

Seq2DomML: Protein Domain Boundary Prediction from Bacterial Protein Sequences Using a Machine Learning Framework

Alana Monks

Wilfrid Laurier University

Azam Asilian Bidgoli

aasilianbidgoli@wlu.ca

Wilfrid Laurier University

Michael Douglas Leo Suits

Wilfrid Laurier University

Research Article

Keywords: Protein construct design, Protein domain annotation, Protein domain prediction, Long short-term memory model, Feature selection, Machine learning

Posted Date: August 27th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10359584/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at BMC Bioinformatics on October 6th, 2026. See the published version at <https://doi.org/10.1186/s12859-026-06666-y>.

Seq2DomML: Protein Domain Boundary Prediction from Bacterial Protein Sequences Using a Machine Learning Framework

Alana Monks¹, Azam Asilian Bidgoli^{2*}, Michael D. L. Suits^{1*}

^{1*}Chemistry and Biochemistry, Wilfrid Laurier University, 75 University Avenue West, Waterloo, N2L3C5, Ontario, Canada.

²Computer Science and Physics, Wilfrid Laurier University, 75 University Avenue West, Waterloo, N2L3C5, Ontario, Canada.

*Corresponding author(s). E-mail(s): abidgoli@wlu.ca; msuits@wlu.ca;
Contributing authors: monk2383@mylaurier.ca;

Abstract

Purpose: The identification of protein domain boundaries is important for designing stable single-domain constructs for protein structural solution. For proteins without experimentally determined structures, boundary selection often relies on conserved domain annotations, predicted structures, or iterative truncation screening. Seq2DomML is developed as a sequence-based machine learning framework for predicting residue-level domain partitions in bacterial proteins without requiring structural coordinates or homologous domain assignments as input.

Methods: Structural domain annotations from the Evolutionary Classification of Domains (ECOD) database were converted into binary residue labels, including residues within annotated domains and domain boundaries, linkers, and terminal flanking regions. Each residue was initially encoded using 2,389 amino acid identity, positional, physicochemical, and interaction features. The feature space was reduced through correlation-based pruning and multi-objective evolutionary feature selection, and the selected features were used to train a bidirectional long short-term memory (LSTM) model.

Results: On the held-out test set, Seq2DomML was evaluated and compared with established tools including InterPro, Pfam, and NCBI CDD. It achieved a macro F1-score of 0.7445, within-domain F1-score of 0.9282, and domain count mean absolute error of 0.9158, consistently outperforming the competing methods.

Conclusion: Seq2DomML demonstrates that sequence-derived residue features can support prediction of structurally defined domain partitions without direct structural input. The model provides a complementary approach to conserved domain annotation tools and may help prioritize candidate splice regions for construct design, particularly when structural or homologous evidence is limited.

Keywords: Protein construct design, Protein domain annotation, Protein domain prediction, Long short-term memory model, Feature selection, Machine learning

1 Introduction

Protein domains are independently folded structural units that play fundamental roles in protein function and organization. Accurate identification of domain boundaries is therefore important for structural biology, protein annotation, and protein engineering. In structural determination workflows, identifying appropriate domain boundaries is also a critical step in protein construct design. Beyond structural biology, reliable domain boundary prediction also facilitates the functional interpretation of proteins, comparative genomics, and protein engineering by identifying independently folded functional units that can be studied or manipulated individually.

Protein structure determination is dependent not only on utilizing an appropriate experimental method, but is also contingent on the design of an appropriate protein construct for expression and solubility. Typically, the most experimentally manageable construct is not the full length protein, but a stable region that preserves an independently folded domain unit whilst excluding flexible linkers, disordered termini, or neighboring domains that may introduce conformational heterogeneity [1]. Identifying the appropriate domain boundaries is therefore a central step in construct design, yet it remains difficult when no experimentally determined structure is available. Furthermore, domain boundaries cannot always be inferred from local sequence composition alone, as domain formation is influenced by long-range structural interactions and evolutionary constraints. However, accurate sequence-based prediction of domain boundaries remains a challenging problem, particularly for large multidomain proteins or proteins lacking experimentally determined structures [2].

Experimental workflows thereby often require creating and screening many truncation constructs in order to identify the optimal positions to preserve the structure, function, and to yield a successful crystal hit of a protein domain [3]. The main drawback with this approach is that the number of truncated constructs screened increases the time and cost of a structural solution workflow, hence why supplementary tools are often required to make predictions about such domain positions for splicing.

Existing sequence-based domain annotation resources, such as Pfam [4] and Interpro [5], identify conserved domains and protein families primarily through sequence homology and curated domain models. While highly effective for well-characterized proteins, they are prone to not recognizing domains where there are few well-annotated homologs [5]. Consequently, there is a need for methods that can predict

domain boundaries directly from a amino acid sequence without relying on structurally characterized homologs. Such methods are particularly valuable for recently discovered proteins, poorly characterized protein families, and proteins whose domain organization is not represented within existing annotation databases.

To address this challenge, this paper proposes Seq2DomML, a sequence-based machine learning (ML) framework for residue-level protein domain boundary prediction. Rather than transferring domain annotations from homologous proteins, Seq2DomML learns domain partitioning directly from structurally characterized bacterial proteins and predicts domain boundaries using only sequence-derived information. The framework is designed to reduce the trial and error involved in truncation screening by producing a tool that helps prioritize splice sites preserving independently folded domains whilst excluding flexible or heterogeneous regions [6]. The trial and error can therefore be reduced in the process of solving structurally uncharacterized proteins when a tool for predicting such domain positions is utilized. Decades of structural biology research have shown that multi-domain proteins, proteins with excess loop regions, and proteins with unrestrained linkers are considered more difficult to crystallize. Motivated by this challenge, Seq2DomML provides a sequence-based framework for predicting residue-level domain boundaries that can guide construct design, particularly for proteins lacking experimentally determined structures or well-annotated homologous domains.

