam* and Gopalakrishnan Subramanian*
Department of Computer Science, The University of Iowa, Iowa City, IA 52242, USA

Rahul Singh†
Department of Computer Science, The University of Iowa, Iowa City, IA 52242, USA
†*Corresponding Author, Email: rahul-singh@uiowa.edu*

**These authors contributed equally to this work.*

Predicting whether a non-coding RNA and a protein physically interact is central to understanding post-transcriptional gene regulation and for screening candidate RNA sequences against a target protein. Existing predictive methods combine protein and RNA embeddings via classical operations—dot products, concatenations, or attention mechanisms—that induce different forms of interaction between the representations. We describe an approach that encodes fixed-weight protein (ESM-2) and RNA (RNA-FM) embeddings in two separate qubit registers of a quantum circuit and then entangles the qubits between those registers. This construct provides a coupling mechanism for functional inference beyond homology. An eight-qubit (four qubits per register), eight-layer circuit produces 48 cross-register XX, YY and ZZ expectations for a nonlinear classifier and exceeds its classical counterpart based on several measures (AUROC 0.792 vs. 0.760) despite having no trainable parameters of its own. We benchmarked this variant against established RNA–protein interaction prediction methods under the same 5-fold, family-and-clan-disjoint protocol on RNAInterAct. With encoders matched for RPIembeddor and the dual- and cross-encoder comparisons, the proposed method achieved 0.792 AUROC, compared with RPIembeddor (0.768), a dual-encoder (0.775) and a cross-encoder (0.781). These experiments provide preliminary evidence that an entangling circuit is a viable approach for combining biological embeddings. Ablation studies support the usefulness of cross-register correlators and nonlinear readout, whereas adding qubits or circuit depth did not consistently improve performance.

Keywords: RNA-Protein Interaction Prediction; Quantum Machine Learning; Hybrid Quantum-Classical Models; Quantum Circuits; Entanglement; Parameter-Matched Ablation; Foundation Model Embeddings.

1. Introduction

RNA-protein interactions are the physical binding events through which RNA-binding proteins recognize specific RNA sequences and structures to regulate splicing, localization, stability, and translation [1]. Experimentally assaying the combinatorial space of candidate pairs is infeasible, so computational predictions can be particularly useful to prioritize candidates for follow-up experimental validation. The recent advent of biological foundation models has changed how sequence data is encoded into features for downstream prediction tasks. Protein language models such as ESM-2, trained on hundreds of millions of evolutionary sequences, capture contextual

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

biochemical and structural signal without requiring a resolved structure. RNA language models such as RNA-FM provide analogous representations for non-coding RNA. Such models produce a fixed-length embedding for an input sequence. For our problem, given a protein embedding and an RNA embedding, the task lies in determining how they could be combined so as to predict whether a pair of protein-RNA molecules interact.

1.1 Related Work

A survey of experimental methods used for solving this problem can be found in [1]. Computational methods that have been proposed to address this problem can be classified into a few categories: (i) *Sequence- and structure-based*: classifiers score pairs typically using hand-crafted features, such as RPISeq's RNA 4-mer and protein 3-mer (7-letter-alphabet) composition vectors [2], or RPI-Pred's 132-feature vector combining RNA secondary-structure-element counts (stem, hairpin, loop, bulge) with protein secondary-structure composition [3]. (ii) *Network-guided deep learning*: incorporates known interaction-network topology [4]. (iii) *Dual-encoder methods*: score pairs by similarity in a shared embedding space, as in RPIPLM [5]. (iv) *Cross-encoder methods*: concatenate or cross-attend between embeddings, as in CORAL [6]. (v) *Fine-tuned transformer systems* attend over- or fine-tune embeddings directly, as in AptaTrans [7] and RNAInterAct [8].

The mechanisms employed to integrate information in each of the above methods have limitations. For instance, a dot product reduces the rich information in the embeddings to a single number (alignment score). Concatenation, on the other hand, does not impose any structure at all, leaving inference of the relationship entirely to downstream processing. Cross-attention forms explicit pairwise scores between the two sets of embeddings. But the coupling it applies is itself parameterized: these scores are computed through learned projections and depend not just on the data but also the parametrization of the learning machine. Quantum machine learning offers a different way to couple the two embeddings. Rather than combining embeddings with a predefined mapping (*e.g*. the dot product), an unconstrained operator (*e.g*. concatenation), or a learned operator (*e.g.* cross-attention [8]), each embedding can be used to set the rotation angles of a small quantum circuit—one register of qubits per molecule and then entangle the two registers together. Entanglement gives us a mechanism to co-involve the information present in the two embeddings within a single quantum state, guaranteeing joint dependence on both molecules by construction (Section 1.2 defines this mechanism concretely). Prior work on data re-uploading [9] and quantum-enhanced feature spaces [10, 11] motivates the use of entangling circuits to construct features for classification. The novelty of our work lies in applying this approach to protein–RNA interaction prediction, using a quantum circuit to fuse information from different biological embeddings.

1.2 Functional Inference Beyond Homology

Computational inference of RNA–protein molecular function increasingly requires methods that move beyond homology-based similarity: many RNA-binding proteins recognize short or structurally mediated motifs, and many functional RNA elements exhibit convergent protein-binding behavior without clear sequence similarity. Consequently, using homology as the only explanatory mechanism is limiting for understanding the diversity of RNA–protein binding events.

Foundation models address part of this gap by embedding sequences into representations that capture contextual signals that may go beyond explicit homology. However, the fusion step by which two such embeddings are combined has limitations as discussed earlier.

Our method uses quantum entanglement as a principled, parameter-free alternative to address this problem: once two registers are entangled, any measurement of any single qubit cannot be attributed to one molecule alone. Thus, joint dependence is enforced in a manner that is different than what may be learned from the data using parametrized learning machines. A qubit is the quantum analogue of a classical bit, and a register is simply the small set of qubits containing a molecule's embedding (four qubits per molecule in our implementation; Section 2.2). Two registers become entangled when a gate such as the cross-register CNOT (Section 2.2) acts on a superposition input, producing a joint state that cannot be factored in terms of the product of single-register states; a consequence is that the measurement outcomes of the two registers become statistically correlated.

1.3 Formulation of the Prediction Problem

Let P denote a protein sequence and R an RNA sequence, drawn from the spaces of valid protein and RNA sequences respectively. Let $y(P,R) \in \{0,1\}$ indicate whether the pair physically interacts, with $y = 1$ denoting a true interaction and $y = 0$ a non-interacting pair. Given a labeled dataset of n protein-RNA pairs with known interaction status, the goal is to learn a scoring function f that maps a protein-RNA pair to an estimate $\hat{y} = f(P,R) \in [0,1]$ of the interaction probability, thresholded to obtain a binary prediction. For example, given a specific RNA-binding protein P and a pool of candidate RNA sequences $R_1, \ldots, R_k$, ranking each candidate by $\hat{y}(P, R_k)$ is precisely how such a model would be used to prioritize sequences for experimental follow-up. Different machine learning or statistical methods may be used to construct f based on P and R. We describe the construction of a small quantum circuit that forms the core of a quantum machine learning method to construct f. The quantum circuit consists of two registers, one seeded by the RNA embedding and one by the protein embedding, and entangled by cross-register CNOT gates, such that every measured qubit reflects information from both molecules.

2. Methods

The inputs to our method are a protein sequence and an RNA sequence. By the term "embedding" hereafter, we imply the fixed-length, real-valued vector that each fixed, pretrained encoder produces for the corresponding sequence (Section 2.1 details both encoders). Given two such embeddings, our research investigated two questions: (1) whether routing the fusion step through an entangling quantum feature map produced a better interaction predictor than a (same-sized) classical network and (2) how the aforementioned approach compared with established protein-RNA interaction prediction methods. Answering this requires a classical model matched to the quantum model in every respect but the fusion mechanism, which is a central part of this study.

2.1. Fixed Weight Foundation-Model Encoders

Both molecules were encoded using fixed weight, pretrained foundation models and neither of the encoders was fine-tuned by us. Protein sequences were encoded with a 150-million-parameter ESM-

2 model [12], mean-pooled over the sequence to a 640-dimensional vector h_P. RNA sequences were encoded with RNA-FM [13], a transformer language model pretrained on 23.7 million non-coding RNA sequences, mean-pooled to a 640-dimensional vector h_R. Both encoders operated on their native alphabets, so no nucleotide remapping was required. Every unique sequence in the dataset was encoded once, and the resulting vector was cached to disk, so training and evaluation only ever read pre-computed embeddings. This ensured that the fusion mechanism was sole variable under study. Equation (1) summarizes both encoders.

$$h_P = MeanPool\big(ESM2(P)\big) \in \mathbb{R}^{640}, h_R = MeanPool\big(RNA - FM(R)\big) \in \mathbb{R}^{640} \quad (1)$$

In addition to ensuring that the fusion mechanism was the sole variable under study, two reasons drove our decision to not fine-tune the encoders. First, by not fine tuning, every unique sequence could be embedded once and cached, so the shared-encoder comparisons involved the same input representations. Fine-tuning, conversely, would have let the encoders adapt differently under each fusion mechanism. Second, by not fine-tuning the encoders, we were able to keep the parameter-matched ablations computationally tractable, since re-encoding the dataset for every configuration would have increased the training cost substantially.

2.2. Quantum-Classical Fusion Architecture

The quantum-fusion head, which we hereafter call Qfusion, turned the pair (h_R, h_P) into a binding-probability logit in the following four stages.

(1) Projection:- Each embedding was projected by a dedicated two-layer MLP projector into a vector of raw angle features, one feature per (wire, rotation axis, circuit layer) triple assigned to that molecule's register. With 4 qubits per register, 8 re-uploading layers, and 3 encoding axes (RX, RY, RZ), each projector output 4 × 8 × 3 = 96 features (Eq. 2). Each projector used 128 hidden units with GELU, input and output layer normalization, and dropout 0.1 before each linear layer. This configuration was selected, from a qubit-count, re-uploading-depth and readout ablation evaluated under the same 5-fold protocol, as the strongest-performing quantum variant; the full ablations are reported in Sections 3.5 and 3.6.

$$a = f_R(h_R) \in \mathbb{R}^{96}, p = f_P(h_P) \in \mathbb{R}^{96} \quad (2)$$

(2) Angle scaling:- The raw features $x = [a \| p] \in \mathbb{R}^{192}$ were mapped to bounded rotation angles by a trainable, elementwise affine-tanh transform. This kept every angle inside $(-\pi, \pi)$ and let the model learn how sharply each feature modulated its rotation (Eq. 3).

$$\theta = \pi \tanh(w \odot x + b) \quad (3)$$

$$u_{l,k,q} = \begin{cases} a_{l,k,q}, & 1 \le q \le 4 \\ p_{l,k,q-4}, & 5 \le q \le 8 \end{cases}$$

In words: every angle was driven by exactly one tower — the RNA projector supplied the angles on wires 1 to 4 and the protein projector supplied the angles on wires 5 to 8, at every layer *l* and on every rotation axis *k*.

(3) Entangled circuit:- The 192 angles drove an eight-qubit circuit, four wires for the RNA register and four for the protein register, ran for eight data re-uploading layers on a statevector simulator. Within each layer: (*i*) every wire received one RX, one RY, and one RZ rotation, each

driven by a different re-uploaded angle from θ; (ii) an intra-register ring of CZ gates entangled the four RNA wires among themselves, and separately entangled the four protein wires among themselves; (iii) four cross-register CNOT gates coupled RNA wire *i* to protein wire *i, for i = 1...4.* (A statevector simulator computed the circuit's exact quantum state—a vector of $2^8 = 256$ complex amplitudes for our eight-qubit circuit—on ordinary classical hardware, rather than executing the circuit on physical quantum hardware.) Step (iii) was the fusion step. The cross-register CNOT gates coupled the RNA and protein registers enabling entanglement and allowing joint information to be accessed through cross-register measurements. All representational learning occurred in the projectors, in the angle-scaling parameters w and b, and in the readout layer described next. The input-dependent circuit rotations changed through their learned encoding angles.

We numbered the eight wires $q = 1,\ldots,8$, with wires 1 to 4 forming the RNA register and wires 5 to 8 the protein register, Fig. 1 depicts the entangled circuit including the pairing of RNA wire *i* with protein wire *i*. $L = 8$ is the number of re-uploading layers and $K = 3$ is the number of encoding axes. Throughout, $\sigma 1 = X$, $\sigma 2 = Y$, and $\sigma 3 = Z$ denote the Pauli matrices, and $R\sigma(t) = \exp(-it\sigma/2)$ denotes the rotation of a single qubit by angle t about axis σ. Equation (4) defines the full block S_l.

$$S_l(\theta) = \left[\bigotimes_{q=1}^{8} R_Z\left(\theta_{l,3,q}\right)\right]\left[\bigotimes_{q=1}^{8} R_Y\left(\theta_{l,2,q}\right)\right]\left[\bigotimes_{q=1}^{8} R_X\left(\theta_{l,1,q}\right)\right], \quad W = W_{\times}(W_R \otimes W_P) \tag{4}$$

Operators acted right to left, so each wire received its RX rotation first, then RY, then RZ. Within one axis the eight rotations acted on distinct wires and commuted. The entangler W was built from an intra-register CZ ring on each register, W_R over wires 1 to 4 and W_p over wires 5 to 8, followed by the four cross-register gates $W\times$ = CNOT(1→5) CNOT(2→6) CNOT(3→7) CNOT(4→8). The CZ gates were diagonal and so commuted among themselves, and W_R and W_p acted on disjoint registers, so $W\times$ was the only term that coupled the two molecules.

$$|\psi(\theta)\rangle = W\, S_L(\theta)\, W\, S_{L-1}(\theta) \cdots W\, S_1(\theta)\, |0\rangle^{\otimes 8} \tag{5}$$

The circuit started from $|0\rangle\otimes 8$ and alternated an encoding block with the entangler for $L = 8$ layers.

(4) *Readout:-* After eight layers we calculated XX, YY and ZZ expectations for every cross-register pair, $i, j = 1...4$. Each axis contributed sixteen correlators, giving forty-eight values in [−1, 1] (Eq. 6), concatenated in XX, YY, ZZ order. A 48-to-32-to-1 classifier with GELU and dropout 0.1 after the hidden activation mapped this vector to a scalar logit; a sigmoid gave the interaction score (Eq. 7). No single-wire values entered this readout. In Eq. (6), $A \in \{X,Y,Z\}$; $D_{0.1}$ in Eq. (7) was dropout during training and the identity at inference. The pair observables exposed joint measurement features to the classifier; Section 3.6 compared ZZ-only and XX/YY/ZZ features with linear and nonlinear classifiers. (An expectation value was the average outcome obtained if the same measurement were repeated many times; "XX" denoted jointly measuring two qubits along the Pauli-X axis, and similarly for YY and ZZ—the three standard ways of reading out a qubit pair's correlated state.)

$$z_{ij}^{(A)} = \langle\psi(\theta)|A_i A_{4+j}|\psi(\theta)\rangle \tag{6}$$

$$\hat{y} = \sigma(w_2^T D_{0.1}[GELU(W_1 z + b_1)] + b_2) \tag{7}$$

The full circuit used 288 gates (64 RX, 64 RY, 64 RZ, 64 CZ, 32 CNOT) at a circuit depth of 64. The model's 193,793 trainable parameters lived entirely in the two projectors, the angle-scaling

vectors, and the readout network (1,601 parameters). Training used binary cross-entropy between the predicted probability $\hat{y}$ and the ground-truth label y (Eq. 8).

$$L = -[ylog(\hat{y}) + (1-y)log(1-\hat{y})] \quad (8)$$

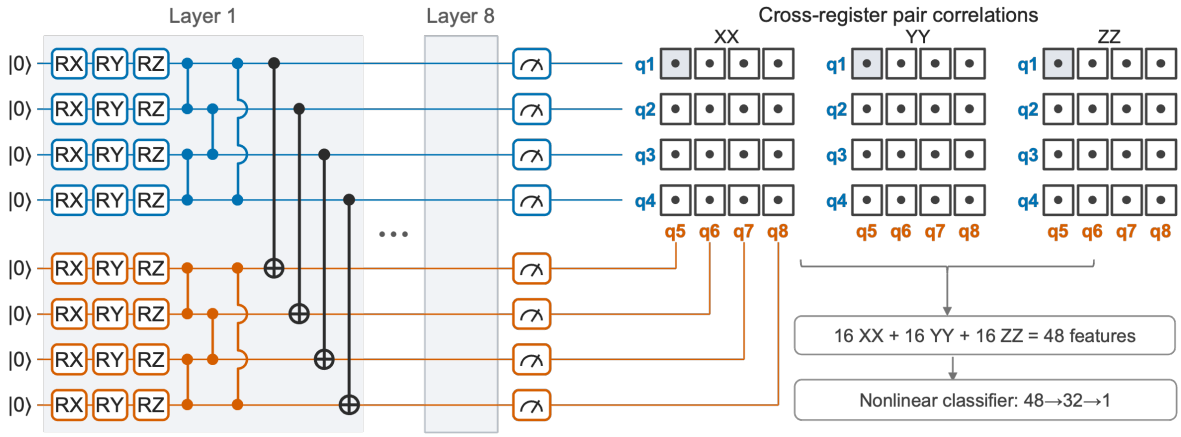


Fig. 1. Schematic representation of the Qfusion entangling circuit, showing one of the eight layers and the four-wire RNA and protein registers. RX, RY, and RZ gates encode the per-tower angle features; intra-register CZ gates form a ring within each register; cross-register CNOT gates couple each RNA control wire to the corresponding protein target wire. The final state yields 16 XX, 16 YY and 16 ZZ pair expectations. These 48 features enter a 48→32→1 nonlinear classifier with GELU and training-time dropout 0.1; a sigmoid converts the output logit to an interaction *score*.

2.3. Parameter-Matched Classical Control

In the first experiment we compared the proposed architecture with a standard deep learning architecture (SDLA) consisting of two projector architectures, producing angle vectors a and p of the same dimension as above. These were passed through $tanh$ and scaled by π without the learned affine transform in Eq. (3), but instead of entering a quantum circuit, the concatenated 192-dimensional vector was fed to a small classical fusion network. This was a 192-to-8-to-1 multilayer perceptron with a GELU nonlinearity, whose output was the logit. The width of this fusion layer set the control's parameter count; we fixed it by selecting the width whose total landed close to Qfusion's, which gave eight hidden units.

The SDLA had 193,361 trainable parameters, within 0.22% of Qfusion's 193,793. The models used the same pretrained embeddings and projector dimensions, with a classical fusion MLP in the control and a fixed entangling circuit followed by a nonlinear readout in Qfusion. This comparison evaluated the complete fusion architectures; matching parameter counts alone did not equate learning capacity or entanglement.

2.4. Baselines Retrained Under the 5-Fold Protocol

We additionally trained and evaluated two classical fusion baselines with frozen encoders on the same embeddings, data, and 5-fold protocol. The dual-encoder ("Dual-encoder" in Table 1) projected each tower into a 256-dimensional embedding, L2-normalized both, and scored the pair by cosine similarity times a learned temperature, in the style of contrastive dual-tower models such

as RPIPLM [5]. The cross-encoder ("Cross-encoder") concatenated the two pooled embeddings and passed them through a three-layer MLP to a scalar logit.

Three further methods, whose performance had previously been known only from externally reported values, were retrained within this study on the same 5-folds. RPIembeddor [8], the reference model released with RNAInterAct, used the same fixed RNA-FM and ESM-2 150M embeddings as Qfusion, with cross-attention and convolutional fusion. The comparison in Table 1 was therefore encoder-matched with model-specific downstream architectures and optimization. AptaTrans [7] was evaluated in two variants. These were fixed weight pretrained encoders with only the prediction stage trained ("AptaTrans-fixed weight"), and full end-to-end fine-tuning ("AptaTrans-retrained"). All seven methods in Table 1 shared the same evaluation partitions, with their stated model-specific training settings.

2.5. Protocol for Training and Testing

We evaluated all methods on the RNAInterAct benchmark [8], restricted to a family-and-clan-disjoint partition and extended to 5-fold cross-validation. RNA families and protein clans were each independently split into 5 groups and for fold i, the test set contained every pair whose RNA family and protein clan both fell in group i, the validation set used group $(i + 1) \bmod 5$, and the remaining three groups formed training. This guaranteed that every test RNA family and every test protein clan were entirely absent from that fold's training and validation data. Checkpoints were selected using each fold's disjoint validation group. This reduced annotated family and clan overlap but did not exclude all evolutionary signal in the embeddings. Across the 5 folds, training sets ranged from 34,758–35,500 pairs, and validation and test sets each ranged from 3,833–4,017 pairs, for 19,606 unique test pairs evaluated in total. Positive-pair prevalence varied more than fold size (23.3–59.9% across folds), since it depended on which families and clans happened to co-occur. There was zero sequence overlap between any fold's training, validation, and test sets for either molecule.

All trainable models were optimized with AdamW [14] and a cosine-annealing learning-rate schedule [15], using binary cross-entropy loss, and trained independently on each of the 5 folds. Qfusion and its four pair-readout variants used learning rate 0.001, weight decay 0.01, batch size 256, projector dropout 0.1, up to 25 epochs and early-stopping patience 10. The cosine schedule had a 100-epoch horizon. The retained baseline and circuit-size experiments used up to 100 epochs with patience 30, or up to 40 epochs with patience 12 for AptaTrans. We reported mean and standard deviation across folds for every method, rather than a single training seed.

Each fold provides its own family-and-clan-disjoint validation partition, so early stopping and checkpoint selection use that fold's validation AUROC. All four pair-readout configurations were completed and their checkpoints fixed before test evaluation. Architecture comparisons are exploratory because they use the same reported benchmark; one fold's validation group is another fold's test group.

For every model we reported AUROC and AUPRC across all thresholds, and at the 0.5 threshold F1, precision, recall, accuracy, balanced accuracy, and MCC, plus the Brier score for calibration. For Qfusion against its classical control, we ran a paired t-test, with a Wilcoxon signed-rank test as a distribution-free check, across the 5-fold-level values of each measure, and a McNemar test on paired 0.5-threshold decisions, both pooled across folds and separately within each fold, since

pooling can manufacture apparent significance from one unrepresentative fold. Full holistic measures required raw per-pair predictions, which we retained for the four models trained through the shared fusion pipeline: Qfusion, its classical control, and the dual- and cross-encoder baselines. Our circuit ran on a classical statevector simulator at a scale that is trivial to simulate using standard hardware, and we claimed no runtime or sample-complexity quantum advantage.

3. Results

3.1. Overall Predictive Performance

Table 1 reports mean AUROC, AUPRC, and F1 (± SD across the 5 folds) for all seven methods, evaluated on the same family-and-clan-disjoint partitions. The primary comparison of this study was Qfusion, the eight-qubit, eight-layer configuration examined in the case studies in Sections 3.3–3.6, against established RNA-protein interaction methods. Under encoder-matched conditions Qfusion reached 0.792 mean AUROC, ahead of RPIembeddor (0.768) and the fixed weight dual-encoder (0.775) and numerically above the cross-encoder (0.781). A second, narrower comparison, against a classical fusion network matched to Qfusion's own encoders and parameter count, was examined in Section 3.3, where Qfusion exceeded that matched control (0.760 mean AUROC) on every measure evaluated. Section 3.3 examines the uncertainty of these differences. AptaTrans trailed the field under both training regimes tested, possibly reflecting differences from its narrower aptamer-protein setting (Section 4). The three stronger baselines were all designed for this same task, and all were evaluated here on the same fixed RNA-FM and ESM-2 150M embeddings that Qfusion used. Holding the encoders fixed across methods controlled their input representations, and under that constraint the entangling fusion head was competitive with the strongest of these baselines.

Table 1. Comparative evaluation across RNA-protein interaction prediction methods on the RNAInterAct family-and-clan-disjoint 5-fold cross-validation (5 folds; training 34,758–35,500 pairs, validation and test 3,833–4,017 pairs per fold). All seven rows were evaluated on these partitions with the model-specific training settings in Section 2.5; values are mean ± SD across the 5 folds. The proposed method (Qfusion) is listed last and shown in bold.

Method	AUROC	AUPRC	F1
RPIembeddor	0.768 ± 0.036	0.656 ± 0.070	0.482 ± 0.079
Cross-encoder	0.781 ± 0.030	0.682 ± 0.115	0.603 ± 0.105
Dual-encoder	0.775 ± 0.023	0.676 ± 0.117	0.617 ± 0.087
Classical control (matched)	0.760 ± 0.041	0.650 ± 0.100	0.601 ± 0.037
AptaTrans (retrained)	0.608 ± 0.034	0.492 ± 0.118	0.537 ± 0.068
AptaTrans (fixed weight)	0.538 ± 0.032	0.446 ± 0.148	0.403 ± 0.178
Quantum-fusion (Qfusion)	**0.792 ± 0.038**	**0.695 ± 0.090**	**0.642 ± 0.043**

3.2. Holistic, Beyond-AUROC Assessment

Table 2 and Figure 2 report the full evaluation suite for the four models trained through the shared fusion pipeline: Qfusion, its classical control, and the dual- and cross-encoder baselines, all of which retained raw per-pair predictions. Qfusion nominally led its classical control on every measure in Table 2, using an entangling circuit that contributed zero trainable parameters of its own, against a learned classical fusion network. At the 0.5 decision threshold Qfusion led its classical control on F1 (0.642 vs. 0.601), recall (0.633 vs. 0.575), MCC (0.404 vs. 0.353) and balanced accuracy (0.705 vs. 0.677). Qfusion also had higher mean F1 and MCC than the dual- and cross-encoder baselines, while the cross-encoder retained higher precision and a lower Brier score. Returning to the classical-control comparison that motivated Section 3.3, these margins were modest relative to the fold-to-fold standard deviations reported in Table 2 (±0.023 to ±0.119 depending on the measure). What made them worth reporting was that the direction of the effect was consistent across every one of the eight measures evaluated, rather than favorable on some and unfavorable on others. Section 3.3 examines matched-fold uncertainty.

3.3. Isolating the Quantum Contribution: Statistical Robustness

We compared Qfusion with the classical control using the same encoders and projector dimensions and closely matched trainable parameter counts (193,793 vs. 193,361, within 0.22%). The architectural differences are specified in Section 2.3. This comparison complemented the established-method benchmarks in Section 3.1. We used paired tests across the five matched evaluation folds because their training sets overlapped, the resulting p-values were interpreted as nominal summaries.

Quantum fusion was ahead of its classical control on both ranking measures in all 5 folds. The AUROC differences ranged from 0.0016 to 0.0511. In its best-performing fold (fold 4), the AUROC value for Qfusion was 0.848 and comparisons with other methods used the matched-fold differences rather than their separate best-fold scores. A paired t-test gave a mean AUROC difference of +0.0316 ($p = 0.027$) and a mean AUPRC difference of +0.0452 ($p = 0.068$). The Wilcoxon signed-rank test gave $p = 0.063$ for each ranking measure. Mean F1 increased by 0.0410 (paired $p = 0.002$); recall, MCC and balanced accuracy also had nominal paired $p < 0.05$, whereas precision and Brier score did not.

At a fixed 0.5 decision threshold, the pooled McNemar test involved 19,606 pairs. Qfusion correctly classified 2,026 pairs that its classical control misclassified, versus 1,552 pairs where the classical control was correct and Qfusion was not (continuity-corrected statistic 62.53; nominal $p = 2.6 \times 10^{-15}$). Within each of the five folds, more discordant pairs favored Qfusion over the classical control; four folds had nominal $p < 0.05$, while fold 2 did not. Shared molecules and biological groups limited the independence assumption of the pooled test. It assessed thresholded errors, not AUROC or an entanglement-specific effect.

Overall, Qfusion's point estimates favored it on every reported measure against this control. The AUROC paired t-test and thresholded comparisons provided encouraging evidence, although the rank-based test and the dependence between folds limited the strength of inference. Against the cross-encoder, the mean AUROC difference was +0.0106 (paired $p = 0.414$), indicating that the entangling-circuit mechanism led to comparable or slightly improved performance.

3.4. Quantum Circuit Resource Profile

Table 3 summarizes the circuit's resource footprint, for assessing the cost of the quantum approach. The circuit used 8 qubits, a width that could be easily simulated using classical hardware. This configuration was identified through the circuit and readout comparisons in Sections 3.5–3.6. We presented it as an architectural ablation, not a quantum-hardware demonstration. At eight qubits the simulated state had 256 amplitudes; the XX/YY/ZZ nonlinear runs took 16.6–22.3 minutes per fold on an RTX A6000-based machine.

Table 2. Full holistic measures at the 0.5 probability threshold, mean ± SD across the 5 family-and-clan-disjoint folds.

Model	AUROC	AUPRC	F1	Prec.	Recall	MCC	Bal. Acc.	Brier
Quantum-fusion	**0.792 ±0.038**	**0.695 ±0.090**	**0.642 ±0.043**	**0.665 ±0.095**	**0.633 ±0.071**	**0.404 ±0.054**	**0.705 ±0.036**	**0.220 ±0.045**
Classical control	0.760 ±0.041	0.650 ±0.100	0.601 ±0.037	0.646 ±0.109	0.575 ±0.059	0.353 ±0.051	0.677 ±0.031	0.244 ±0.058
Dual-encoder	0.775 ±0.023	0.676 ±0.117	0.617 ±0.087	0.660 ±0.119	0.585 ±0.080	0.381 ±0.042	0.690 ±0.028	0.230 ±0.030
Cross-encoder	0.781 ±0.030	0.682 ±0.115	0.603 ±0.105	0.677 ±0.103	0.549 ±0.118	0.375 ±0.040	0.680 ±0.028	0.211 ±0.037

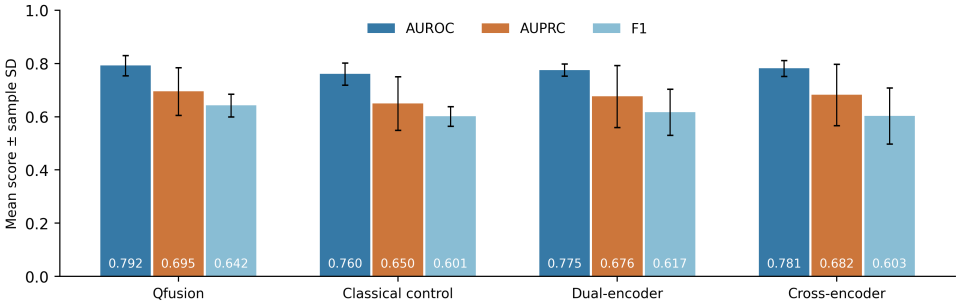


Fig. 2. AUROC, AUPRC, and F1 (mean ± SD, error bars) for the four models across the 5 family-and-clan-disjoint folds.

Table 3. Quantum circuit resource summary.

Property	Value
Qubits	8 (4 RNA register + 4 protein register)
Data re-uploading layers	8
Encoding axes per wire per layer	3 (RX, RY, RZ)
Readout observables	48 (16 XX + 16 YY + 16 ZZ cross-register expectations)
Total gates	288 (64 RX, 64 RY, 64 RZ, 64 CZ, 32 CNOT)
Circuit depth	64
Simulator	PennyLane default.qubit, exact statevector simulation
Qfusion trainable parameters (total)	193,793

Property	Value
Classical-control trainable parameters (total)	193,361
Parameter match	within 0.22%

3.5. Qubit-Count and Re-uploading-Depth Ablation

Six circuit configurations were trained and evaluated under the identical 5-fold protocol, spanning 2 to 12 qubits and 6 to 256 re-uploading layers per register, all using the single-wire (xyz) readout only. With a fixed readout type, changing width and depth also changed projector dimensions and parameter counts. The results from this experiment are in Table 4. Under a single-wire readout the quantum head plateaued near AUROC 0.76 across a wide range of widths (Table 4). The deepest configurations had lower mean AUROC, as low as 0.707, despite using up to three times the default parameter count. Neither added width nor added depth lifted the single-wire readout past its plateau, which motivated the readout ablation in Section 3.6. None of the six configurations in Table 4 outperformed the classical control from Table 1 (0.760 ± 0.041); every one of them measured single-wire observables only. Section 3.6 examines cross-register correlators and classifier form as alternative ways to improve the readout.

Table 4. Qubit-count / re-uploading-depth ablation for Qfusion, mean ± SD AUROC across the 5 family-and-clan-disjoint folds. "Params" was the total trainable parameter count of the full model (projectors, angle-scaling, and readout). All six configurations used the single-wire (xyz) readout only, without the cross-register correlators of Section 3.6.

Configuration	Qubits	Layers	Params	AUROC
6q, 8 layers	6 (3+3)	8 (sym.)	185,827	0.759 ± 0.041
4q, 128 layers	4 (2+2)	128 (sym.)	370,957	0.759 ± 0.014
12q, 6 layers	12 (6+6)	6 (sym.)	195,421	0.757 ± 0.051
4q, 256 layers	4 (2+2)	256 (sym.)	575,245	0.746 ± 0.028
2q, 160 layers	2 (1+1)	160 (sym.)	294,343	0.738 ± 0.040
2q, 256 layers	2 (1+1)	256 (sym.)	370,951	0.707 ± 0.066

3.6. Readout Ablation: Cross-Register Correlators

In Table 5 we present results from changing the measurement readout within the circuit family. The twelve-qubit circuit measured 36 single-wire observables, twice as many as the six-qubit circuit, and did no better (AUROC 0.757 versus 0.759). Adding nine cross-register ZZ correlators to the same six-qubit, eight-layer circuit raised mean AUROC to 0.767 at a cost of nine readout parameters. The eight-qubit linear classifier with 24 single-wire and 16 ZZ features reached 0.780. Because it also changed register size, this comparison did not isolate the effect of correlators at eight qubits.

Table 6 compares pair-only readouts for the same eight-qubit, eight-layer circuit under fixed training settings. ZZ features gave AUROC 0.774 with a linear classifier and 0.782 with a nonlinear

classifier; XX/YY/ZZ features gave 0.781 and 0.792, respectively. Thus richer pair measurements and a nonlinear classifier both improved the mean, with the combined design also giving the highest AUPRC (0.695) and F1 (0.642).

Table 5. Readout ablation for Qfusion, mean ± SD AUROC across the 5 family-and-clan-disjoint folds. "Single-wire obs." counted the per-wire Pauli-X/Y/Z expectation values and "Cross-register obs." the two-qubit Z_iZ_j correlators between registers. All configurations used the same five evaluation partitions and the linear readout.

Configuration	Qubits	Single-wire obs.	Cross-register obs.	Params	AUROC
6q, 8 layers, xyz	6 (3+3)	18	0	185,827	0.759 ± 0.041
12q, 6 layers, xyz	12 (6+6)	36	0	195,421	0.757 ± 0.051
6q, 8 layers, xyz+zz	6 (3+3)	18	9	185,836	0.767 ± 0.033
8q, 8 layers, xyz+zz	8 (4+4)	24	16	192,233	0.780 ± 0.041

Table 6. Pair-readout comparison for the same eight-qubit, eight-layer circuit and fixed training settings (Section 2.5). All readouts omitted single-wire features. Values are mean ± sample SD over the five folds. Nonlinear denotes a 32-unit GELU classifier with dropout 0.1.