The proposed framework utilizes structural domain annotations derived from the Evolutionary Classification of Domains (ECOD) database [7], to convert the annotations into binary residue-level labels across a dataset of structurally characterized bacterial protein sequences from the Protein Data Bank (PDB) [8]. Following the target labeling, thousands of sequence-derived physicochemical features are extracted from the GenomeNet Database [9] for each residue in every protein sequence, which are then reduced to an optimized subset through two-stage feature selection. The final ML model is subsequently trained via a bi-directional long short-term memory (LSTM) network, enabling residue-level domain boundary predictions to be made on sequences that lack structural characterization or well annotated homologous domains.

Through facilitating the identification of stable, independently folded domains for protein construct design, Seq2DomML aims to tackle a notoriously difficult component of structural solution workflows by assisting in the identification of single unit domains. This task is particularly valuable for proteins lacking experimentally determined structures or reliable homologous domain annotations, thereby allowing structural solution to extend beyond well studied proteins.

2 Background and Literature Review

2.1 Protein Domains

In the context of this model, a protein domain is defined as a contiguous sequence region corresponding to an independently folded structural unit. Protein domains are commonly understood as modular units of protein structure and function that may fold with some degree of independence from the rest of a polypeptide chain [10]. Large proteins are often composed of multiple domains connected by intervening regions

defined as linkers [11]. Since the positive class does not represent a domain itself, but rather, the residues belonging to sequence contexts surrounding or separating domains, this class semantically covers linkers, domain boundaries, and terminal flanking regions. This distinction is important because the model is not simply predicting whole domain membership, but identifying the sequence positions most relevant to domain partitioning.

Flanking regions in this study are defined as terminal sequence segments located outside the first or last annotated domains of a protein chain. These regions may include disordered termini, affinity tag adjacent sequences, or flexible extensions that are not part of the independently folded domain units being targeted for structural solution. In construct design, such regions are often removed because terminal flexibility and disorder can reduce sample homogeneity and therefore interfere with stable packing interactions [12].

The inclusion of flexible linkers in protein constructs can lead to increased conformational heterogeneity across a protein sample, largely due to the uncontrolled multiple relative orientations it can possess [13], which additionally can hinder protein packing into crystal lattices, and ultimately the resolution of diffraction images.

Flexible loops and linkers can also lack packing constraints through sampling many backbone torsions, enabling broad conformational variability that makes rigid crystal packing much more difficult to achieve [14]. In tangent, the deletion of linkers in protein samples can significantly improve the formation of crystals due to a reduction in conformational heterogeneity [12] thereby enhancing crystal lattice formation.

On the other hand, multi domain proteins connected to one another often populate multiple inter domain arrangements [12] which increases conformational heterogeneity, thereby disrupting the formation of a well-ordered crystal lattice. Not only do multi domain proteins exhibit more conformational heterogeneity, but they introduce more complexity for achieving the optimal experimental conditions that will result in successful crystallization outcomes for a variety of reasons. Additionally, multidomain proteins may be difficult to stabilize under a single set of solution conditions because the physicochemical activity from the pH, salt, or additives that preserve the stability and homogeneity of one region may not be optimal for another, subsequently increasing partial unfolding and aggregation whilst reducing the number of productive crystallization hits due to the heterogeneity within the environmental preferences of distinct domains [15].

As a result of the described behaviors observed in protein crystallization, samples with flexible linkers, flanking regions, and/or multiple domains, become undesirable for crystallographers aiming to achieve clean crystal packing when selecting constructs for structural solution. Furthermore, crystallography workflows commonly pursue domain isolation as truncating multi domain proteins to individual functional domain units becomes an effective approach to improving the crystallization success of a protein sample [16].

2.2 Long Short Term Memory (LSTM) Models in Protein Sequence Prediction Tasks

The selected model architecture for the workflow is a Bidirectional Long Short Term Memory (LSTM) model [17], due to its ability to model both local residue neighborhoods and long range relationships along the sequence.

LSTMs are a type of recurrent neural network designed for sequential data, where information from earlier positions in a sequence can influence later predictions. LSTM models maintain an internal memory state that is updated as the sequence is processed, and mediated through learned three main gating mechanisms, including the input gate, the forget gate, and output gate [18]. This architecture was originally developed to reduce an observed problem of the vanishing gradients seen in conventional recurrent neural networks [17], thereby enhancing its applicability for time-series or interval based datasets.

Bidirectional LSTM variants are becoming increasingly common in proteomics based prediction tasks largely due to their utility in integrating contexts from both N termini to C termini as well as C termini to N termini [19]. This becomes especially valuable for prediction tasks such as the ones in this study where the identity of a residue depends largely on its surrounding sequence patterns at multiple scales. There are a wide variety of successful applications that inspired the decision to select the LSTM model architecture for Seq2DomML. For example, CLPred’s use of the LSTM has shown success for predicting whether or not k -mers across a protein sequence will yield diffraction quality crystals [20]. Other successful applications of LSTM architectures include protein function prediction [21] and protein-protein interaction binding site prediction tasks [22].

Overall, for residue level tasks such domain boundary identification, LSTMs naturally support samples formatted as sequences in learning, where an output label is produced at each sequence position rather than a single label per protein. It was therefore evident that this model architecture would be valuable for Seq2DomML’s prediction task, as decisions at a given residue depend both on its local neighborhood and on distal sequence contexts.

2.3 Multi-Objective Feature Selection

Multi-objective feature selection refers to an optimization problem where the goal is to select an optimal set of features to improve performance of a machine learning task (e.g., classification) across multiple objectives that are often simultaneously conflicting. Rather than performing optimization to improve a single score, multi-objective optimization ranks or preserves candidate solutions according to tradeoffs across objectives. A solution is considered non-dominant when no other solution performs better across all objectives and strictly better in at least one objective, allowing the algorithm to approximate the Pareto front for tradeoff solutions [23].

To solve such optimization problems, evolutionary algorithms, inspired by biological evolution, are a highly effective class of optimization algorithms [24]. During the optimization, a population of candidate solutions is iteratively improved through selection, crossover, and mutation operators. Candidate solutions are evaluated according

to their fitness, and stronger performing solutions are preferentially carried forward as surviving populations across generations [23].

Non-Dominated Sorting Genetic Algorithm (NSGA-III) is an evolutionary algorithm designed for multi-objective optimization problems, where the goal is to improve performance across multiple objectives that are often simultaneously conflicting. Rather than performing optimization to improve a single score, NSGA methods rank candidate solutions according to Pareto dominance. A solution is considered non-dominant when no other solution performs better across all objectives, allowing the algorithm to approximate the Pareto front for tradeoff solutions.

NSGA-III is a widely used implementation of this framework that uses reference points distributed across the objective space to preserve diversity among candidate solutions. Candidate solutions are sorted into non-dominated fronts, and the top-ranking fronts are preferentially carried forward as surviving populations across generations. The reference-point-based selection mechanism allows NSGA-III to maintain a more diverse set of candidate feature subsets across competing objectives [25], hence why this method was chosen.

The Non-Dominated Sorting Genetic Algorithm (NSGA-III) [25] is a many-objective evolutionary algorithm designed to solve optimization problems involving multiple conflicting objectives. It first ranks candidate solutions using non-dominated sorting, where a solution is considered non-dominated if no other solution is at least as good in all objectives and strictly better in at least one objective. The set of non-dominated solutions forms an approximation of the Pareto front, representing different trade-offs among the objectives. To maintain diversity, NSGA-III associates solutions with a predefined set of reference points and selects individuals to achieve a well-distributed approximation of the Pareto front. This reference-point-based selection enables NSGA-III to maintain both convergence and diversity more effectively than traditional crowding-distance-based approaches, particularly in many-objective optimization problems.

2.4 Existing Sequence-based Domain Prediction Approaches

ECOD is known as a gold standard tool, as it relies on the atomic coordinates of existing solved structures, and homology modeling to feed into as input for its domain annotations, and thereby classifies domains based on structural and homologous characterization. Some other tools that are commonly used for predicting domains for structural solution workflows include Pfam [4], Interpro [5], and NCBI CDD [26]. Although, unlike ECOD, these tools do not rely on structural input in order to make their predictions, making them more applicable for structurally uncharacterized sequences, yet less confident overall.

Pfam is a database of protein families/domains in which each entry is represented by a seed multiple sequence alignment and a profile hidden Markov model (profile HMM). Pfam is still a sequence-based annotation method at query time, but modern Pfam curation increasingly uses AlphaFold predictions [27] as structural evidence to refine families and improve domain boundary predictions [4]. The 2025 Pfam update reports that Pfam release 37.0 contains 21,979 families and 709 clans, with 76.3%

sequence coverage and 48.8% residue coverage across the UniProtKB [28] reference proteomes.

NCBI CDD is a collection of multiple sequence alignment derived domain and protein family models, represented as position specific scoring matrices (PSSMs) [26]. NCBI CDD is a valuable comparator because it performs sequence to domain annotations without needing a query structure, but conceptually it is still a library search system based on curated domain models. They report 59,693 proteins and protein domain models in v3.20, including 19,178 Pfam derived samples, 18,882 CDD in house curated models, and 4,541 superfamily clusters as well as 42,937 site annotations across 15,943 of the staff curated models [26].

InterPro is an integrative protein sequence classification resource. It does not rely on one modeling strategy; instead it integrates signatures from multiple member databases, including Pfam [4], CDD [26], SMART [29], PANTHER [30], PROSITE [31] and others. InterPro consolidates overlapping evidence from multiple databases into a single annotation framework, attaches InterPro entries, abstracts, and often GO terms, and now provides representative domains chosen to maximize coverage while minimizing overlap [5]. InterPro is best thought of as a meta-resource with its domain annotation framework and is not exclusively a single homogeneous predictor. Its output quality depends on the underlying member database signatures.

ECOD is widely used as a domain classification resource built around proteins with experimentally determined 3D structures, providing domain assignments and boundary definitions on PDB chains [7]. It is best described as the structural reference standard used to derive domain annotations, because it classifies domains using a combination of sequence and structural data, and its hierarchy is built to emphasize evolutionary relationships, including distant homology, rather than only local sequence similarity or shared topology [7]. Although, because ECOD is fundamentally anchored to existing structural coordinates, it is limited in its utility for unsolved protein sequences.

Some recent work with The Domain Parser For Alpha Fold Models (DPAM) [32], has demonstrated it's ability to fill in this gap using predicted Alpha Fold structures along with sequences to make the domain predictions, although, the confidence of such domain predictions is thereby limited when the structure predictions from Alpha Fold possess a lower confidence. On average this endeavor has shown that they could provide positive classification for approximately 90% of residues across 48 distinct proteomes using ECOD proteomes [33]. Although, they also show that a substantial part of the proteomes still land outside straightforward ECOD assignments. Of the residues DPAM attempted, about 66% were confidently assigned to an ECOD homologous group, while 4% were judged globular but still could not be confidently assigned to ECOD, 12% were called flexible linker or disordered regions, and another 12% were low confidence AlphaFold regions that were not assigned as domains.