Pair features	Classifier	AUROC	AUPRC	F1
ZZ (16)	Linear	0.774 ± 0.026	0.667 ± 0.103	0.622 ± 0.083
XX/YY/ZZ (48)	Linear	0.781 ± 0.036	0.674 ± 0.092	0.636 ± 0.068
ZZ (16)	Nonlinear	0.782 ± 0.036	0.675 ± 0.089	0.627 ± 0.070
XX/YY/ZZ (48)	**Nonlinear**	**0.792 ± 0.038**	**0.695 ± 0.090**	**0.642 ± 0.043**

4. Discussion and Conclusion

We have described a hybrid quantum-classical fusion model in which fixed weight protein and RNA embeddings were projected into an eight-qubit circuit with cross-register entangling gates. Its mean AUROC of 0.792 across disjoint RNA families and protein clans exceeded the parameter-matched control and was competitive with established methods. The ablations favored richer pair measurements and nonlinear readout over increasing circuit width or depth. Qfusion used quantum state evolution and Pauli observables, evaluated by classical statevector simulation. Standard attention as in RPIembeddor instead forms compatibility-weighted combinations of value vectors from queries and keys [8, 16]. Both represent input interactions; Qfusion constructed its joint state with unitary rotations and fixed CZ/CNOT gates.

These results suggest that an entangling coupling mechanism can serve as a practical substitute for a learned fusion operator when predicting RNA–protein interactions across families and clans absent from training. The next step is to test whether this advantage holds on larger, independently curated interaction sets and extends to screening candidate RNA sequences against a fixed protein target, the use case this method is intended to support.

Acknowledgments

This work was supported in part by the National Science Foundation grants #IIS-1817239 and #IIS-3401096.

References

1. M. Ramanathan, D. F. Porter, P. A. Khavari, Methods to study RNA-protein interactions, Nat. Methods 16(3), 225-234 (2019).
2. U. K. Muppirala, V. G. Honavar, D. Dobbs, Predicting RNA-protein interactions using only sequence information, BMC Bioinformatics 12, 489 (2011).
3. V. Suresh, L. Liu, D. Adjeroh, X. Zhou, RPI-Pred: predicting ncRNA-protein interaction using sequence and structural information, Nucleic Acids Res. 43(3), 1370-1379 (2015).
4. H. Liu, Y. Jian, C. Zeng, Y. Zhao, RNA-protein interaction prediction using network-guided deep learning, Commun. Biol. 8(1), 247 (2025).
5. Y. Liu, T. Bao, P. Yin, S. Wang, Y. Wang, RPIPLM: Prediction of ncRNA-protein interaction by post-training a dual-tower pretrained biological model with supervised contrastive learning, PLoS ONE 20(8), e0329174 (2025).
6. M. Catalano, G. Pepe, G. Ausiello, C. McWhite, G. Gambosi, M. Helmer Citterich, P. F. Gherardini, Cross-Attention Over RNA and Protein Sequences Enables Generalizable Interaction Prediction, bioRxiv 2026.04.22.720174 (2026).
7. I. Shin, K. Kang, J. Kim, S. Sel, J. Choi, J.-W. Lee, H. Y. Kang, G. Song, AptaTrans: a deep neural network for predicting aptamer-protein interaction using pretrained encoders, BMC Bioinformatics 24, 447 (2023).
8. D. Matus, F. Runge, J. K. Franke, L. Gerne, M. Uhl, F. Hutter, R. Backofen, RNA-Protein Interaction Classification via Sequence Embeddings, bioRxiv 2024.11.08.622607 (2024).
9. A. Perez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, J. I. Latorre, Data re-uploading for a universal quantum classifier, Quantum 4, 226 (2020).
10. V. Havlicek, A. D. Corcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature 567(7747), 209-212 (2019).
11. M. Schuld, N. Killoran, Quantum machine learning in feature Hilbert spaces, Phys. Rev. Lett. 122(4), 040504 (2019).
12. Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, A. Rives, Evolutionary-scale prediction of atomic-level protein structure with a language model, Science 379(6637), 1123-1130 (2023).
13. J. Chen, Z. Hu, S. Sun, Q. Tan, Y. Wang, Q. Yu, L. Zong, L. Hong, J. Xiao, T. Shen, I. King, Y. Li, Interpretable RNA foundation model from unannotated data for highly accurate RNA structure and function predictions, arXiv:2204.00300 (2022).
14. I. Loshchilov, F. Hutter, Decoupled Weight Decay Regularization, International Conference on Learning Representations (2019), arXiv:1711.05101.
15. I. Loshchilov, F. Hutter, SGDR: Stochastic Gradient Descent with Warm Restarts, arXiv:1608.03983 (2016).
16. A. Vaswani et al., Attention Is All You Need, Advances in Neural Information Processing Systems 30 (2017), arXiv:1706.03762.

Homology-Based Variant-Effect Predictors Break Down on Cytochrome P450 Pharmacogenes*

Helen Xu[1], Issah Samori[2], Gowri Nayar[3], Russ B. Altman[1,2,3,4]

[1]*School of Medicine, Stanford University, Stanford, CA, USA*
[2]*Department of Bioengineering, Stanford University, Stanford, CA, USA*
[3]*Department of Biomedical Data Science, Stanford University, Stanford, CA, USA*
[4]*Department of Genetics, Stanford University, Stanford, CA, USA*

E-mail: russ.altman@stanford.edu

Keywords: Cytochrome P450; variant-effect prediction; pharmacogenomics; protein language models; deep mutational scanning.

Cytochrome P450 (CYP) enzymes metabolize roughly three-quarters of clinically used drugs, and genetic variation in these enzymes is a leading source of interindividual differences in drug response. Predicting a variant's functional effect is therefore critical, yet the consequences of most CYP variants remain unknown. Many state-of-the-art variant-effect predictors rest on a homology-based paradigm that scores variants by evolutionary conservation — an assumption that pharmacogenes, including CYPs, violate. Focusing on human CYPs, we show that homology-based models fail systematically. AlphaMissense (AM) assigns variants to its "ambiguous" class at nearly twice the proteome-wide rate across six clinically important CYPs, and within that class its scores are essentially uninformative about measured CYP2C9 deep mutational scan (DMS) activity ($\rho = -0.130$). Using catalytic activity and cellular abundance from the same CYP2C9 DMS, we show that AM is largely an abundance signal: it tracks activity well for destabilizing, low-abundance variants ($\rho = -0.568$) but far more weakly for abundance-independent variants ($\rho = -0.357$; Fisher r-to-z $z = -8.99$, $p < 10^{-18}$). We further hypothesized that non-homology-based features (sequence position, substitution chemistry, distance to the substrate-recognition sites, and secondary structure) might account for where AM errs, but their explanatory power is weak. Using CYP2C9 DMS activity as ground truth, we built a k-nearest-neighbors model over ESM-2 embeddings and ensembled it with AM and ESM-2 masked marginal probability, raising the correlation between predicted and measured activity within the ambiguous class from $\rho = 0.080$ for AM alone to 0.399, and reaching an overall $\rho = 0.721$ that exceeds every out-of-the-box predictor we tested ($\rho = 0.523$ to 0.686). However, neither this improved ensemble model nor any other method agrees significantly with clinical annotations. Drawing on evidence that a variant's effect can depend on the drug, we hypothesize that substrate identity is the key missing feature in current models, and that predicting function for these multi-substrate enzymes may require redefining function as substrate-conditioned.

*Code and supplementary materials are available at `https://github.com/hhelenxu/cyp-variant-effect-prediction`.

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

1. Introduction

Adverse drug reactions are among the leading causes of death in hospitalized patients. Much of this harm is not idiosyncratic but genetic: interindividual differences in the enzymes that metabolize drugs mean that the same dose can be therapeutic for one patient and fatal for another.[1,2] Chief among these enzymes are the cytochrome P450s (CYPs), which are involved in the metabolism of roughly three-quarters of all clinically used drugs.[3–5] Genetic variation in CYP genes is correspondingly one of the largest sources of interindividual variability in drug response.[6,7] CYP2C9 alone metabolizes up to ∼15% of small-molecule drugs, including the widely prescribed anticoagulant warfarin, for which variant-driven differences in metabolism directly affect safety and efficacy.[8] Pharmacogenomic testing and genotype-guided prescribing aim to pre-empt these outcomes,[9,10] but in practice they rest on a small set of well-characterized "star" alleles. The functional consequences of the vast majority of observed CYP variants remain unknown, so we generally cannot predict whether a given variant will increase, decrease, or leave unchanged the metabolism of a specific drug. Closing this prediction gap is a prerequisite for scaling pharmacogenomics beyond its current handful of actionable alleles.

The number of possible variants vastly exceeds what deep-mutational, biochemical, or clinical characterization can hope to interrogate, so filling this gap necessarily requires computational variant-effect prediction. Yet most widely used predictors annotate a variant by its similarity to sequences shaped by evolution — the same homology-based paradigm that computational function annotation has relied on for decades. SIFT aligns related sequences and scores a substitution by how often the mutant residue is tolerated at that position;[11] AlphaMissense (AM) couples AlphaFold with multiple-sequence-alignment (MSA) input and human and primate allele frequencies to predict pathogenicity against ClinVar benchmarks;[12,13] protein language models of the ESM family absorb the same signal by masked-language modeling over divergent sequences, whether repurposed zero-shot through masked marginal probabilities (MMP), as in ESM-2,[14,15] or purpose-built for variant-effect prediction, as in ESM-1v.[16] Three further model families build on this idea: REVEL is a supervised ensemble that combines conservation- and structure-derived component scores against pathogenicity labels;[17] DeMaSk is fitted to a substitution matrix derived from deep mutational scans and is therefore trained on measured function rather than on conservation alone;[18] and PharmGScore is tuned specifically for pharmacogene variants, the setting where generic predictors are known to degrade.[19]

We use *homology-based* throughout for predictors whose variant-effect signal derives, explicitly or implicitly, from the distribution of residues observed at a position across related sequences — retrieved at inference by alignment, as in SIFT and AM; absorbed during pretraining, as in ESM-2; aggregated from component scores that are themselves alignment-derived, as in REVEL; or supplied as population allele frequencies, as in other related models. This definition is deliberately broad, and it includes the protein-specific model we build here.

The premise that evolutionary tolerance is a good proxy for functional consequence works well for the disease genes these models are usually tuned on: such genes are under strong evolutionary pressure to stay unchanged, so a variant that breaks conservation is likely to be harmful. Yet pharmacogenes violate that assumption. CYPs evolved largely to metabolize compounds in the diet and environment, and their central role in drug metabolism is in a sense

incidental to that history.[20,21] As a result, reduced or even absent activity of an individual CYP is frequently benign on its own, with consequences that surface only when a particular food or drug is encountered. CYP variants are therefore usually not disease-causing; they frequently occur at high population frequencies; and they arise in a gene family that is unusually polymorphic and functionally diverse rather than strongly conserved.[4,6] Consistent with this mismatch, homology- and pathogenicity-oriented predictors perform markedly worse on pharmacogenetic than on disease variants, motivating a growing body of pharmacogene-specific methods and evaluations.[19,22–25] What has remained unclear is whether this degradation reflects mere low model confidence or a genuine information ceiling, in which an important determinant of function is missing and cannot be recovered by better calibration alone.

Among pharmacogenes, CYPs are one of the more tractable settings in which to examine this question, combining the violated assumptions with clinical necessity and a rich, though still incomplete, body of ground truth. Decades of research have produced variant-level data spanning high-throughput deep mutational scanning (DMS) of enzymatic activity and abundance,[8,26] clinical annotations, and curated pharmacogenomic databases, yet the functional consequences of most CYP variants are still unresolved. Because CYPs are comparatively well characterized, cases where homology-based predictors fail more plausibly expose a limit of the homology paradigm than an absence of data.

In this work, we use human CYP pharmacogenes as a case study for where and why homology-based variant-effect prediction breaks down. Our contributions are:

(1) We show that AM assigns CYP variants to the "ambiguous" class at nearly twice the proteome-wide rate across six CYPs, and that on CYP2C9 — the one CYP with paired activity and abundance DMS — its activity signal is abundance-based and weak on the abundance-independent, catalytic axis, particularly within the ambiguous class, with structural and biochemical priors accounting for almost none of the residual error.
(2) We show that a k-nearest-neighbors model over ESM-2 embeddings, ensembled with AM and ESM-2 MMP, outperforms generic, ensemble, function-trained, and pharmacogene-tuned predictors on CYP2C9 activity, yet this accuracy does not translate into reliable agreement with clinical or pharmacogenetic labels.
(3) We propose substrate identity as the leading candidate for the missing, non-homologous variable, drawing on pharmacogenetic evidence that a single variant's consequence can depend on the drug, and argue that substrate-conditioned prediction is a promising route beyond homology.

Taken together, these results reframe the annotation problem for pharmacogenes: the barrier is not simply that homology-based similarity is too weak, but that the definition of function these models predict — a single substrate-agnostic effect — is the wrong target for multi-substrate enzymes. Moving beyond homology here may therefore require redefining function itself, as a quantity conditioned on substrate, rather than more or better similarity search.

2. Methods

2.1. *Datasets and variants*

We drew on four ground-truth resources, each encoding a distinct notion of variant effect. DMS activity measurements were taken from Amorosi et al. (2021),[8] which reports click-

seq (enzymatic activity, $N = 6{,}273$) and vamp-seq (cellular abundance, $N = 6{,}559$) scores for CYP2C9 missense variants, with $n = 4{,}478$ overlapping variants. We used the normalized click-seq activity score as the primary quantitative target throughout and the abundance readout for the abundance-stratification analysis (Section 2.3.2). ClinVar provided disease-oriented clinical significance labels following the ACMG framework;[27,28] after filtering for CYP2C9 single-missense variants with a functional classification, 50 variants at 42 unique positions remained. PharmVar provided pharmacogenetic function classes;[29] we used the 77 CYP2C9 star alleles defined by a single missense substitution with functional classification. ClinPGx (PharmGKB) provided literature-curated variant–drug annotations for the substrate-resolved analysis (Section 2.3.5).[30]

The enrichment analysis (Section 2.3.1) and the substrate-resolved analysis (Section 2.3.5) spanned six clinically important CYPs (CYP1A2, CYP2C19, CYP2C9, CYP2D6, CYP2E1, and CYP3A4) whereas activity-benchmarked modeling focused on CYP2C9, for which comprehensive DMS data were available. Reference protein sequences for all CYPs were obtained from UniProt.[31] All analyses were done in Python 3.12.11 on the Sherlock computing cluster operated by Stanford Research Computing.

2.2. *Variant-effect predictors and models*

2.2.1. *Predictors and model selection*

We adapted three predictors (AM, ESM-2, and KNN) for CYP2C9 activity prediction. For CYP2C9 (UniProt P11712) and the five additional CYPs, we obtained AM pathogenicity scores together with AM's discrete class labels (likely benign, ambiguous, or likely pathogenic), which AM assigns by thresholding its predicted pathogenicity score. ESM-2 (`esm2_t33_650M_UR50D`) was used as a zero-shot variant-effect signal via its MMP ($\text{MMP} = \log P(x_i = \text{mut} \mid x_{\setminus i}) - \log P(x_i = \text{wt} \mid x_{\setminus i})$), and as protein embeddings for the KNN model (Section 2.2.2). We selected ESM-2 over ESM-1v for the main analyses because ESM-2's zero-shot scores tracked CYP2C9 DMS activity more closely (Fig. S1).

Because AM and ESM-2 MMP scores are not on the activity scale, we calibrated each to click-seq activity using monotonic regression. We compared isotonic and logistic calibrators by 5-fold out-of-fold (OOF) R^2 and selected the better-performing option. For AM this was isotonic regression; the monotonicity constraint encodes the expected biological direction, in which higher pathogenicity corresponds to lower activity. For ESM-2 MMP it was a four-parameter monotone logistic function, constrained to be monotone increasing so that higher MMP corresponds to higher activity. Calibrators were fit within each training fold only.

We evaluated four external baselines, scored through the same evaluation pipeline: DeMaSk,[18] PharmGScore,[19] SIFT,[11] and REVEL.[17] PharmGScore values were taken from a precomputed release and mapped to the Ensembl canonical CYP2C9 transcript (ENST00000260682) by CDS coordinate; SIFT and REVEL scores came from dbNSFP v4.8.[32]

To avoid data leakage, no predictor was evaluated on data to which it was fit. SIFT is unsupervised, and DeMaSk, though trained on DMS data, predates the CYP2C9 scan; both were therefore included in the activity and the clinical-alignment analyses. REVEL and PharmGScore were trained on pathogenicity annotations that include ClinVar labels, so both are reported for activity prediction but excluded from the clinical-alignment comparison.

2.2.2. *CYP-specific KNN over ESM-2 embeddings*

We built a KNN regressor predicting click-seq activity from ESM-2 embeddings. Each mutated CYP2C9 protein, also referred to as a variant, was represented by the final-layer, per-residue ESM-2 representation, pooled over all sequence positions (excluding the `BOS`/`EOS` tokens) to a 1,280-dimensional vector.

All cross-validation in this work was grouped by residue position (`GroupKFold`) to prevent leakage from near-duplicate observations on both sides of a split, so that no variant is ever predicted by a model that has seen another substitution at its own residue. The scheme is nested: an outer 5-fold position-grouped split for evaluation; within each outer training partition, an inner 4-fold position-grouped split that generates the base-learner meta-features for the ensemble; and, for KNN hyperparameter selection, a 3-fold position-grouped split within each outer training partition that scores every configuration in the grid. The hyperparameter configuration selected most often across outer folds (mean pooling, Euclidean distance, distance-weighted averaging, $K = 30$) was used in all subsequent analyses (Fig. S2). For the clinical comparisons (Section 2.3.4), ClinVar and PharmVar variants were excluded from model training at the position level.

2.2.3. *Ensemble model*

We combined the calibrated AM activity, calibrated ESM-2 MMP activity, and KNN-predicted activity using a ridge-regression meta-learner on standardized features (`StandardScaler` followed by `RidgeCV` over $\alpha \in [10^{-3}, 10^{3}]$). All models were evaluated by outer 5-fold cross-validation, and we report OOF Spearman's ρ against click-seq activity.

2.3. *Statistical analyses*

2.3.1. *Ambiguous-class enrichment across CYPs*

For each of the six CYPs we retrieved all possible missense variants and their AM classes, and compared the fraction assigned to the ambiguous class against the proteome-wide baseline reported by the original AM paper.[12] Significance was assessed with a one-sided binomial test.

2.3.2. *Abundance stratification*

We split variants at the median abundance into a low-abundance (destabilizing) and a normal/high-abundance (abundance-independent) subset, and compared the model's score–activity Spearman correlation between strata, both across the full dataset and within each AM class. Because abundance is itself correlated with predictor score, a difference between strata could in principle reflect a narrower score range in one of them; we therefore verified that score range, interquartile range (IQR), and standard deviation were matched between the low- and high-abundance strata. Differences between strata were tested with an independent-samples Fisher r-to-z test.

2.3.3. *Structural and biochemical priors*

We asked whether four interpretable features explain AM's prediction error, defined as the absolute residual between its calibrated activity prediction and the measured DMS activity. **(1)**

Sequence position of the point mutation, correlated with error by two-sided Spearman. **(2)** *Substitution chemistry*: error compared across wild-type residue groups (nonpolar GAVLIMPFW, polar STCYNQ, positively charged RKH, negatively charged DE) by Kruskal–Wallis test with η^2 as the effect size. **(3)** *Substrate-recognition-site proximity*: minimum $C\alpha$ distance from each variant's mutated residue to any residue of the six substrate-recognition site (SRS) regions, with SRS boundaries taken from the universal selectivity-determining positions[33] and mapped onto CYP2C9's AlphaFold structure[13] using SecStrAnnotator secondary-structure annotations,[34] correlated with error by Spearman. **(4)** *Secondary structure*: structure elements assigned with SecStrAnnotator,[34] with error compared across types by Kruskal–Wallis test.

2.3.4. *Clinical and pharmacogenetic benchmarking*

We compared model-predicted activity against the held-out ClinVar and PharmVar labels (Section 2.1), which were excluded from training, using Spearman's ρ between predicted activity and the ordinal function labels, where higher values denote greater retained function. PharmVar comparisons were additionally stratified by evidence level (definitive; definitive + moderate; all known-function alleles). As a baseline reference, we also correlated measured DMS activity directly against the same labels. Note the limited available data: only six CYP2C9 missense variants both carry a non-uncertain ClinVar label and are present in the DMS, and PharmVar evidence tiers range from $n = 9$ (definitive) to $n = 27$ (all known function) alleles.

2.3.5. *Substrate-resolved analysis*

ClinPGx variant–drug annotations for all six CYPs were deduplicated to the best-evidence entry per (variant, drug) pair. A variant can exhibit substrate dependence only if it has been annotated against more than one drug, so the discordance rate was computed over the subset of variants carrying at least two drug annotations. A variant was called discordant if its function label was not identical across its drugs.

3. Results

3.1. *AM fails systematically on CYPs*

AM assigns CYP missense variants to its ambiguous pathogenicity class far more often than it does across the human proteome as a whole. Pooling all possible missense variants across the six CYPs (N=56,677 variants), 18.16% were classified as ambiguous, versus a proteome-wide baseline of 10.28% reported by AM[12] — a 1.77-fold enrichment that is highly significant by a one-sided binomial test ($p < 10^{-16}$) and consistent across all six CYPs examined (Fig. 1).

The correlation between AM's pathogenicity score and measured activity is strong overall but uneven across classes: moderate within the likely benign ($\rho = -0.354$) and likely pathogenic ($\rho = -0.423$) classes and weak within the ambiguous class ($\rho = -0.130$) (Fig. S3). These correlations are negative because higher predicted pathogenicity tracks lower activity. To ask what AM is missing, particularly in the ambiguous class, we split the CYP2C9 DMS variants at the median abundance into destabilizing (low-abundance) and abundance-independent (normal/high-abundance) subsets, which reveals an abundance bias (Fig. 2). Across the full

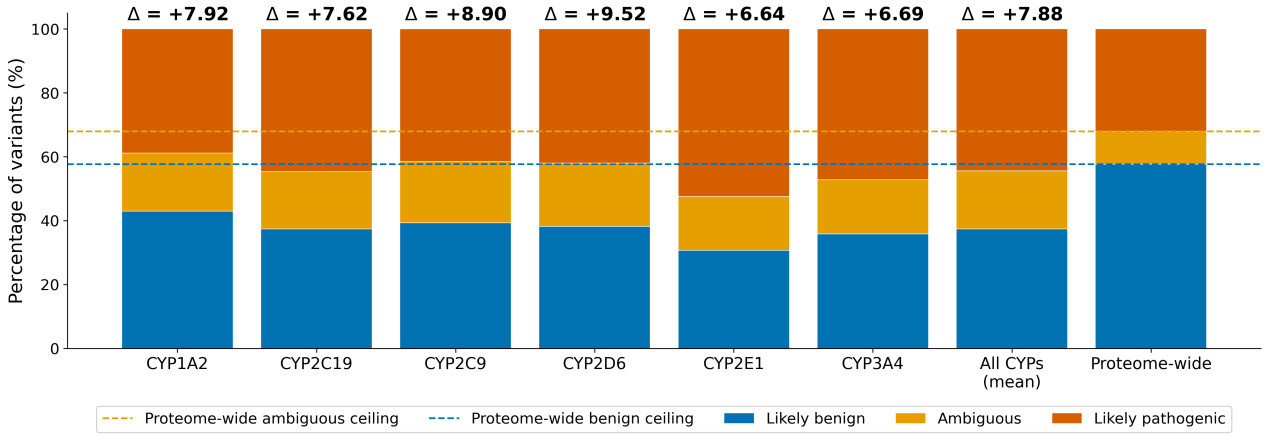


Fig. 1. Rate of AM ambiguous pathogenicity labels in CYPs versus the proteome-wide baseline.[12] Each Δ above a bar is the difference (in percentage points) between that CYP's ambiguous fraction and the proteome-wide ambiguous fraction. Dashed lines mark the proteome-wide ambiguous and likely benign reference rates.

dataset, AM's correlation with activity is strong for destabilizing variants ($\rho = -0.568$) but substantially weaker for abundance-independent variants ($\rho = -0.357$), a highly significant difference by an independent-samples Fisher r-to-z test ($z = -8.99$, $p < 10^{-18}$). The two strata are matched on score range, IQR, and standard deviation, so the difference is not a range-restriction effect. Within the ambiguous class the gap is even more pronounced: AM retains signal for destabilizing ambiguous variants ($\rho = -0.186$, $p = 1.5 \times 10^{-4}$) but marginally at best for abundance-independent ones ($\rho = -0.059$, $p = 0.234$). The collapse is specific to this class; in the benign and pathogenic classes both abundance strata remain significant (ρ from -0.343 to -0.251, all $p < 10^{-14}$). Consistent with this, AM's score tracks cellular abundance directly, correlating with DMS abundance at $\rho = -0.598$ (Fig. S4), nearly as strongly as with activity itself ($\rho = -0.643$), despite AM being neither trained on abundance nor given it as input. AM therefore predicts the protein abundance component of variant effect well and the abundance-independent, catalytic component only weakly, pointing to a missing axis of activity independent of protein level.

3.2. *Biological priors only weakly explain AM prediction error*

If AM's failure had a structural or biochemical basis, biologically interpretable features should account for where its predictions go wrong. We tested four such features against AM's absolute prediction error across the full dataset ($N = 6{,}141$). None explain the error well (Fig. S5).

Sequence position is unrelated to error ($\rho = -0.002$, $p = 0.858$; Fig. S5A). Substitution chemistry and secondary-structure type both reach significance on the full dataset ($H = 19.954$, $p = 1.7 \times 10^{-4}$; and $H = 13.029$, $p = 0.023$; Fig. S5B, D), albeit with very small effect sizes ($\eta^2 = 0.003$ and 0.001, respectively). Each accounts for under 0.5% of the variance in prediction error, so significance here largely reflects statistical power at this sample size rather than a substantive effect.

Another structural feature with a defensible mechanistic rationale is proximity to the substrate-recognition sites, since CYP substrate selectivity is largely determined by the six SRS regions. SRS distance is the strongest of the four priors and is highly significant ($\rho = -0.091$, $p = 8.6 \times 10^{-13}$), with variants nearer the SRS regions carrying somewhat higher error (Fig. S5C). It nonetheless accounts for under 1% of the variance in error ($\rho^2 \approx 0.008$).

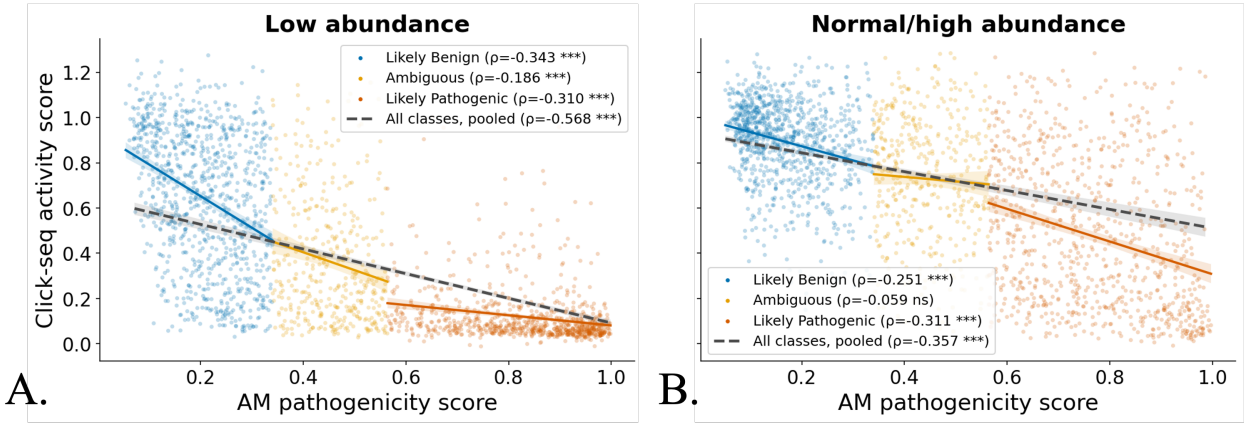


Fig. 2. AM pathogenicity score versus CYP2C9 DMS click-seq activity, stratified by variant abundance (low abundance N = 2,210, **A**; normal/high-abundance N = 2,211, **B**). Each point is one missense variant, colored by AM class (likely benign, ambiguous, likely pathogenic); solid lines are ordinary least-squares fits within each class and the dashed line is the fit across the entire dataset, both with 95% confidence bands. Two-sided Spearman's ρ is annotated per class and for the pooled data, where $^{***}p < 0.001$; ns, not significant. Per-class variant counts are n = 952, 410, and 848 (likely benign, ambiguous, likely pathogenic) in (**A**) and n = 953, 410, and 848 in (**B**).

3.3. *A CYP-specific ensemble is the most accurate activity predictor*

Although AM and ESM-2 are strongly correlated overall ($\rho = -0.797$), their agreement is far weaker within the ambiguous class ($\rho = -0.223$) than within the benign ($\rho = -0.488$) or pathogenic ($\rho = -0.615$) classes (Fig. S3C). The two predictors thus diverge precisely where AM is least informative. Because AM and ESM-2 carry partly independent signal, most notably in the ambiguous class, we reasoned that combining the ESM-2 representation under protein-specific supervision with AM's calibrated predictions could recover the catalytic axis AM misses. On a footing-matched comparison across an identical set of variants (n = 2,040, limited by a variant's single-nucleotide reachability and dbNSFP coverage), a CYP2C9-specific KNN over ESM-2 embeddings, ensembled with calibrated AM and ESM-2 MMP, is the most accurate activity predictor we evaluate (overall $\rho = 0.721$; Table 1), exceeding AM (0.650) and ESM-2 (0.686) as well as every external baseline: the pharmacogene-tuned PharmGScore (0.604), the alignment-based SIFT (0.584), the supervised ensemble REVEL (0.539), and the function-trained DeMaSk (0.523). Because we did not test these differences for significance, the ensemble is nominally rather than demonstrably the best overall predictor.

The interpretable gain is concentrated where AM is weakest. Within the ambiguous class, AM alone reaches only $\rho = 0.080$, and adding it to ESM-2 changes nothing (0.341 for AM + ESM-2, identical to ESM-2 alone at 0.341) — AM contributes no usable ranking information there. The KNN alone reaches 0.353 and the full ensemble 0.399, a gain of +0.319 over AM alone; in comparison, the full ensemble improves on AM by +0.230 in the benign class and +0.098 in the pathogenic class, so the ambiguous class is where the ensemble changes most relative to AM's own starting point (Table S1). These per-class correlations are computed on the full DMS set rather than on the footing-matched subset, since the external baselines are not defined for all variants; overall values therefore differ slightly from Table 1.

Table 1. Predictive performance against CYP2C9 DMS activity: Spearman's ρ between predicted and measured activity (out-of-fold, position-grouped 5-fold cross-validation), on a footing-matched set of $n = 2{,}040$ variants scored by all nine models. The protein-specific ensemble is the most accurate and exceeds every external baseline on equal footing.

Model	Spearman ρ	Paradigm
AM + ESM-2 + KNN	**0.721**	protein-specific ensemble
AM + ESM-2	0.713	ensemble
ESM-2	0.686	protein language model (zero-shot)
AM	0.650	homology / pathogenicity
KNN (ESM-2 embeddings)	0.625	protein-specific
PharmGScore	0.604	pharmacogene-tuned
SIFT	0.584	alignment-based conservation
REVEL	0.539	supervised ensemble
DeMaSk	0.523	function-trained (DMS)

3.4. *Predicted activity does not align with clinical or pharmacogenetic annotations*

Recovering biochemical activity does not translate into reliable agreement with CYP2C9 clinical or pharmacogenetic function labels from ClinVar and PharmVar, respectively (Fig. 3A–B). Both label sets are ordinally encoded from no function to normal function, so agreement would appear as a positive correlation. Against ClinVar, no method reaches significance and the largest nominal correlations point the wrong way (ρ from -0.219 to -0.538, with corresponding p from 0.637 to 0.206; Fig. 3C). The same pattern holds against PharmVar under a high evidence bar: restricted to definitive-evidence alleles, every method is non-significant (ρ from -0.137 [SIFT] to -0.365 [KNN]), and admitting moderate evidence does not change this (Fig. 3D). Only when the threshold is relaxed to all alleles with any known-function label do several methods turn positive and significant (AM $\rho = 0.505$, $p = 0.004$; ESM-2 $\rho = 0.399$, $p = 0.039$; AM + ESM-2 $\rho = 0.498$, $p = 0.008$; AM + ESM-2 + KNN $\rho = 0.450$, $p = 0.018$; DeMaSk $\rho = 0.436$, $p = 0.016$). That tier adds the limited-evidence alleles and therefore the noisiest labels, and the effect does not survive at either stricter threshold, so we read it as weak evidence rather than recovered agreement.

Even measured DMS click-seq activity reaches no significant agreement at any threshold (ClinVar $\rho = -0.676$, $p = 0.140$; PharmVar definitive $\rho = 0.408$, $p = 0.363$; all known function $\rho = 0.399$, $p = 0.059$). Every method we evaluate hits the same ceiling, and so does the experimentally measured assay itself. The limit is therefore not that some implementations read homology poorly while others perform better; the best use of homology we can construct sits at the ceiling with the rest. What thus fails is the shared target: a mismatch between what these methods predict and what the clinical labels encode.