Although Pfam, InterPro, and NCBI CDD provide valuable domain annotations, their performance is inherently tied to the availability and quality of homologous sequence evidence. For proteins from poorly characterized families, or for sequences with few well studied homologs, these tools may return incomplete annotations, overly broad domain matches, missing boundaries, or no confident domain assignment at

all [34], which limits the success of domain isolation in crystallography, Cryo-EM, and Nuclear Magnetic Resonance (NMR) towards already-well studied proteins. The homology reliance of these tools therefore presents a strength for domain isolation when there are sufficient homologs, yet remains weaker for poorly studied protein families posing constraints in the structural solution process for each approach.

A key difficulty in X-ray diffraction is that splice boundaries must be chosen to retain a complete, independently folding unit [10]. When boundaries cut into structured elements or disrupt domain integrity, the resulting construct can misfold, lose function, or become unstable. This process can therefore be high stakes as a single position can separate crystallizable from non crystallizable variants, or can be the difference between a functional domain unit, as opposed to a dysfunctional domain unit from a partial domain construct [3]. Moreover, domains can also be made up of non-contiguous sequence elements, meaning that these domains would be defined from multiple sequence segments. For example, in Cryo-EM, flexible regions and mobile domains often blur density, reduce local resolution, and make classification and refinement harder. Sequence guided removal of such flexible regions or separation of domain units would therefore improve particle homogeneity, and increase local map interpretability [35]. For NMR based structural studies, careful construct optimization is often necessary to improve solubility, folding behavior, and spectral quality. By suggesting domain consistent splice regions from sequences alone, an annotation tool may help prioritize constructs that better preserve independently folded units while excluding destabilizing flanking segments [36].

In contrast to the described annotation tools, the method proposed in Seq2DomML does not operate based on domain conservation, sequence alignments, experimentally solved structural inputs, or computationally predicted structural inputs, and instead relies exclusively on an amino acid sequence as input. The trained model then learns sequence based signals associated with domain residues and non-domain residues, allowing predictions to be made directly from amino acid sequence features alone. This distinction between a homology based approach and Seq2DomML’s model is important because the model is not intended to replace curated domain databases yet instead can be used to provide an additional prediction layer for cases where homologous domain evidence is low, structural coordinates are unavailable, or predicted structures possess too much uncertainty for other tools to make confident domain annotations.

3 Methods

The overall aim of this workflow is to develop a sequence-based model that predicts residue-level domain labels directly from amino acid sequence features, capable of providing domain partitioning guides for structurally uncharacterized sequences. As summarized in Figure 1, the workflow consists of three main stages. First, bacterial X-ray crystallography chains were collected from the PDB, then filtered, deduplicated, and matched to ECOD annotations. These ECOD annotations were then converted into binary residue labels, where residues within annotated domains were encoded as domain residues (label 0) and residues outside annotated domains were encoded as non-domain residues (label 1). Next, feature extraction was performed to derive

Methods Overview



Fig. 1: Overview of the Seq2DomML Methodological Workflow. The workflow consists of three main stages. First, bacterial protein chains are collected, filtered, and converted into residue-level labels using ECOD annotations. Second, sequence-derived features are extracted for each residue and used to train a bidirectional LSTM model. Finally, the feature space is optimized through correlation-based pruning and evolutionary feature selection, and the resulting model is evaluated against Pfam, NCBI CDD, and InterPro.

2,389 features for each residue in each sequence. These features included amino acid identities, positional descriptors, amino acid indices, and interaction matrix features. Finally, the feature space was reduced in the first phase of feature selection through correlation-based pruning and subsequently further optimized using multi objective evolutionary feature selection.

The following sections describe each methodological phase in greater detail, including dataset construction, target label encoding, feature extraction, model training, feature reduction, evolutionary feature selection, and the final model evaluation.

3.1 Dataset Construction and Target Label Encoding

The dataset construction began by acquiring all chains from the PDB [8] that were proteins annotated within the Kingdom of Bacteria, had a refinement resolution of ≤ 2.5 Angstroms, and were solved with the experimental method of X-ray Diffraction. Additionally, all proteins from the taxonomic class of *Escherichia-coli* were removed due to the overwhelming abundance of this species posing a threat of species specific bias in Seq2DomML’s training and evaluation datasets. Next, in order to reduce extreme heterogeneity within the dataset, all membrane proteins were removed due to the highly distinct nature of their protein domains, as well as the notably diverse

approach required for structural solution in this specific class of proteins [37]. Lipoproteins are also excluded from this dataset because in bacterial species, they are most often embedded or closely associated with membrane proteins [38].

After the initial pre-processing stage, 45,680 protein chains remain. The October 2025 archive of ECOD annotations is then utilized to acquire the target labels, which successfully covers 43,278 annotated sequences out of the dataset. The next step in filtering involves a quick pass to remove all sequence chains with non-standard amino acids, such as X in order to allow amino acid features to be extracted.

Following the filtering steps, a deduplication process is performed in order to remove sequences that are near duplicates of one another in order to preserve dataset diversity and reduce bias. This is an important step of protein based machine learning tasks because data leakage can occur when overly similar protein chains are used or if mutant variations of one another are present in a training and evaluation set [39], which would therefore produce overoptimistic generalization. This step is performed by removing sequences using 90 percent global sequence identity clustering. Firstly, candidate pairs were prefiltered by a sequence length difference of less than 10 residues and greater than 90 percent identity across the first 50 amino acids, before being globally aligned using Biopython’s PairwiseAligner tool. Then, the pairs exceeding the 90 percent global identity threshold were grouped into redundancy clusters, and one representative chain was randomly selected to be retained from each cluster. After this step, the full dataset consists of 7,599 unique chains. Table 1 provides a summary of the data acquisition and filtering process.

After the preprocessing stage, ECOD annotations of the 7,599 protein chains are converted into binary residue-level labels that distinguish within domain residues from non-domain residues, across the length of each protein’s sequence. The within domain class (the negative label) includes all residues that are included within an annotated domain, whereas the non-domain class (the positive label) includes all residues outside of a labeled domain, thereby referring to domain boundaries, linkers, and flanking segments. Due to the inherent label imbalance present amongst the two labels, an initial stretching step is executed in order to inflate the length of domain boundaries to spanning nine residues in length. This allows the model to possess more exposure to the minority label, as well as ensures that flanking and linker regions were not under-represented in length. Additionally, it reflects the biological observation that domain boundaries represent transition regions rather than precisely defined single-residue positions. Furthermore, because protein construct design typically tolerates small shifts in splice-site selection around domain boundaries, representing these regions as short contiguous segments better aligns the training labels with their intended downstream application. Fig. 2 represents an overview of this process. Finally, the constructed dataset is then split into train, test, and validation sets for model construction and evaluation. The training dataset consists of 5,700 samples, the validation set consists of 664 samples and the test dataset consists of 1,235 samples.

3.2 Feature Extraction

With the target-label encoded dataset, features are extracted to encode residue-specific signals across all positions of the sequences. The features include one hot encodings

Table 1: Construction and Filtering of the Bacterial Protein Sequence Dataset. Summary of the sequential inclusion and exclusion steps used to construct the final dataset from bacterial protein chains solved by PDB X-ray crystallography. The first column identifies each stage of dataset preparation, the second specifies the criterion applied at that stage, and the third reports the number of sequence chains remaining after the criterion was applied. Chains were filtered by structural resolution, taxonomic classification, protein type, availability of an ECOD domain annotation, and sequence similarity. Deduplication at 90 percent global sequence identity produced the final dataset of 7,599 sequence chains used for target label construction and subsequent model development.

Step	Inclusion / Exclusion Criteria	Remaining Sequence Chains
Initial bacterial PDB X-ray chains	Resolution ≤ 2.5 Å	71,942
Removal of <i>Escherichia coli</i> sequences	Taxonomic classification is not <i>Escherichia coli</i>	60,791
Removal of membrane and lipoproteins	Does not contain “lipoprotein”, “lipo”, or “membrane”	45,680
ECOD annotation matched	Domain annotation present in ECOD archive from October 2025	43,278
Deduplication at 90% global sequence identity	Retain one representative from each similarity cluster	7,599

of all 20 amino acids, relative positions of each residue, amino acid indices, and interaction matrices as seen in the AA index database [40]. In total, 2,389 features are extracted for each residue.

The one-hot amino acid encodings were used to preserve the categorical identity of each residue, allowing the model to distinguish between the 20 standard amino acids at each sequence position. However, because one-hot encoding does not directly describe the biochemical similarities or differences between amino acids, amino acid index features were additionally included in order to represent residue level physicochemical properties, including properties such as the hydropathy, charge, size, polarity, flexibility, and secondary structure preferences. These properties are relevant to domain boundary prediction because such physicochemical tendencies are observably distinct in different structural contexts, particularly when located within stable domain interiors, flexible linkers, solvent exposed loops, or terminal flanking regions. Positional features were included because domain boundary and flanking regions are not distributed randomly across protein sequences, and terminal regions in particular may have different label behavior than internal domain boundaries. Interaction matrix features were also included to encode pairwise amino acid relationships such as statistical contact potentials and substitution potentials, allowing the model to infer information about how a residue may relate to a neighboring residue. Together, these feature



Fig. 2: Example of Residue Label Encoding and Boundary Expansion for PDB 1H6C Chain B. The raw amino acid sequence is shown together with the initial and expanded residue labels derived from the three ECOD annotated domains. In the initial encoding, label 0 residues shown in black represent positions within annotated domains, while label 1 residues shown in pink represent terminal flanking regions, linkers, and single residue domain edges. Each continuous label 1 region is then expanded to a minimum length of nine residues, as shown in the final row, to improve representation of the minority class and increase the visibility of domain boundary regions during model training.

categories allow each residue to be represented using both simple amino acid identity and broader biochemical context. Table 2 provides the summary and counts of all extracted feature types.

3.3 Model Training

The LSTM is trained on the binary labels across training set of sequences so that the model learns to predict sequence of binary labels using the amino acid features across each position of the input sequence. The validation set is used to select the best-performing model from experimental trials, and the test set is held out for final model evaluation and comparison across other tools. After the final epoch of training, a smoothing function is performed before evaluation that allows the model’s non-domain

Table 2: Overview of the Feature Space Used in Seq2DomML. Breakdown of the four feature categories used to encode each amino acid position within the input protein sequences. The first column identifies the feature category, the second summarizes the sequence or biochemical information represented by that category, and the third reports the number of individual features contributed to each residue representation. The feature space includes 20 amino acid identity indicators, three positional descriptors, 566 physicochemical amino acid indices, and 1,800 interaction matrix features, producing a total of 2,389 features per residue before feature reduction and selection.

Feature Type	Description	Count
One-hot amino acid encoding identities	One-hot encodings of each of the 20 standard amino acids	20
Positional features	Relative position across sequence, and relative distance to N-termini and C-termini	3
Amino acid indices	Physicochemical indices [40]	566
Interaction matrices	Interaction matrices, contact potentials, and substitution matrices derived from [40]	1,800

residue predictions to remain biologically plausible, and consistent with the stretching rule utilized within the target label encoding. Predicted non-domain stretches that were separated by short domain residue stretches were merged when the adjacent gap was less than or equal to 9 residues. This smoothing step was not used during training, but was applied only after prediction to reduce biologically implausible fragmentation of domain boundary predictions. The 9 residue minimum was also consistent with how the target label encoding was performed, in which domain boundaries were deliberately stretched out to 9 residues.