These correlations rest on very few labeled variants, so individual annotations carry large leverage. CYP2C9*3 (I359L) is predicted to retain 71–91% of wild-type activity by every model, diverging not only from its "No Function" PharmVar label (assigned at the highest tier of evidence) but also from its measured DMS activity of 45%. L208V, ClinVar's sole CYP2C9 variant carrying a "pathogenic" clinical-significance assertion, shows the opposite pattern: its

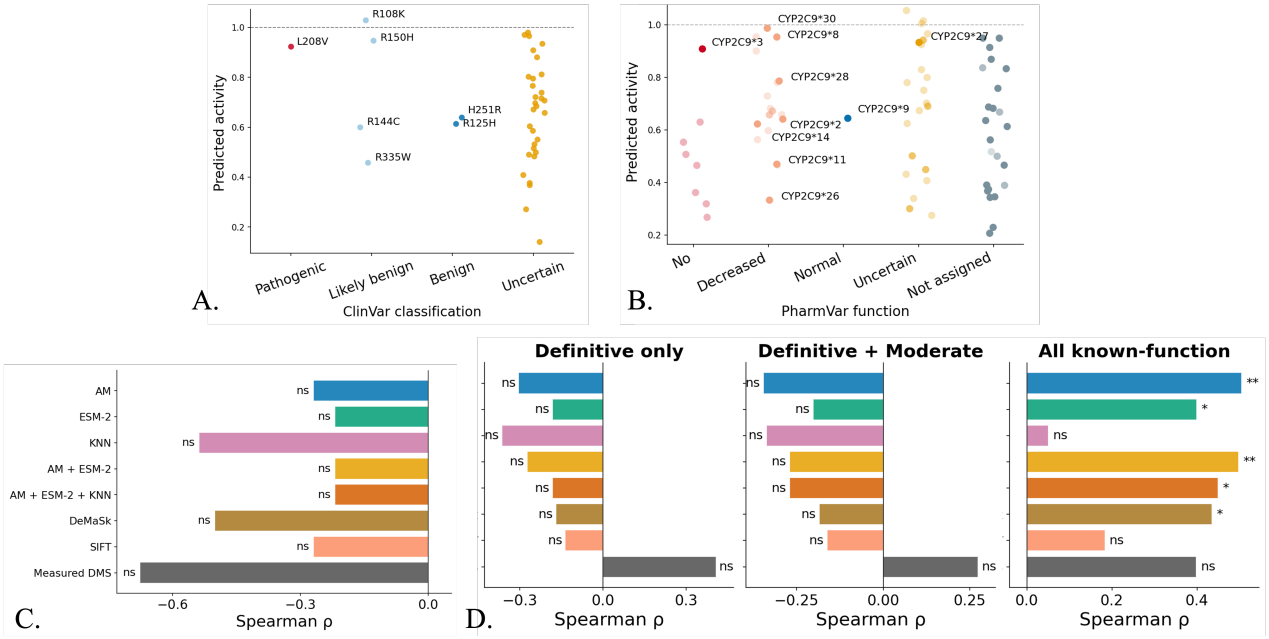


Fig. 3. Predicted activity of CYP2C9 missense variants plotted against categorical labels from ClinVar and PharmVar. **A, B**. Predicted CYP2C9 activity (y-axis) versus ClinVar (**A**) or PharmVar (**B**) annotations (x-axis) for the AM + ESM-2 + KNN ensemble model. Each point is a single missense variant; select variants are labeled. In (**B**), marker opacity encodes PharmVar evidence level: fully opaque markers denote definitive-evidence alleles and progressively more transparent markers denote moderate- and limited-evidence alleles. **C, D**. Spearman correlation of predicted activity with ClinVar (**C**) or PharmVar (**D**) annotation, by model. Significance is indicated per bar, where $^{*}p < 0.05$; $^{**}p < 0.01$; $^{***}p < 0.001$; ns, not significant. In (**D**), separate correlations are shown at the three PharmVar evidence thresholds (definitive only [$n = 9$], definitive + moderate [$n = 11$], all known function [$n = 27$]). The near-uniform absence of significance is the result: neither any method nor the measured assay separates from the rest.

predicted and measured activities agree (0.92 vs. 0.93) and both contradict the clinical label. DMS activity, clinical pathogenicity, and pharmacogenetic function are therefore related but distinct constructs, and accuracy against a biochemical assay does not guarantee concordance with clinical or pharmacogenetic annotations.

3.5. *Variant effects depend on the substrate*

In ClinPGx, the same variant is annotated with different functional consequences for different drug substrates, across all six CYPs examined (Fig. 4 shows CYP2C9 and CYP2D6; all six are in Fig. S6). Pooling across the six genes and restricting to variants with at least two drug annotations — the only variants for which substrate dependence is observable at all — 23 of 32 (71.9%) carry a function label that differs by drug. Restricting further to annotation pairs sharing an endpoint type leaves discordance essentially unchanged, at 20 of 28 (71.4%). Although sparse per-variant annotations leave formal testing underpowered, a discordance rate this high indicates that substrate identity, fixed by construction in the DMS assay and absent altogether from every model evaluated above, carries variant-effect information that a single per-variant activity value cannot represent.

4. Discussion

Using CYP pharmacogenes, we found that homology-based prediction fails systematically. AM reads the abundance axis of variant effect and misses an abundance-independent, catalytic one,

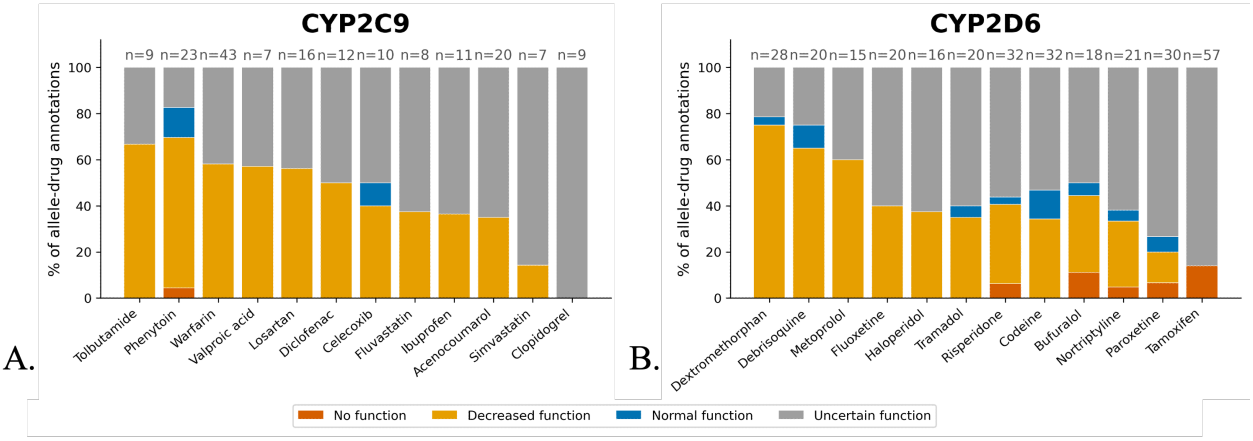


Fig. 4. Substrate-resolved variant effects. Stacked bar charts showing, for two representative CYP genes (CYP2C9 (**A**) and CYP2D6 (**B**)), the distribution of ClinPGx function annotations (no function, decreased function, normal function, and uncertain function) across the top 12 drug substrates for each CYP (x-axis). Each bar is one drug and shows the percentage of allele–drug annotations assigned each function label (y-axis, % of allele–drug annotations, 0–100%), where one annotation is a single allele–drug pair. The number of annotations (n) is given above each bar. The complete version for all six CYPs examined is Fig. S6.

most completely within the ambiguous class that CYPs are enriched for — a residual that sequence position, substitution chemistry, substrate-recognition-site proximity, and secondary structure do not explain. A protein-specific ensemble drawing on homology-derived features (KNN on ESM-2 embeddings, AM, and ESM-2 MMP) is nominally the most accurate activity predictor we evaluate, exceeding every generic, function-trained, and pharmacogene-tuned baseline. Yet it too fails against clinical and pharmacogenetic labels, as do all of those baselines and the measured assay itself. The breakdown we report is therefore not that these models read homology badly, which better use of the same representation would repair; it is that the best available use of homology-derived information reaches a ceiling short of what the clinical application of pharmacogenomics requires. We further show that the same variant is annotated with different consequences for different drug substrates across CYPs in ClinPGx, pointing to substrate identity as a determinant of CYP activity that no current predictor captures.

4.1. *Why homology breaks down on pharmacogenes*

Homology-based predictors succeed when evolutionary tolerance is a good proxy for functional consequence. For fold stability it usually is: a substitution that destabilizes the protein tends also to be evolutionarily disfavored, so scores derived from the residue distribution across homologs track it. Catalytic activity toward a specific substrate is different — it can be tuned by active-site substitutions that leave the fold intact and that selection, acting on xenobiotic breadth rather than on a single conserved catalytic output, need not have constrained.[3,4,6] Our abundance decomposition makes this concrete: AM's signal survives among destabilizing variants and vanishes among abundance-independent ones, exactly the split between the abundance axis it can see and the catalytic axis it cannot. This agrees with reports in other proteins that AM scores overlap with computed destabilization energies ($\Delta\Delta G$) most strongly at buried residues,[35] and with broader evidence that generic pathogenicity scores carry structural rather than catalytic information.[36,37]

4.2. *Function is multi-dimensional, and existing ground truth reflects this*

A predictor is only as well posed as its target, and our results make explicit that the resources used to define variant effect are not interchangeable: biochemical activity toward a probe substrate (DMS), disease causation (ClinVar), pharmacogenetic function (PharmVar), and substrate-resolved consequence (ClinPGx) answer different questions. That a model can reproduce measured activity closely yet not align with clinical labels is therefore expected rather than anomalous once function is recognized as multi-dimensional. The recent literature independently supports this. A DMS of CYP2C19 uncovered a substrate-specificity–abundance tradeoff within a single assay,[26] showing that even one experiment's notion of function is substrate-contingent; the difficulty of assigning substrate-dependent allelic effects is a standing clinical challenge;[38] and CYP2C9 activity depends even on cofactor context.[39] The ceiling that variant-effect predictors (VEPs) encounter is therefore as much definitional as methodological, since the target itself is underspecified when a CYP variant's effect is collapsed to a single substrate-agnostic number.

4.3. *Relationship to prior computational approaches*

Our protein-specific recovery of activity signal fits an active trend of adapting general predictors to narrower problems, from fine-tuning protein language models on DMS data[40–42] to pharmacogene-specific models that outperform off-the-shelf tools.[19,22–25] Using established predictors, including the DMS-derived DeMaSk[18] and the pharmacogene-calibrated PharmGScore,[19] alongside our CYP2C9-specific ensemble, we show that adaptation to functional data improves activity accuracy without moving a model toward the clinical target, pointing to a ceiling that adaptation alone cannot cross.

While some models do incorporate substrate identity, they remain narrow proofs of principle. Site-of-metabolism and enzyme-kinetics models predict how the wild-type enzyme handles a given drug,[43–48] but do not predict how a genetic variant alters metabolism of a specific substrate. Conversely, the widely used VEPs are substrate-agnostic. Work that does combine the two remains protein- or task-specific.[49,50] To our knowledge, no widely used, generalizable VEP conditions on the substrate.

4.4. *Toward substrate-conditioned variant-effect prediction*

We therefore propose substrate identity as the leading candidate for the non-homologous variable that current predictors omit, and substrate-conditioned prediction as the path beyond homology. Rather than mapping a variant to a scalar, such a model would predict function for a (variant, substrate) pair, most naturally as substrate-specific kinetic parameters of the kind that enzyme-kinetics models already target for wild-type enzymes.[47,48] Direct biochemistry is consistent with this hypothesis: a single active-site substitution can shift CYP2C9 metabolism in opposite directions across substrates. F476W raises (S)-warfarin catalytic efficiency five-fold while lowering (S)-flurbiprofen twenty-fold,[51] and active-site mutations reorder turnover of warfarin, flurbiprofen, and phenytoin differently.[52] These findings corroborate the substrate discordance we observe statistically in ClinPGx: for a multi-substrate enzyme, a variant's effect is a function of the substrate, which is exactly what a substrate-conditioned predictor

must learn. With the biological rationale thus established, the principal remaining obstacle is data: substrate-resolved measurements are sparse, so we present this as a direction rather than a result, though multi-substrate proof-of-concept studies suggest it is increasingly tractable.[53]

We consider two additional hypotheses as well. The first is that the missing axis is not substrate identity but cellular and physiological context. Expression level, redox-partner and cofactor availability, membrane environment, and compensating activity from other CYPs all modulate the phenotype a variant produces in vivo, and none is captured by a sequence-based score or by a single-cell-line assay. CYP2C9 activity is known to depend on cofactor context,[39] and this hypothesis could likewise explain the failure against clinical labels. The second is that the gap is substantially label-side. PharmVar and ClinVar labels are assigned from heterogeneous evidence of varying quality, are frequently driven by a single early study, and encode clinical judgment rather than a biochemical quantity, so part of the ceiling may reflect noise in the annotations rather than a missing biological variable. We regard substrate dependence as the most likely hypothesis, for three reasons: it is directly observed in the annotation data, it is corroborated by direct CYP2C9 biochemistry,[51,52] and it predicts precisely the abundance-independent, catalysis-specific blind spot our decomposition finds.

The clinical motivation also aligns with this framing. Genotype-guided prescribing is inherently drug-specific,[9,10] so a substrate-conditioned predictor targets exactly what the clinic requires, whereas a single activity score does not. The most promising route beyond homology here thus runs less through better similarity search than through enriching the definition of function, so that prediction is conditioned on substrate in addition to sequence conservation.

4.5. *Limitations*

Several limitations bound these conclusions. Our class-stratified analyses inherit AM's own class definitions: the ambiguous class is a band of AM's score axis fixed by the thresholds AM applies proteome-wide, not an independently motivated partition of CYP2C9 variants, so results reported are conditional on where those thresholds fall. Our activity and abundance benchmarks rest on a single DMS study of one gene;[8] whether the abundance bias generalizes to other CYPs awaits comparable paired data. The recovered accuracy comes from a protein-specific, recalibrated model and does not generalize beyond CYP2C9. Combining biochemical, clinical, and pharmacogenetic data is confounded by differences in measurement, thresholds, expression system, and evidence level. The clinical and pharmacogenetic comparisons rest on few labeled variants, so correlations are largely non-significant, sensitive to individual labels, and unstable in sign across evidence tiers. Finally, the sparse substrate-resolved annotations in ClinPGx establish that substrate-dependent effects occur for a meaningful fraction of variants but leave their overall prevalence underpowered to estimate.

Acknowledgements

This work is supported by NIH GM153195, NIH NLM F31LM014646, Burroughs Wellcome Fund IRSA 1074128, and a Stanford Medical Scholars award. We thank Stanford Research Computing for providing computational resources that contributed to this research.

References

1. J. Lazarou, B. H. Pomeranz and P. N. Corey, Incidence of adverse drug reactions in hospitalized patients: a meta-analysis of prospective studies, *JAMA* **279**, 1200 (4 1998).
2. J. Sultana, P. Cutroneo and G. Trifirò, Clinical and economic burden of adverse drug reactions, *Journal of Pharmacology & Pharmacotherapeutics* **4** (12 2013).
3. F. P. Guengerich, Human cytochrome P450 enzymes, in *Cytochrome P450: Structure, Mechanism, and Biochemistry, Fourth Edition*, (Springer International Publishing, 2015) pp. 523–785.
4. U. M. Zanger and M. Schwab, Cytochrome P450 enzymes in drug metabolism: Regulation of gene expression, enzyme activities, and impact of genetic variation, *Pharmacology and Therapeutics* **138**, 103 (4 2013).
5. F. P. Guengerich, Roles of Individual Human Cytochrome P450 Enzymes in Drug Metabolism, *Pharmacological Reviews* **76**, 1104 (11 2024).
6. M. Ingelman-Sundberg, Pharmacogenetics of cytochrome P450 and its applications in drug therapy: The past, present and future, *Trends in Pharmacological Sciences* **25**, 193 (4 2004).
7. D. M. Roden, R. B. Altman, N. L. Benowitz, D. A. Flockhart, K. M. Giacomini, J. A. Johnson, R. M. Krauss, H. L. McLeod, M. J. Ratain, M. V. Relling, H. Z. Ring, A. R. Shuldiner, R. M. Weinshilboum and S. T. Weiss, Pharmacogenomics: Challenges and Opportunities, *Annals of Internal Medicine* **145**, p. 749 (5 2006).
8. C. J. Amorosi, M. A. Chiasson, M. G. McDonald, L. H. Wong, K. A. Sitko, G. Boyle, J. P. Kowalski, A. E. Rettie, D. M. Fowler and M. J. Dunham, Massively parallel characterization of CYP2C9 variant enzyme activity and abundance, *American Journal of Human Genetics* **108**, 1735 (9 2021).
9. M. Pirmohamed, Pharmacogenomics: current status and future perspectives, *Nature Reviews Genetics* **24**, 350 (1 2023).
10. R. Mosch, M. van der Lee, H. J. Guchelaar and J. J. Swen, Pharmacogenetic Panel Testing: A Review of Current Practice and Potential for Clinical Implementation, *Annual Review of Pharmacology and Toxicology* **65**, 91 (1 2025).
11. R. Vaser, S. Adusumalli, S. N. Leng, M. Sikic and P. C. Ng, Sift missense predictions for genomes, *Nature protocols* **11**, 1 (1 2016).
12. J. Cheng, G. Novati, J. Pan, C. Bycroft, A. Žemgulyte, T. Applebaum, A. Pritzel, L. H. Wong, M. Zielinski, T. Sargeant, R. G. Schneider, A. W. Senior, J. Jumper, D. Hassabis, P. Kohli and Ž. Avsec, Accurate proteome-wide missense variant effect prediction with AlphaMissense, *Science* **381** (9 2023).
13. J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, Highly accurate protein structure prediction with AlphaFold, *Nature* **596**, 583 (7 2021).
14. Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido and A. Rives, Evolutionary-scale prediction of atomic-level protein structure with a language model, *Science* **379**, 1123 (3 2023).
15. N. Brandes, G. Goldman, C. H. Wang, C. J. Ye and V. Ntranos, Genome-wide prediction of disease variant effects with a deep protein language model, *Nature Genetics* **55**, 1512 (2023).
16. J. Meier, R. Rao, R. Verkuil, J. Liu, T. Sercu and A. Rives, Language models enable zero-shot prediction of the effects of mutations on protein function, in *Advances in Neural Information Processing Systems*, eds. M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang and J. W. Vaughan (Curran Associates, Inc., 2021).