3.4 Handling Label Imbalance through Loss Customizations

Despite earlier steps to inflate the presence of the minority label by stretching out the length of domain boundaries, the label imbalance still remains significant enough to require further handling for promoting the detectability of domain boundaries. To address this issue, effective class weights were incorporated into the binary cross-entropy loss during LSTM training, which has been shown to reinforce the impact of a minority label in loss functions for imbalanced datasets [41]. This approach assigns larger weights to underrepresented classes while accounting for the diminishing contribution of repeated samples within the majority class, thereby increasing the influence of minority-class residues during optimization.

For each residue with true binary label $y_i \in \{0, 1\}$ and predicted probability $\hat{y}_i$, the weighted binary cross-entropy loss is defined as

$$\mathcal{L}_{\text{BCE}} = -\alpha_{y_i} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)],$$

where α_{y_i} denotes the class-specific weight associated with the true residue label. In this study, $y_i = 0$ corresponds to domain residues, whereas $y_i = 1$ corresponds

to boundary residues, including domain boundaries, linkers, and terminal flanking regions. The class-specific weight α_{y_i} was computed using the effective number of samples approach. First, the effective number of samples for each class c was calculated as

$$E_{n_c} = \frac{1 - \beta^{n_c}}{1 - \beta},$$

where n_c denotes the number of training residues belonging to class c , and β is the effective number weighting parameter. The raw class-balanced weight was then computed as $w_c = \frac{1}{E_{n_c}}$. To maintain a consistent overall loss magnitude, the class weights were normalized such that they summed to two:

$$\tilde{w}_c = \frac{2w_c}{w_0 + w_1}.$$

Finally, the residue-specific weight used in the loss was assigned according to the true residue label:

$$\alpha_{y_i} = y_i \tilde{w}_1 + (1 - y_i) \tilde{w}_0.$$

Consequently, boundary residues received the normalized weight $\tilde{w}_1$, whereas domain residues received $\tilde{w}_0$.

In addition to the effective class weights, a second weighting scheme is introduced in order to prevent over-confidence at predicting non-domain residues exclusively at tail ends of sequences due to the proportion of non-domain residue types being inherently imbalanced towards tail regions of sequences within the pool of all non-domain residues. This customization specifically broke down the residue-based loss functions into three categories: (1) negative residue weights; referring to all true within domain residues, (2) positive terminal weights; referring to all true non-domain residues belonging to a stretch of non-domain residues adjacent to either termini of the sequence, and (3) internal positive weights; referring to all true non-domain stretches that were not adjacent to either sequence termini, and can therefore be referred to as domain boundaries. A residue-type weight γ_i was additionally assigned as follows:

$$\gamma_i = \begin{cases} \gamma_{\text{domain}}, & \text{if } y_i = 0, \\ \gamma_{\text{terminal}}, & \text{if } y_i = 1 \text{ and residue } i \text{ belongs to a terminal positive region,} \\ \gamma_{\text{internal}}, & \text{if } y_i = 1 \text{ and residue } i \text{ belongs to an internal positive region.} \end{cases}$$

Subsequently, each residue's individual loss was derived by multiplying the residue type weight, the class balancing weight, and the base loss of binary cross entropy as follows:

$$\mathcal{L}_i = \gamma_i \alpha_{y_i} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)].$$

Each of these were aggregated through all residues per batch, in order to compute the loss through each epoch.

3.5 Feature Selection

With a total of 2,389 features, the computational cost of the model became high, particularly when running the evolutionary algorithm for feature selection. Redundancy

with the full feature set size was also suspected, which inspired the described feature reduction approach before final model training.

A two phase feature selection framework is proposed to improve both computational efficiency and feature subset quality. Since evolutionary algorithms become computationally expensive in very high-dimensional search spaces [42], the first phase removes irrelevant and redundant features using correlation-based feature selection, thereby reducing the search space. This preprocessing step has been shown to improve the efficiency and effectiveness of subsequent evolutionary optimization by eliminating noisy and highly correlated features. In the second phase, a multi-objective evolutionary algorithm is employed to identify an optimal subset of features from the reduced search space.

3.5.1 Phase 1: Correlation-based Feature Pruning

The first phase of feature selection was performed by testing out model performance on a variety of feature sets that included different proportions of ranked features based on the correlation each feature possessed with the target variable.

The correlation was computed with the bi-point serial correlation in order to capture the numerical and binary features correlation with the binary target variable. Each correlation was then converted into absolute values in order to rank the features based on how strong they correlate with the target. After ranking, multiple subsets of feature set sizes were produced based on the ranking to enable inclusion. Smaller feature sets represented a subset of features most correlated to the target, whereas larger feature sets represented lenient inclusion of more sparsely correlated features.

Finally, the LSTM model was trained on each distinct pruned feature set in order to test out model performance and identify a promising candidate of features that would reduce the complexity and improve the efficiency of the subsequent phase of feature selection.

3.5.2 Phase 2: Multi-Objective Evolutionary Feature Selection

Following correlation-based feature pruning, evolutionary computation for multi-objective optimization was used as the second phase of feature selection to find a stronger feature set amongst the retained candidate features within the pruned feature pool. In this stage, each individual represented a candidate subset of features selected from the pruned feature space. To mitigate premature convergence and preserve exploration throughout the feature selection, the worst performing 10 percent of individuals (ranked by macro F1-score) were replaced every ten generations with newly initialized individuals, injecting diversity and maintaining variation for the process of crossover. The feature subsets were evaluated according to the performance of a bidirectional LSTM trained using only those selected features as input. Because the goal of the model was not only to maximize overall classification performance, but also to improve detection of the minority non-domain label, feature subsets were assessed using multiple validation metrics, including macro F1-score, non-domain precision, and non-domain recall. This allowed the feature selection process to preserve candidate

solutions that represented different tradeoffs between distinct residue level classification metrics. Fig. 3 represents the methodological breakdown of the evolutionary algorithm’s process.