17. N. M. Ioannidis, J. H. Rothstein, V. Pejaver, S. Middha, S. K. McDonnell, S. Baheti, A. Musolf, Q. Li, E. Holzinger, D. Karyadi, L. A. Cannon-Albright, C. C. Teerlink, J. L. Stanford, W. B. Isaacs, J. Xu, K. A. Cooney, E. M. Lange, J. Schleutker, J. D. Carpten, I. J. Powell, O. Cussenot, G. Cancel-Tassin, G. G. Giles, R. J. MacInnis, C. Maier, C. L. Hsieh, F. Wiklund, W. J. Catalona, W. D. Foulkes, D. Mandal, R. A. Eeles, Z. Kote-Jarai, C. D. Bustamante, D. J. Schaid, T. Hastie, E. A. Ostrander, J. E. Bailey-Wilson, P. Radivojac, S. N. Thibodeau, A. S. Whittemore and W. Sieh, Revel: An ensemble method for predicting the pathogenicity of rare missense variants, *American journal of human genetics* **99**, 877 (10 2016).
18. D. Munro and M. Singh, Demask: a deep mutational scanning substitution matrix and its use for variant impact prediction, *Bioinformatics* **36**, 5322 (4 2021).
19. J. Hajto, M. Piechota, I. Krätschmer, P. Konowalska, G. E. Boyle, D. M. Fowler, M. Borczyk and M. Korostynski, Computational variant predictors for pharmacogenomics: from evaluation of single alleles to assessment of adverse drug reactions to antidepressants, *The Pharmacogenomics Journal* **26**, p. 8 (3 2026).
20. M. Mouterde, Y. Daali, V. Rollason, M. Cizkova, A. Mulugeta, K. A. A. Balushi, G. Fakis, T. C. Constantinidis, K. Al-Thihli, M. Cerna, E. Makonnen, S. Boukouvala, S. Al-Yahyaee, G. Yimer, V. Cerny, J. Desmeules and E. S. Poloni, Joint Analysis of Phenotypic and Genomic Diversity Sheds Light on the Evolution of Xenobiotic Metabolism in Humans, *Genome Biology and Evolution* **14**, p. evac167 (2022).
21. D. W. Nebert, J. E. Jones, J. Owens and A. Puga, Evolution of the P450 gene superfamily, *Progress in Clinical and Biological Research* **274**, 557 (1988).
22. Y. Zhou, S. Mkrtchian, M. Kumondai, M. Hiratsuka and V. M. Lauschke, An optimized prediction framework to assess the functional impact of pharmacogenetic variants, *The Pharmacogenomics Journal* **19**, 115 (9 2018).
23. Y. Zhou, S. Pirmann and V. M. Lauschke, APF2: an improved ensemble method for pharmacogenomic variant effect prediction, *The Pharmacogenomics Journal* **24**, p. 17 (5 2024).
24. R. Tremmel, A. Honore, Y. Park, Y. Zhou, M. Xiao and V. M. Lauschke, Machine learning models for pharmacogenomic variant effect predictions – recent developments and future frontiers, *Pharmacogenomics* **26**, 171 (4 2025).
25. G. McInnes, R. Dalton, K. Sangkuhl, M. Whirl-Carrillo, S. B. Lee, P. S. Tsao, A. Gaedigk, R. B. Altman and E. L. Woodahl, Transfer learning enables prediction of CYP2D6 haplotype function, *PLOS Computational Biology* **16**, p. e1008399 (11 2020).
26. G. E. Boyle, K. A. Sitko, J. G. Galloway, H. K. Haddox, A. H. Bianchi, A. Dixon, M. K. Wheelock, A. J. Vandi, Z. R. Wang, R. E. S. Thomson, R. K. Garge, A. E. Rettie, A. F. Rubin, R. C. Geck, E. M. J. Gillam, W. S. Dewitt, F. A. Matsen and D. M. Fowler, Deep mutational scanning of CYP2C19 in human cells reveals a substrate specificity-abundance tradeoff, *Genetics* **228**, p. iyae156 (11 2024).
27. M. J. Landrum, J. M. Lee, G. R. Riley, W. Jang, W. S. Rubinstein, D. M. Church and D. R. Maglott, ClinVar: public archive of relationships among sequence variation and human phenotype, *Nucleic Acids Research* **42**, D980 (2013).
28. S. Richards, N. Aziz, S. Bale, D. Bick, S. Das, J. Gastier-Foster, W. W. Grody, M. Hegde, E. Lyon, E. Spector, K. Voelkerding and H. L. Rehm, Standards and Guidelines for the Interpretation of Sequence Variants: A Joint Consensus Recommendation of the American College of Medical Genetics and Genomics and the Association for Molecular Pathology, *Genetics in Medicine* **17**, 405 (2015).
29. A. Gaedigk, S. T. Casey, M. Whirl-Carrillo, N. A. Miller and T. E. Klein, PharmVar: A Global Resource and Repository for Pharmacogene Variation, *Clinical Pharmacology & Therapeutics* **110**, 542 (2021).
30. M. Whirl-Carrillo, R. Huddart, L. Gong, K. Sangkuhl, C. F. Thorn, R. Whaley and T. E. Klein,

An Evidence-Based Framework for Evaluating Pharmacogenomics Knowledge for Personalized Medicine, *Clinical Pharmacology & Therapeutics* **110**, 563 (2021).
31. The UniProt Consortium, UniProt: the universal protein knowledgebase in 2021, *Nucleic Acids Research* **49**, D480 (2021).
32. X. Liu, C. Li, C. Mou, Y. Dong and Y. Tu, dbnsfp v4: a comprehensive database of transcript-specific functional predictions and annotations for human nonsynonymous and splice-site snvs, *Genome medicine* **12** (12 2020).
33. Łukasz Gricman, C. Vogel and J. Pleiss, Identification of universal selectivity-determining positions in cytochrome p450 monooxygenases by systematic sequence-based literature mining, *Proteins: Structure, Function and Bioinformatics* **83**, 1593 (9 2015).
34. A. Midlik, V. Navrátilová, T. R. Moturu, J. Koča, R. Svobodová and K. Berka, Uncovering of cytochrome P450 anatomy by SecStrAnnotator, *Scientific Reports* **11**, p. 12345 (6 2021).
35. N. Rotenberg, C. Fortuno, M. J. Varga, A. C. Chamberlin, L. Ramadane-Morchadi, B. J. Feng, M. de la Hoya, M. E. Richardson and A. B. Spurdle, Integration of protein stability and AlphaMissense scores improves bioinformatic impact prediction for p53 missense and in-frame amino acid deletion variants, *American Journal of Human Genetics* **112**, 1003 (2025).
36. F. J. Guzmán-Vega, K. J. C.-L. no, A. C. González-Álvarez, K. A. P. na Guerra, A. Althagafi, T. Khan, R. Hoehndorf and S. T. Arold, Varland: A pipeline to map the structural landscape of missense variants at the proteome scale, *Journal of Biological Chemistry* **302**, p. 111071 (2025).
37. G. Zhong, Y. Zhao, D. Zhuang, W. K. Chung and Y. Shen, Premode predicts mode-of-action of missense variants by deep graph representation learning of protein sequence and structural context, *Nature Communications* **16**, p. 7189 (2025).
38. L. A. Marcath, A. L. Pasternak and D. L. Hertz, Challenges to assess substrate-dependent allelic effects in CYP450 enzymes and the potential clinical implications, *The Pharmacogenomics Journal* **19**, 501 (10 2019).
39. P. I. Koroleva, A. V. Kuzikov, A. A. Gilep, S. V. Ivanov, A. I. Archakov and V. V. Shumyantseva, The Activation of Cytochrome P450 2C9 Is Facilitated by the Coenzyme Forms of Vitamin B2, *Molecules* **30**, p. 3673 (9 2025).
40. A. Lafita, F. Gonzalez, M. Hossam, P. Smyth, J. Deasy, A. Allyn-Feuer, D. Seaton and S. Young, Fine-tuning Protein Language Models with Deep Mutational Scanning improves Variant Effect Prediction, *arXiv preprint arXiv:2405.06729* (5 2024).
41. M. Glaser and J. Brägelmann, ESM-Effect: An Effective and Efficient Fine-Tuning Framework towards accurate prediction of Mutation's Functional Effect, *bioRxiv* , p. 2025.02.03.635741 (2 2025).
42. T. Deguchi, N. S. A. Ghani, Y. Kurumida, S. Iida, K. Kobayashi and Y. Saito, Data-efficient protein mutational effect prediction with weak supervision by molecular simulation and protein language models, *Briefings in Bioinformatics* **26** (8 2025).
43. H. Kim, E. Lee, S. Chung, J. Park, S. Baek and J. Kang, ATTNSOM: Learning Cross-Isoform Attention for Cytochrome P450 Site-of-Metabolism, *arXiv preprint arXiv:2601.20891* (1 2026).
44. J. Chang, X. Fan and B. Tian, DeepP450: Predicting Human P450 Activities of Small Molecules by Integrating Pretrained Protein Language Model and Molecular Representation, *Journal of Chemical Information and Modeling* **64**, 3149 (4 2024).
45. X. Huang, J. Chang and B. Tian, GLMCyp: A Deep Learning-Based Method for CYP450-Mediated Reaction Site Prediction, *Journal of Chemical Information and Modeling* **65**, 2322 (3 2025).
46. Z. Nie, H. Zhang, H. Jiang, Y. Liu, X. Huang, F. Xu, J. Fu, Z. Ren, Y. Tian, W.-B. Zhang and J. Chen, OmniESI: A unified framework for enzyme-substrate interaction prediction with progressive conditional deep learning, *arXiv preprint arXiv:2506.17963* (2025).
47. F. Li, L. Yuan, H. Lu, G. Li, Y. Chen, M. K. M. Engqvist, E. J. Kerkhoven and J. Nielsen,

Deep learning-based kcat prediction enables improved enzyme-constrained model reconstruction, *Nature Catalysis* **5**, 662 (6 2022).
48. H. Yu, H. Deng, J. He, J. D. Keasling and X. Luo, UniKP: a unified framework for the prediction of enzyme kinetic parameters, *Nature Communications* **14**, p. 8211 (12 2023).
49. I. Johansson, E. M. Xanthopoulou, Y. Zhou, A. Sanchez-Spitman, M. van der Lee, B. M. Wollmann, E. Størset, J. J. Swen, H. J. Guchelaar, E. Molden, M. M. Jukic and M. Ingelman-Sundberg, Improved Prediction of CYP2D6 Catalyzed Drug Metabolism by Taking Variant Substrate Specificities and Novel Polymorphic Haplotypes into Account, *Clinical Pharmacology and Therapeutics* **118**, 218 (7 2025).
50. Y. Park, Y. Zhou, M. Xiao, A. T. Nies and V. M. Lauschke, Feed-Forward Deep Neural Networks Predict Substrate-Specific Effects of Transporter Variants to Explain Drug Response Variability, *Clinical and Translational Science* **19**, p. e70592 (5 2026).
51. C. M. Mosher, M. A. Hummel, T. S. Tracy and A. E. Rettie, Functional Analysis of Phenylalanine Residues in the Active Site of Cytochrome P450 2C9, *Biochemistry* **47**, 11725 (2008).
52. C. M. Mosher, G. Tai and A. E. Rettie, CYP2C9 Amino Acid Residues Influencing Phenytoin Turnover and Metabolite Regio- and Stereochemistry, *Journal of Pharmacology and Experimental Therapeutics* **329**, 938 (2009).
53. R. Vanella, S. Boult, C. Küng and M. A. Nash, Decoding the substrate specificity landscape of a promiscuous enzyme through multi-substrate mutational scanning, *Nature Communications* **17**, p. 3245 (2 2026).

Auditable Algebraic Counting Field for Cryptic-Pocket Detection from Apo Structures

Shan Yu[1] and Xuening Wu[2,†]
[1]*Independent Researcher, Hangzhou, China*
[2]*Pfizer, Shanghai, China*
[†]*E-mail: xuening.wu@pfizer.com*

Cryptic ligand-binding pockets are not apparent in experimentally determined apo structures, making them difficult to identify from unbound receptor geometry. A complementary challenge is to make the structural measurements and learned evidence behind each prediction directly inspectable. We introduce a supervised algebraic counting field (ACF) for predicting cryptic-pocket residues from apo structures. ACF compiles explicit geometric, physicochemical, and topological features into compact, integer-weighted lookup tables. Each prediction score can be reconstructed from feature values, training counts, table weights, and spatial aggregation, without sequence search, structural-template transfer, or a protein language model at inference. We evaluate ACF on CryptoBench and two locked external collections, separating ranking performance from the effects of residue-calling budgets. On an external set of 57 post-CryptoBench apo–holo units, ACF exceeded P2Rank by +0.044 in mean paired ROC-AUC (multiplicity-adjusted 95% CI [+0.010, +0.079]). The advantage was dataset-dependent: official-fold ROC-AUC and matched-budget F1 differences against P2Rank remained unresolved, and a second external evaluation did not confirm gains from added structural features. ACF thus provides a compact predictor with externally validated signal and an inspectable path from structural measurements and training counts to residue scores.

Keywords: cryptic binding sites; protein function; structure-based prediction; apo proteins; interpretable machine learning; algebraic counting field

1. Introduction

Cryptic ligand-binding sites may be absent or closed in an observed apo structure, although local geometry and chemistry may indicate where opening is possible.[1–4] Identifying their lining residues therefore tests structure-based functional inference beyond annotation transfer from homologous sequences or structural templates.

Structure-based predictors span explicit geometry, energetic probes, simulation, and learned representations. The geometry-based method Fpocket detects cavities from the spatial arrangement of protein atoms.[5] Other established approaches include geometric and physicochemical site scoring in SiteMap and DoGSiteScorer and energetic probe mapping in FTMap.[6–8] DeepSite, Kalasanty, and DeepPocket instead formulate pocket detection or segmentation with three-dimensional convolutional networks.[9–11] P2Rank learns ligandability scores for local chemical

© 2026 The Authors. Open Access chapter published by World Scientific Publishing Company and distributed under the terms of the Creative Commons Attribution Non-Commercial (CC BY-NC) 4.0 License.

neighborhoods on the protein surface.[12] PocketMiner uses a graph neural network to identify residues likely to participate in cryptic-pocket opening.[13] CryptoBench adds a supervised protein-language-model baseline (pLM-NN), drawing on the sequence representations learned by large protein language models, and sequence-clustered splits for cryptic-site evaluation.[14,15] These methods show that single structures and learned representations contain useful site information. They also motivate a complementary question: can a predictor remain competitive while allowing each score to be mechanically reconstructed from explicit local measurements and training counts? This focus on traceability distinguishes explanations constructed after prediction from models whose predictive computations are directly inspectable.[16]

We answer this question with an *algebraic counting field* (ACF)—a discrete counting architecture over local structural neighborhoods in which every score decomposes into explicit table lookups and integer-weighted additions. Within each protein chain, structural features are rank-discretized into four levels and paired to address compact lookup tables compiled from training-fold residue counts. A regularized compilation step assigns bounded integer multiplicities to the tables. The deployed score requires no neural-network forward pass or external database search, although the method is *not* training-free: its table counts, residue propensity, and integer multiplicities are derived from the training fold.

The contribution is an explicit structural representation with a compact, exactly decomposable readout, evaluated against strong baselines with ranking and operating-point effects separated. Two external reads support transferable signal but delimit accuracy claims; controlled training-fold probes identify additional structural information whose benefit was not confirmed externally. Figure 1 summarizes the training and prediction workflow.

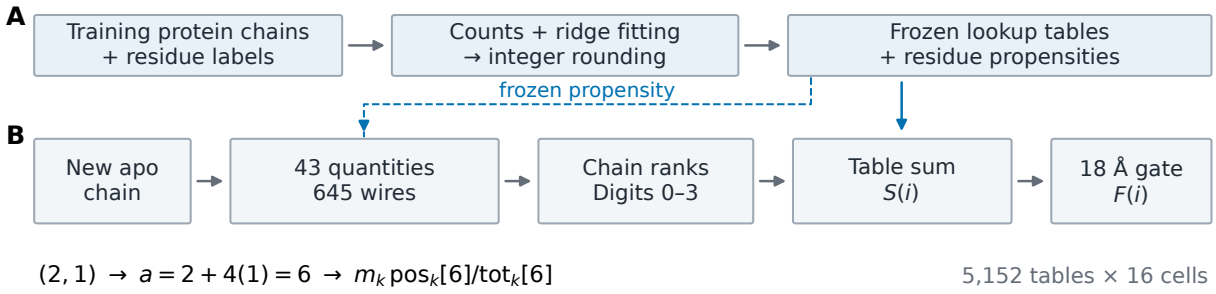


Fig. 1. ACF workflow. (A) Training uses apo protein chains and residue labels to compile lookup-table counts, integer weights, and residue-type propensities, which are then frozen. (B) Prediction uses a new apo chain's structural features and within-chain ranks with the frozen parameters to obtain residue scores through table lookup and spatial aggregation; no residue labels are required.

2. Materials and Methods

2.1. *Benchmark, evaluation units, and leakage controls*

We use the official CryptoBench apo split, in which the training and test partitions are disjoint under MMseqs2 clustering at 10% sequence identity.[15,17] Receptor structures are retrieved from

the RCSB Protein Data Bank (RCSB PDB),[18] and every coordinate and label file is verified against a recorded SHA-256 hash. Data loading halts if the manifest is absent, a file hash has changed, the declared clustering metadata differ from the recorded values, or a cluster identifier occurs in both partitions.

Each evaluation unit is identified by a PDB entry–chain pair. The official analysis includes 192 single-chain units drawn from 190 PDB entries. Thirty-eight cases requiring multi-chain residue mapping, plus one chain not retrievable in legacy PDB format, are excluded. The multi-chain cases are excluded because the benchmark labels do not unambiguously assign residues to individual chains. This exclusion is an explicit scope restriction. All structure-based predictors receive only apo-receptor coordinates. A runtime guard rejects inputs containing non-solvent heteroatoms or other non-polymer atoms, preventing ligand coordinates from entering receptor-only predictions. Table 1 summarizes the frozen evaluation partitions.

Table 1. Frozen CryptoBench partitions. Each unit is a PDB entry–chain pair.

Partition	Units	Residues	Role
Training fold	770	234,838	ACF compilation
Fit half	386	118,716	Candidate-readout fitting
Selection half	384	116,122	Configuration selection
Official test subset	192	—	Final paired evaluation

2.2. *Local quantities, wires, and ACF compilation*

Each residue has 43 explicit base quantities: 35 geometric and topological descriptors, seven fixed physicochemical constants, and one Laplace-smoothed residue-type propensity estimated only from the training fold. The structural descriptors cover surface exposure, local spectral geometry, density-field calculus, ultrametric organization, curvature, anisotropy, and global position. The seven fixed physicochemical quantities are Kyte–Doolittle hydropathy,[19] Zamyatnin side-chain volume,[20] aromatic-ring count, formal charge, hydrogen-bond donor and acceptor counts, and the number of rotatable side-chain bonds. The public implementation documents all definitions, sources, neighborhoods, boundary cases, and expansion rules. Geometric and physicochemical extensions require recompilation and separate evaluation. Propensity is fixed during prediction and re-estimated for a different training fold.