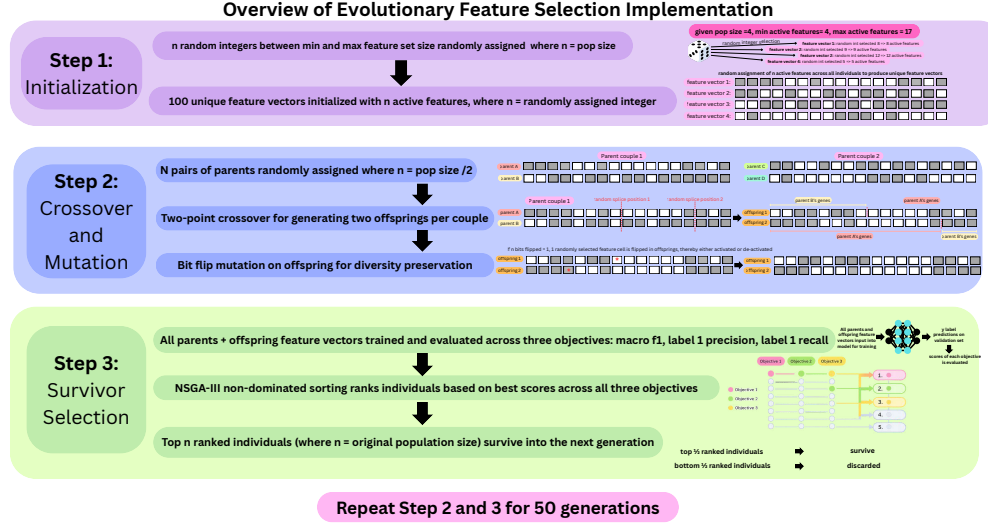


Fig. 3: Evolutionary Feature Selection Workflow. The evolutionary feature selection is performed in Seq2DomML through three main stages. In the initialization stage, a population of unique binary feature vectors is created. In the crossover and mutation stage, candidate vectors are paired to generate offsprings. In the survivor selection stage, all parent and offspring feature subsets are evaluated using LSTM model.

The primary practical constraint faced with the NSGA-III usage revolved around the computational cost of running it with an LSTM model. Therefore, to reduce the runtime, a few aspects of the optimization stage were restricted including limiting the maximum number of epochs during LSTM training, setting an early stopping rule on the validation macro F1-score, and utilizing only 15 percent of the training and validation sets. During this stage, the purpose was not to obtain the final performance of each model, but to efficiently rank feature subsets and identify promising regions of the feature selection space, hence why the dimensionality restrictions executed were useful. The evolutionary search therefore functioned as a screening step, narrowing the candidate feature subsets before more computationally intensive retraining was performed.

The output of this phase was therefore not a single optimized feature set immediately produced by the reduced feature selection optimization, but a final population of candidate feature subsets was retrained on the full size of the training set, as well as evaluated on the full validation set. The final population from the optimization phase

was then retrained on the full sized train set and evaluated on the full sized validation set. The final selected set of features was chosen after this evaluation stage based on not only the scores of each objective, but also by evaluating the correctness of the implied number of domains predicted across each sequence using the mean absolute error.

4 Experiments and Results

This section outlines the results from each of the above described experiments performed. Each result is interpreted, to demonstrate the decisions made about parameter selection, model tuning, and model selection from one step to the next.

4.1 Evaluation Metrics

Model performance was evaluated using both residue level classification metrics and a domain architecture level metric. At the residue level, performance was measured using domain residue F1-score, non-domain residue precision, non-domain residue recall, non-domain residue F1-score, and macro F1-score. Domain residue F1-score measures how accurately residues within annotated domains were recovered, while non-domain residue precision and recall evaluate the model’s ability to identify boundary, linker, and terminal flanking residues without over assigning these regions. Macro F1-score was included to summarize performance across both residue classes while reducing the dominance of the majority domain residue class. Since the intended application is domain partitioning for construct design, domain count mean absolute error (MAE) was also used to evaluate how closely each predicted residue label pattern resulted in the true number of implied domain units in each sequence. This metric is calculated by counting the number of domains predicted against the true number of domains determined by ECOD, thereby providing an important method of evaluating the overall domain assignment architecture implied by the model predictions.

4.2 LSTM Hyperparameter Setting

The first set of experiments evaluated how the LSTM model responded to different loss-weighting and prediction-threshold settings before feature selection was performed. These trials were designed to identify a configuration that could preserve strong classification of domain residues while improving detection of the minority non-domain class, which includes boundary, linker, and terminal flanking residues. Specifically, the experiments varied the effective class weight beta term used, the residue type weights assigned to domain residues, terminal non-domain residues, and internal non-domain residues, as well as the final prediction threshold used to convert residue probabilities into binary labels. All results in this subsection are reported on the validation set. For the residue type weights and beta terms, a wide combination of configurations were screened in order to find a parameter set that met a high standard for this model.