For residue i, define $N_r(i) = \{j : \|c_i - c_j\| \le r\}$. Here, c_i is the centroid of its observed heavy atoms, and the intra-chain neighborhood includes i. Each quantity x is used directly and through its neighborhood mean, standard deviation, centered difference, and local rank:

$$\mathrm{sd}_r[x]_i = \sqrt{\max\left(\overline{x^2}_{N_r(i)} - \left(\overline{x}_{N_r(i)}\right)^2, 0\right)}, \quad D_r[x]_i = x_i - \overline{x}_{N_r(i)}, \quad K_r[x]_i = \frac{|\{j \in N_r(i) : x_j < x_i\}|}{|N_r(i)|}.$$

Each resulting residue-level scalar feature is called a wire. Neighborhood means are evaluated at 6, 10, 14, 20, and 26 Å, whereas dispersion, centered difference, and local rank are evaluated at 6, 14, and 20 Å. These predefined transformations retain all 15 versions of each base quantity, yielding $43(1 + 5 + 3 + 3 + 3) = 645$ wires without a further feature-selection step.

Each wire is discretized separately within the scored chain. For a chain of n residues, let $r_j(i)$ be the zero-based rank of wire j, with ties assigned their mean rank. Its digit is $d_j(i) = \min\{3, \lfloor 4r_j(i)/\max(n-1,1)\rfloor\}$. This discretization is invariant to strictly increasing transformations of that wire; ties need not produce equally populated bins. No training-set bin edges are transferred.

A table's *width* is the number of input wires. For a width-2 table with ordered wire indices (u_k, v_k), the address is $a_k(i) = d_{u_k}(i) + 4d_{v_k}(i) \in \{0, \ldots, 15\}$. Thus a table has $4^2 = 16$ cells: for example, digits $(2, 1)$ select cell 6. The bank is generated by 16 successive permutations of all 645 wire indices using NumPy `default_rng(20260725)`. Adjacent indices are paired and the final singleton is discarded in each round; pairs are not selected by labels or biological criteria. Each round contributes $\lfloor 645/2 \rfloor = 322$ tables, hence $16 \times 322 = 5{,}152$ tables and $5{,}152 \times 16 = 82{,}432$ cells. A wire is used at most once per round; partners can recur across rounds. With the frozen seed, 629 wires occur in 16 tables and 16 wires in 15 tables, so none is absent from the complete bank.

For table k, cell a stores two training-fold counts: $\mathrm{tot}_k[a]$, the number of residues reaching it, and $\mathrm{pos}_k[a]$, the number labeled cryptic. Its value is the unsmoothed fraction $\mathrm{pos}_k[a]/\mathrm{tot}_k[a]$, or the training-fold base rate for an unseen cell. There are 825 unseen cells (1.00%) in the full-fold compilation.

The unaggregated residue score is

$$S(i) = \sum_{k=1}^{5{,}152} m_k \frac{\mathrm{pos}_k[a_k(i)]}{\mathrm{tot}_k[a_k(i)]}, \qquad m_k \in \mathbb{Z} \cap [-32, 32],$$

with the training-fold base rate substituted for an unseen cell. The deployed score is

$$F(i) = S(i) + \overline{S}_{18}(i) \frac{\mathrm{sd}(S)}{\mathrm{sd}(\overline{S}_{18})}.$$

Here, $\overline{S}_{18}(i)$ is the mean score among residues within 18 Å of residue i, and the standard deviations are computed across residues in the scored chain. This spatial gate favors coherent clusters of high-scoring residues over isolated high scores. The standard-deviation ratio places the neighborhood term on the same dispersion scale as S. Radius 18 Å and weight 1 were selected on a UniProt accession-disjoint training half and frozen; this internal split does not guarantee sequence-cluster separation. The gate is skipped if the neighborhood mean has zero standard deviation.

The integer multiplicities m_k are obtained by solving a ridge-regularized closed-form system over training-fold table outputs and rounding the resulting coefficients. Let Σ be the within-class scatter of table outputs divided by $\max(N-2, 1)$, where N is the number of training residues, and let μ_1, μ_0 be the class means. The solution is

$$w = \left[\Sigma + \left(0.03\,\frac{\mathrm{tr}\,\Sigma}{K} + 10^{-12}\right) I\right]^{-1} (\mu_1 - \mu_0), \qquad K = 5{,}152.$$

Weights are $m_k = \mathrm{round}(32\, w_k / \max_j |w_j|)$, or all zero when the solution is zero. The ridge penalty scales with the mean diagonal, not the unnormalized trace. Compilation uses a real-valued solve; inference uses floating-point cell fractions, integer coefficients, and spatial statistics. Of the 5,152 compiled tables, 4,797 have nonzero multiplicity in the full-training-fold ACF.

2.3. *Baselines and operating points*

Rigid geometry (`geometric_foundation`) is a training-free static-coordinate baseline that ranks free-space points by ray enclosure and a local cavity-volume proxy. It uses neither ACF's 43 quantities nor its 645-wire neighborhood expansion; it is not an ACF ablation.

P2Rank 2.5.1 is executed locally with a pinned binary and Java environment. Ranking metrics use its native residue probability. Its shipped binary output is the native pocket assignment. PocketMiner is rebuilt and checked on its published self-test, where the reconstructed score is 0.8677 versus the reported 0.87. The pLM-NN baseline uses the benchmark's ESM2-3B embedding and trained head; reconstruction fidelity to the released example is 0.9987 mean cosine similarity. A logistic regression on the same 645 continuous wires, standardized on the fit half, tests the readout on the training selection half. For post-hoc external evaluation, standardization and logistic coefficients were refitted on all 770 training chains, retaining the training-derived propensity and the logistic model's training-selected gate (18 Å, weight 0.5); the deployed ACF retains its 18 Å gate with weight 1. Both models scored the same residues. Set A and Sets B+C were evaluated separately using 4,000 paired chain-bootstrap resamples (seed 20260929) and unadjusted 95% intervals. This supplementary comparison was not preregistered.

The counting field's deployed binary call is the top 0.09 fraction of residues per chain, selected on the training fold. To remove an operating-point asymmetry, we additionally call the top 0.09 fraction of residues in each chain for both the field and P2Rank. Applying the same grid search to P2Rank's training-fold predictions also selects 0.09, so the common-budget and separately training-selected rules coincide. Threshold curves from 2% to 40% are descriptive robustness analyses; they do not change the deployed operating point. P2Rank's test-fold-best threshold is shown only as a descriptive comparison in Fig. 3.

2.4. *External validation*

The first external set (Set A) contains 57 apo–holo units and 905 positive residues from depositions released after 2024-05-08, the newest structure date in CryptoBench. It retains one unit per UniRef50 cluster and shares no cluster with either CryptoBench partition. Coordinates, labels, inclusion decisions, and hashes were frozen before scoring. A committed analysis plan names three co-primary paired ROC-AUC comparisons—the deployed field against P2Rank, pLM-NN, and PocketMiner—and applies a Bonferroni correction. Binary comparisons use the same matched top-9% budget as the official-fold robustness analysis.

A second external read (Sets B+C) tests whether the training-fold lift from four additional feature families transfers. These sets contain 45 scorable single-particle cryo-EM apo–holo units, are disjoint at UniRef50 from CryptoBench, Set A, and each other, and were frozen before the additional families were developed. One prespecified unit was excluded before scoring because its three-character chain identifier could not be represented by the legacy PDB writer. After the expanded 1269-column field was fixed, a committed plan pinned four co-primary mean paired ROC-AUC comparisons: expanded field minus the deployed 645-wire field, pLM-NN, P2Rank, and PocketMiner. All five methods were evaluated on the same residue universe; no unit or residue was lost to the intersection. Bonferroni-adjusted intervals control the four

comparisons. The repository commit graph is part of the chronology evidence for both external reads and must accompany the archival submission.

2.5. *Statistical analysis*

Metrics are computed per unit and averaged across units. We report both ROC-AUC and PR-AUC because precision–recall summaries are informative under class imbalance.[21] For the official-fold and external comparisons and the internal logistic comparison, intervals for paired differences use a paired nonparametric bootstrap over units.[22] The main evaluations use 10000 resamples; the internal and post-hoc external logistic comparisons use 4,000. Undefined per-unit metrics are excluded from the corresponding paired comparison; they are never imputed as zero. "Resolved" denotes a paired interval excluding zero, with multiplicity adjustment where comparisons are co-primary. The mean paired ROC-AUC remains the primary estimand. A 20% trimmed mean that was selected on the training partition is reported as exploratory because 4 indexed reads of the official fold, 3 from the present architecture lineage, preceded its registration.

Structural-family intervals are instead $\bar{d} \pm 1.96\, s_d/\sqrt{S}$, where $\bar{d}$ and s_d are the mean and sample standard deviation of paired ROC-AUC gains across $S = 12$ half-splits. These descriptive intervals ignore dependence from reused data and support no confirmatory significance or population-coverage claims.

2.6. *Training-only structural-family probes*

To test what information the deployed representation omits, we appended predefined feature families while holding the deployed pair bank, gate, ridge, fan-out cap, and quantization fixed. Mean per-unit ROC-AUC lift over the frozen deployed detector was measured on 12 UniProt accession-disjoint halvings of the CryptoBench training fold (seeds 20260725–20260736). Each family was compared with a size-matched control that added the same number of tables over already deployed wires. For families that read additional atom-level data, a second control permuted rows within each chain, preserving column marginals while destroying residue assignment. These probes are exploratory analyses of the representation; they do not use the official or external folds for candidate selection or change the deployed detector.

3. Results

3.1. *Official-fold ranking and strong baselines*

Table 2 and Fig. 2 summarize the official-fold results. The counting field reached ROC-AUC 0.799 [0.766, 0.830], PR-AUC 0.376 [0.339, 0.411], MCC 0.304 [0.270, 0.338], and F1 0.333 [0.301, 0.365] on the official fold. P2Rank reached 0.793, 0.358, 0.285, and 0.302, respectively. The paired mean ROC-AUC difference was +0.0058 [-0.0168, +0.0276] ($p = 0.598$); PR-AUC was likewise unresolved (+0.0177 [-0.0094, +0.0450]). The field was ahead on 115 chains and behind on 77, with a median difference of +0.0412. The exploratory 20% trimmed mean was +0.0281 [+0.0117, +0.0427], but it cannot repair the prior exposure of this fold and is not the primary claim.

Table 2. Official-fold results. The first three rows report all four metrics; the last two report ROC-AUC for the reconstructed baselines.

Method	ROC-AUC	PR-AUC	MCC	F1
Counting field	0.799	0.376	0.304	0.333
P2Rank 2.5.1	0.793	0.358	0.285	0.302
Rigid geometry	0.665	0.276	0.239	0.287
PocketMiner	0.7523	—	—	—
pLM-NN	0.8235	—	—	—

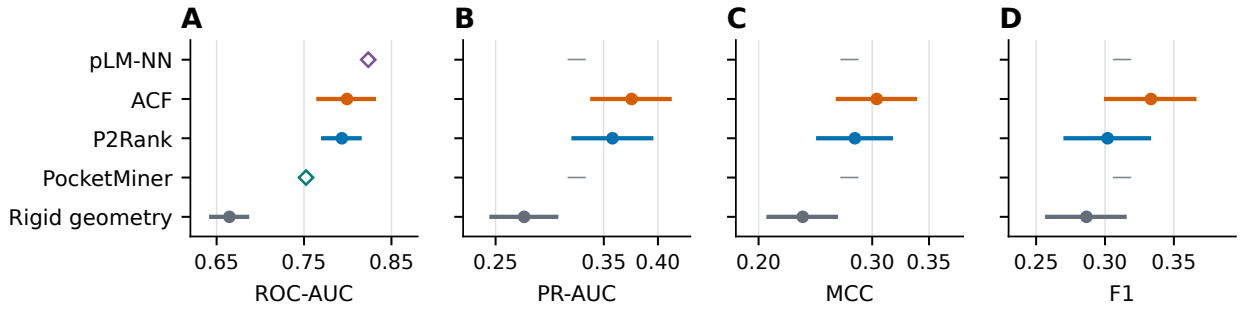


Fig. 2. Official-fold means (192 units). Filled points: means with 95% bootstrap intervals; open diamonds: reported means without marginal intervals; dashes: metrics not shown. MCC/F1 use method-specific calls (ACF: top 9%; P2Rank: native pockets). Axes differ. The paired ACF–P2Rank ROC-AUC difference remains unresolved.

The stronger baselines bound the accuracy claim. The field-minus-pLM-NN ROC-AUC difference was -0.0243 [-0.0465, -0.0033]; the field scored lower on 111 of 192 chains. At a common calling budget, its F1 and MCC differences were -0.0522 [-0.0786, -0.0274] and -0.0561 [-0.0863, -0.0269]. pLM-NN remained ahead at all 39 calling fractions. The field exceeded PocketMiner by +0.0468 [+0.0251, +0.0682], but P2Rank also exceeded PocketMiner by +0.0410 [+0.0211, +0.0614]. Differences in label definitions may contribute to the PocketMiner gap; these comparisons do not isolate its cause or establish general methodological superiority.

On the training selection half, the same-wire logistic model reached ROC-AUC 0.8008; the field-minus-logistic difference was +0.0038 [-0.0057, +0.0125]. This training-fold comparison does not establish an accuracy advantage for the table architecture.

3.2. *The F1 conclusion depends on the operating convention*

Fig. 3 compares the shipped and matched operating conventions. Under shipped outputs, counting-field F1 exceeded P2Rank by +0.0315. This compares the field's top-fraction rule (0.09 of residues) with P2Rank's native pocket assignment, which called 1.65 times as many residues. When both methods called the top 0.09 fraction of residues in each chain, their mean F1 values were 0.3334 and 0.3194, and the paired difference was +0.0140 [-0.0085, +0.0364] ($p = 0.224$). MCC was also unresolved under the common budget (+0.0169 [-0.0092, +0.0424]). These results do not support a convention-independent F1 advantage.

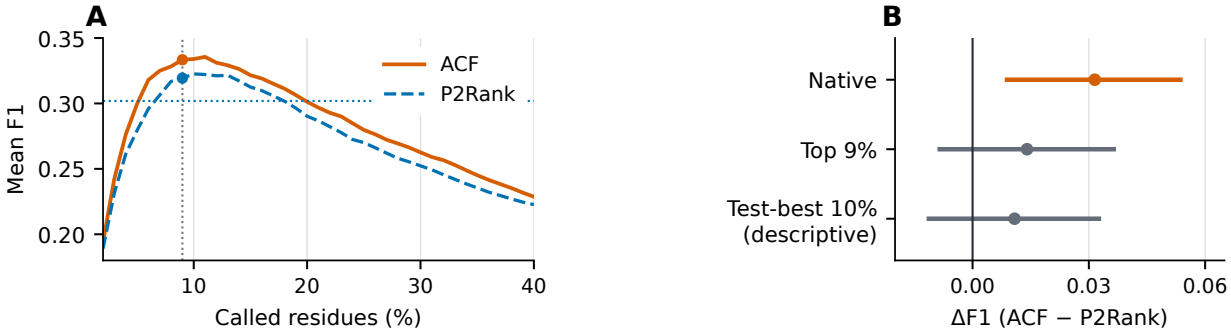


Fig. 3. Calling-budget sensitivity. (A) F1 curves; dotted lines mark training-selected 9% calls (vertical) and P2Rank native-pocket F1 (horizontal). (B) Paired differences with 95% intervals: native rules, matched top-9%, and ACF at 9% versus P2Rank's test-fold-best 10% (descriptive). Matched-budget F1 is unresolved.

Table 3. Locked external validation on 57 apo–holo units. Differences are counting field minus comparator in mean paired ROC-AUC; intervals are unadjusted. Bonferroni-adjusted intervals appear in Fig. 4A.

Comparator	Comparator AUC	Difference	95% paired CI
P2Rank	0.7968	+0.0443	[+0.0162, +0.0724]
pLM-NN	0.8751	-0.0340	[-0.0701, -0.0006]
PocketMiner	0.7734	+0.0677	[+0.0328, +0.1013]

Across all 39 thresholds from 2% to 40%, the field's F1 remained directionally above P2Rank's, although pointwise intervals excluded zero at only 10 thresholds. pLM-NN remained above the field throughout, with intervals excluding zero at all 39 thresholds. This robustness curve supports a directional comparison but does not create 39 independent confirmatory tests.

3.3. *First locked external validation*

Table 3 and Fig. 4A report the predeclared external analysis. The field reached mean ROC-AUC 0.8411. Its paired advantage over P2Rank was +0.0443 [+0.0162, +0.0724], with a Bonferroni-adjusted interval of [+0.0102, +0.0792]; it was ahead on 42 of 57 units. This is the paper's principal confirmatory accuracy result.

The point-estimate deficit to pLM-NN persisted externally (-0.0340 [-0.0701, -0.0006]), although its multiplicity-adjusted interval [-0.0784, +0.0071] included zero. The advantage over PocketMiner was also replicated (+0.0677 [+0.0328, +0.1013]). At the matched top-9% budget, external F1 against P2Rank was unresolved (+0.0246 [-0.0224, +0.0708]), agreeing with the official-fold matched-threshold analysis. P2Rank abstained on 1 external unit; ranking analysis retained its residue scores, whereas undefined threshold metrics followed the declared missing-value rule.

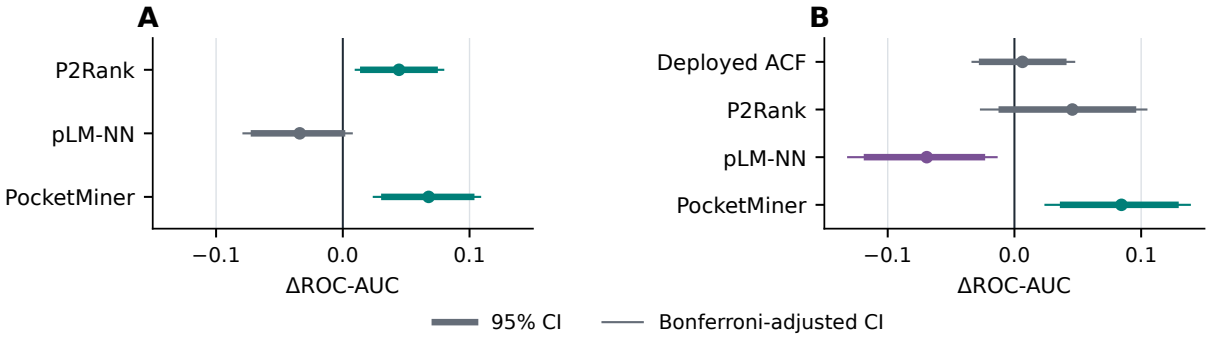


Fig. 4. Locked external evaluations. (A) Set A, 57 units: deployed ACF minus comparator. (B) Sets B+C, 45 cryo-EM units: expanded ACF minus comparator. Points are paired mean ROC-AUC differences; thick/thin lines denote 95%/Bonferroni-adjusted intervals (three comparisons in A, four in B). Positive favors the named ACF variant. Expanded-versus-deployed improvement is unresolved; evaluations are not pooled.

3.4. *Second external read of the expanded representation*

The expanded field reached mean per-unit ROC-AUC 0.7980 on the 45 cryo-EM units, compared with 0.7917 for the deployed field, 0.7523 for P2Rank, 0.7136 for PocketMiner, and 0.8672 for pLM-NN (Fig. 4B). The primary expanded-minus-deployed difference was +0.0063 [-0.0255, +0.0385], ahead on 27 units and behind on 18. Its interval contains zero and the training-fold estimate +0.0121; the read therefore does not confirm that the four added families improve the deployed representation.

Against P2Rank, the expanded field differed by +0.0457 [-0.0100, +0.0935] and remained unresolved after multiplicity correction ([-0.0264, +0.1041]). It exceeded PocketMiner by +0.0844 [+0.0384, +0.1271], with the adjusted interval [+0.0245, +0.1384] excluding zero. Conversely, pLM-NN remained ahead: the expanded-minus-pLM-NN difference was -0.0692 [-0.1163, -0.0257] (adjusted [-0.1312, -0.0141]). Thus, the expanded representation transfers signal relative to PocketMiner, without a resolved gain over the deployed field or P2Rank.

In the post-hoc comparison, the deployed 645-wire ACF and logistic regression achieved mean ROC-AUCs of 0.8411 and 0.8257 on Set A (57 units; ACF minus logistic: +0.0154, 95% CI [−0.0012, +0.0347]) and 0.7917 and 0.7968 on Sets B+C (45 units; −0.0051, 95% CI [−0.0415, +0.0333]), respectively. Neither interval excludes zero; these results establish neither ACF superiority nor statistical equivalence.

3.5. *Discarded atom-level structure carries training-fold signal*

The controlled family ladder distinguishes new structural information from re-encoding. A chemistry family that mainly resolves residue-identity collisions in the existing physicochemical wires changed mean ROC-AUC by only +0.0002 [-0.0007, +0.0010]. In contrast, atom-level backbone descriptors improved ROC-AUC by +0.0044 [+0.0033, +0.0055], side-chain conformational descriptors by +0.0048 [+0.0032, +0.0063], and void-connectivity descriptors by +0.0026 [+0.0012, +0.0039]. The combined backbone–side-chain family yielded +0.0072 [+0.0055, +0.0089], and adding void geometry produced +0.0095 [+0.0081, +0.0109]. All three

corresponding size-matched controls remained near zero; within-chain row permutation yielded negative gains for the backbone, side-chain, and void families.

The same screen identified a fourth source outside the deployed coordinate summary. Within-chain ranks and ratios derived from deposited temperature factors, occupancies, and alternate locations produced a +0.0070 [+0.0050, +0.0090] lift, driven primarily by temperature-factor quantities (+0.0063 [+0.0049, +0.0077]) rather than alternate-conformer quantities (+0.0005 [-0.0007, +0.0017]). These training-fold results support a specific representation-level interpretation: useful information remains in atom positions, empty-space connectivity, and structure-relative displacement records that are lost when each residue is reduced to a centroid and a contact neighborhood. They do not establish an accuracy gain for a revised detector on held-out data.

The repository contains the full family ladder and controls. At the manuscript snapshot, the expanded field's only held-out evaluation was Sets B+C; neither the official fold nor Set A had been scored with it. Later probes on the spent official fold remain exploratory.

3.6. *Architecture sensitivity, auditability, and cost*

Sensitivity models were fitted on the fit half and compared on the training selection half; each row reports its best gate from the stated training-only search, so the sweep is not uniformly a fixed-gate ablation. Three, four, and five quantization levels yielded ROC-AUC 0.7995, 0.8045, and 0.7996. Fan-out caps 16, 32, and 64 yielded 0.8018, 0.8045, and 0.8010. Ridge values 0.03, 0.1, and 0.3 yielded 0.8045, 0.8042, and 0.8023. At four levels, within-chain ranking reached 0.8045 versus 0.7905 for pooled ranking. The selected architecture ranked first on all 4 CryptoBench training folds and all 25 accession-disjoint half-splits. These results concern predictive performance, not stability of individual integer weights. On the selection half, the frozen field scored 0.7873 without the spatial gate and 0.8045 with it. This is not an independently held-out estimate of the gate benefit.

Exact recomputation of four residue scores from addresses, cell fractions, multiplicities, and the spatial gate agreed with stored scores to within 5×10^{-16}. Fig. 5 shows two rule-selected structural disagreement cases. In these audited cases, the coarse geometry-family margin between labeled hits and misses was +15.2, and the gate added +4.4. No coarse family separated true positives from false positives by more than 1.3, so the decomposition identifies where the score came from but is not a causal biological explanation.

On the matched Apple M4 environment, warm end-to-end median time was 0.298 s per chain for the field and 0.162 s for P2Rank: ACF took 1.84 times as long and used 2.95 times as much CPU time as P2Rank. Feature extraction consumed 0.212 s, or 71% of the field runtime, while table scoring consumed 0.0873 s. In a cold one-chain invocation, P2Rank took 5.98 times as long as ACF because of JVM startup overhead. Thus, compact model size, rather than faster served inference, is the supported efficiency advantage (1.79 MiB, 0.39 MiB compressed, versus 22.01 MiB for the P2Rank model).

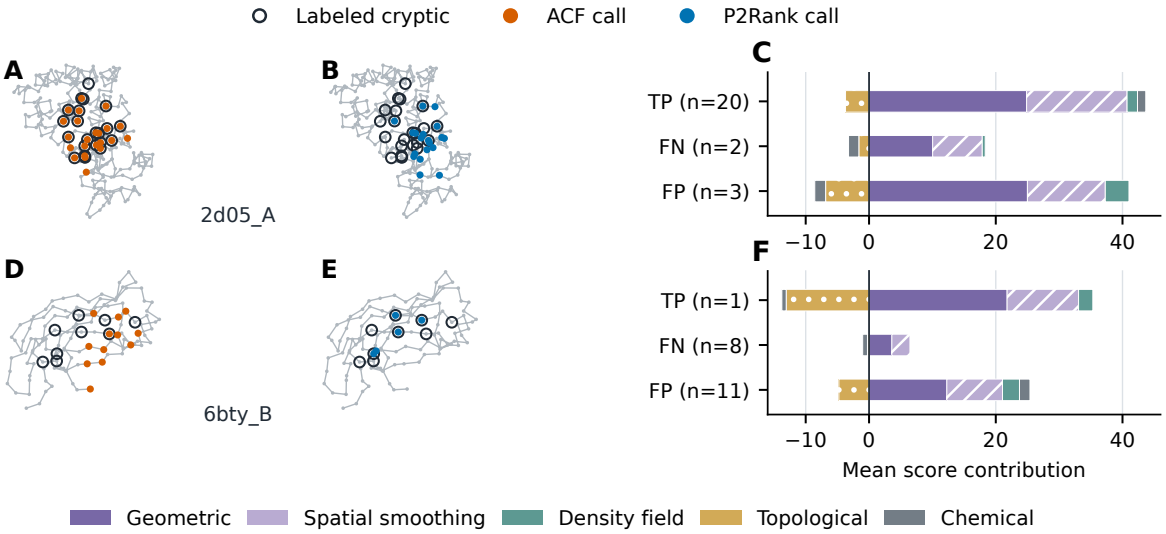


Fig. 5. Rule-selected disagreement cases: 2d05_A (A–C) and 6bty_B (D–F). A/D show ACF calls; B/E show P2Rank calls under deployed rules. Rings mark cryptic labels. Cα projections preserve the coordinate-defined viewing direction and equal spatial scale within each case. C/F show mean ACF contributions for true positives (TP), false negatives (FN), and false positives (FP); n counts residues. Cases illustrate numerical traceability, not confirmatory evidence or causal mechanisms.

4. Discussion

4.1. *Positioning the contribution*

ACF's accuracy advantage over P2Rank is specific to Set A; the cryo-EM comparison did not confirm gains from the expanded representation. Same-wire logistic comparisons likewise do not establish a benefit of the table readout. With pLM-NN stronger on the official fold and directionally stronger externally, ACF's supported contributions are numerical traceability and compact storage relative to P2Rank, not leading accuracy or faster warm inference.

4.2. *Limitations*

The official test fold was inspected repeatedly: the manuscript snapshot records three principal development rescoring rounds, 13 indexed reads (12 from this architecture lineage and one from another), and 12 scored architectures. The subsequently updated repository reports 20 scored architectures; the original counts describe the snapshot, not the current development history. Small official-fold margins are therefore descriptive rather than confirmatory. Set A mitigates but does not eliminate this concern: it contains only 57 units, excludes ambiguous intermediate-displacement cases through its labeling guard band, and may be easier than CryptoBench. The second external read contains only 45 scorable units and consists entirely of single-particle cryo-EM structures, so its wider intervals and modality shift limit interpretation. The decline of all coordinate-reading methods relative to the official fold is consistent with, but does not prove, reduced usable local geometric signal in these models. Both external collections are now spent and cannot confirm a future revision. Chronology must remain verifiable from the full repository commit graph.

The existing sensitivity and family-addition analyses do not establish that all 645 wires are essential, quantify their redundancy, or replace systematic deletion ablations. Nonzero table weights indicate use, not independent biological importance. Coordinate-perturbation robustness and stability of individual weights across training splits and pairing seeds remain untested here; bootstrap-seed stability addresses a different question. B-factor-derived features may reflect structure quality or experimental refinement as well as motion. Numerical auditability can support score inspection, but an actionable experimental benefit has not been demonstrated.

The analysis excludes 38 multi-chain units and does not establish performance at interfaces or in assemblies. Exact numerical decomposition is not the same as mechanistic or causal explanation, particularly because hits and false positives have similar coarse-family contributions. The target is a benchmark cryptic-residue label, not ligand identity, binding affinity, druggability, free-energy change, or biological efficacy. Broader locked datasets, a frozen chain-aware mapping, and prospective experimental or simulation follow-up are needed.

5. Reproducibility

Code, analysis protocols, and result artifacts are available at `https://github.com/shaneraphel/geoaudit-cryptobench`. The original results, macros, and deployed configuration are pinned to commit `5d1726a`. The artifact `data/cryptobench_apo/TABLE_FIELD.json` records wire names, pairs, counts, weights, propensity, operating point, and code/manifest hashes. The manuscript uses `paper/frozen_numbers.tex` from this snapshot. Defining modules are `wide_descriptors.py`, `table_bank.py`, and `table_field.py` under `src/pocket_bench/methods/`. The post-hoc logistic protocol and results are in the companion repository (commit `6c5a640`). Pinned versions distinguish manuscript artifacts from later development.

`make recompute` regenerates metrics from archived predictions; `make verify` checks artifacts. Inference and compilation require the frozen inputs, caches, baseline binaries/weights, and environment. `make extorder` checks lock/read ancestry in the retained Git history; this supports chronology but cannot exclude unrecorded reads.

Runtime measurements used Apple M4, macOS 26.2 (arm64), Python 3.12.8, NumPy 2.5.1/Accelerate, OpenJDK 17, and P2Rank 2.5.1. The record separates warm/cold wall time, CPU time, threading, extraction, scoring, and peak RSS (1180 MiB). These results are environment-specific; artifact checks are not experimental reruns. This work received no funding and used only public data (sources in Methods). X.W. is employed by Pfizer; the authors declare no competing interests related to this work.

6. Conclusion

ACF provides compact, numerically traceable structure-based prediction without homology or template searches at inference. On Set A, its mean paired ROC-AUC advantage over P2Rank was 0.0443 (adjusted 95% CI [0.0102, 0.0792]). Official-fold ROC-AUC and matched-budget F1 differences against P2Rank remained unresolved; cryo-EM evaluation did not confirm gains from added features. These findings support an auditable supervised predictor, without establishing lookup-table superiority or a causal explanation of pocket opening.

References

1. S. Vajda, D. Beglov, A. E. Wakefield, M. Egbert and A. Whitty, Cryptic binding sites on proteins: definition, detection, and druggability, *Current Opinion in Chemical Biology* **44**, 1 (2018).
2. P. Cimermancic, P. Weinkam, T. J. Rettenmaier, L. Bichmann, D. A. Keedy, R. A. Woldeyes, D. Schneidman-Duhovny, O. N. Demerdash, J. C. Mitchell, J. A. Wells, J. S. Fraser and A. Šali, CryptoSite: Expanding the druggable proteome by characterization and prediction of cryptic binding sites, *Journal of Molecular Biology* **428**, 709 (2016).
3. D. Beglov, D. R. Hall, A. E. Wakefield, L. Luo, K. N. Allen, D. Kozakov, A. Whitty and S. Vajda, Exploring the structural origins of cryptic sites on proteins, *Proceedings of the National Academy of Sciences of the United States of America* **115**, E3416 (2018).
4. G. R. Bowman and P. L. Geissler, Equilibrium fluctuations of a single folded protein reveal a multitude of potential cryptic allosteric sites, *Proceedings of the National Academy of Sciences of the United States of America* **109**, 11681 (2012).
5. V. Le Guilloux, P. Schmidtke and P. Tufféry, Fpocket: an open source platform for ligand pocket detection, *BMC Bioinformatics* **10**, p. 168 (2009).
6. T. A. Halgren, Identifying and characterizing binding sites and assessing druggability, *Journal of Chemical Information and Modeling* **49**, 377 (2009).
7. A. Volkamer, D. Kuhn, T. Grombacher, F. Rippmann and M. Rarey, Combining global and local measures for structure-based druggability predictions, *Journal of Chemical Information and Modeling* **52**, 360 (2012).
8. D. Kozakov, L. E. Grove, D. R. Hall, T. Bohnuud, S. E. Mottarella, L. Luo, B. Xia, D. Beglov and S. Vajda, The FTMap family of web servers for determining and characterizing ligand-binding hot spots of proteins, *Nature Protocols* **10**, 733 (2015).
9. J. Jiménez, S. Doerr, G. Martínez-Rosell, A. S. Rose and G. De Fabritiis, DeepSite: protein-binding site predictor using 3D-convolutional neural networks, *Bioinformatics* **33**, 3036 (2017).
10. M. M. Stepniewska-Dziubińska, P. Zielenkiewicz and P. Siedlecki, Improving detection of protein–ligand binding sites with 3D segmentation, *Scientific Reports* **10**, p. 5035 (2020).
11. R. Aggarwal, A. Gupta, V. Chelur, C. V. Jawahar and U. D. Priyakumar, DeepPocket: ligand binding site detection and segmentation using 3D convolutional neural networks, *Journal of Chemical Information and Modeling* **62**, 5069 (2022).
12. R. Krivák and D. Hoksza, P2Rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure, *Journal of Cheminformatics* **10**, p. 39 (2018).
13. A. Meller, M. Ward, J. Borowsky, M. Kshirsagar, J. M. Lotthammer, F. Oviedo, J. Lavista Ferres and G. R. Bowman, Predicting locations of cryptic pockets from single protein structures using the PocketMiner graph neural network, *Nature Communications* **14**, p. 1177 (2023).
14. A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma and R. Fergus, Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences, *Proceedings of the National Academy of Sciences of the United States of America* **118**, p. e2016239118 (2021).
15. V. Škrhák, M. Novotný, C. P. Feidakis, R. Krivák and D. Hoksza, CryptoBench: cryptic protein–ligand binding sites dataset and benchmark, *Bioinformatics* **41**, p. btae745 (2025).
16. C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, *Nature Machine Intelligence* **1**, 206 (2019).
17. M. Steinegger and J. Söding, MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets, *Nature Biotechnology* **35**, 1026 (2017).
18. S. K. Burley, H. M. Berman, C. Bhikadiya, C. Bi, L. Chen, L. Di Costanzo, C. Christie, K. Dalenberg, J. M. Duarte, S. Dutta *et al.*, RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology

and energy, *Nucleic Acids Research* **47**, D464 (2019).
19. J. Kyte and R. F. Doolittle, A simple method for displaying the hydropathic character of a protein, *Journal of Molecular Biology* **157**, 105 (1982).
20. A. A. Zamyatnin, Protein volume in solution, *Progress in Biophysics and Molecular Biology* **24**, 107 (1972).
21. T. Saito and M. Rehmsmeier, The precision–recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets, *PLOS ONE* **10**, p. e0118432 (2015).
22. B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap* (Chapman & Hall/CRC, New York, 1993).