Overall, the beta term had a clear effect on the non-domain recall and precision tradeoff across the hyperparameter trials. Utilizing a more aggressive beta term of 0.999 demonstrated to improve the recall of non-domain residues, although it was at

the cost of significantly decreasing the precision of such non-domain residues. Specifically, the maximum non-domain recall scoring trial achieved a non-domain recall of 0.964, but yielded a non-domain precision of only 0.174. In contrast, the use of the beta term 0.99999 demonstrated to yield the strongest macro F1-score values and substantially stronger non-domain precision. The maximum non-domain precision scoring trial reached a non-domain precision of 0.953, but yielded a non-domain recall of only 0.365. These results demonstrate that the highest recall configurations were associated with severe precision loss, while the highest precision configurations were associated with reduced non-domain recall. The balanced trials in Table 3 show that both beta values could produce usable precision-recall compromises, although the more aggressive beta term generally appeared in models where recall was maintained with a moderate reduction in precision.

The residue type weight configurations also affected whether the model behaved as a recall oriented, precision oriented, or more balanced predictor. Overall, it was found that down weighting the positive terminal weights, whilst up weighting the internal positive weights yielded the best non-domain recall, particularly for the domain boundaries. The highest recall models used low terminal positive weights of 0.10 and higher internal positive weights of either 0.70 or 1.00. However, higher negative residue weights, such as $NRW = 1.00$ or 0.75 , combined with higher terminal and internal positive weights, shifted the model toward much stronger non-domain precision but weaker non-domain recall. The balanced results in Table 3 suggest that no single residue type weight configuration was universally optimal, yet, downweighting the terminal positive weight seemed to be the best contributor to maintaining decent non-domain recall. This observation aligns with the reality of the imbalance between non-domain residue’s types where more residues in this group is located at terminal regions which, for an LSTM makes the prediction task hyper fixated on position based assignments of such non-domain regions at sequence ends as opposed to considering the physicochemical contexts of neighboring residues within domain boundary regions of non-domain residues.

The prediction threshold further controlled the final precision recall balance after training. It was also found that allocating a more conservative prediction threshold to the model during training, whilst utilizing a lower prediction threshold for the final prediction yielded stronger recall of domain boundaries whilst preserving a moderately strong non-domain precision. The strongest non-domain recall values occurred at lower prediction thresholds of 0.45 to 0.55, where recall ranged from 0.911 to 0.964 but precision remained low, ranging from 0.174 to 0.192. In contrast, the strongest non-domain precision values occurred at higher prediction thresholds of 0.675 to 0.700, where precision exceeded 0.944 but recall decreased to approximately 0.365 to 0.382. The balanced results in Table 3 fall between these extremes, with prediction thresholds ranging from 0.45 to 0.625, indicating that moderate thresholds were most effective when attempting to preserve non-domain recall while avoiding the severe precision loss observed in the highest recall trials. These results lead to selecting a beta value of 0.99999, a negative residue weight of 0.5, a terminal positive weight of 0.25, and an internal positive weight of 1.0 for the subsequent correlation based feature pruning trials. The complete selected LSTM training configuration is presented in Table 4.

Table 3: LSTM Hyperparameter Trials with Balanced Non-domain Precision and Recall. Five validation trials selected to represent balanced performance for the non-domain class’s precision and recall. Trials were included when non-domain precision exceeded 0.650 and non-domain recall exceeded 0.450. NRW denotes the residue weight applied to within-domain residues, TPW denotes the weight applied to terminal non-domain residues, and IPW denotes the weight applied to internal non-domain residues associated with boundaries and linker regions. The column abbreviations depict specific metrics as follows: ND = non-domain; WD = within-domain; Pred. Thresh. = prediction threshold; Prec. = precision.

NRW	TPW	IPW	Beta	Pred. Thresh.	Macro F1	ND Prec.	ND Recall	WD F1	ND F1
1.000	0.500	1.500	0.99999	0.450	0.756	0.693	0.467	0.932	0.580
1.000	1.000	1.000	0.999999	0.500	0.764	0.721	0.466	0.935	0.594
0.500	0.250	1.000	0.99999	0.550	0.749	0.675	0.459	0.930	0.567
1.000	0.800	1.200	0.999999	0.550	0.767	0.738	0.459	0.936	0.598
0.750	0.750	1.500	0.999999	0.625	0.760	0.715	0.458	0.934	0.586

Table 4: Final LSTM Training Configuration. The table summarizes the final hyperparameter and loss-weighting configuration used to train the bidirectional LSTM model after validation-based screening of learning settings, class-weighting parameters, residue-type weights, and prediction thresholds. NRW denotes the weight applied to within-domain residues, TPW denotes the weight applied to terminal non-domain residues, and IPW denotes the weight applied to internal non-domain residues associated with boundary and linker regions.

Parameter	Value
Learning rate	1×10^{-4}
Spatial Dropout 1D	0.1
LSTM dropout	0.1
Loss function	Binary cross entropy
Optimizer	Adam
Bidirectional	True
Early stopping	No Validation macro F1-score improvement for 10 epochs
Batch size	50
LSTM units	100
CB beta	0.99999
Negative residue weight (NRW)	0.5
Terminal positive weight (TPW)	0.25
Internal positive weight (IPW)	1.0
Prediction threshold	0.55

4.3 Correlation-based Feature Pruning Experiments

The next investigation of the model’s performance was tested under different proportions of inclusion based on a threshold of each feature’s correlation with the target variable. To quantify the relationship between each feature and the binary target variable, the point-biserial correlation coefficient was calculated for every feature. Variations of feature sets were therefore deposited where each variation accounted for a distinct proportion of the n most correlated features with the target variable.

As shown in Table 5, with the primary goal of improving non-domain recall whilst maintaining strong non-domain precision, it was observed that the feature set sizes closer to the maximum size yielded the best macro F1-score, whereas the feature set sizes closer to the minimum size yielded the strongest non-domain recall. This led us to select the feature set size of 25 percent of the full feature set in order to maintain a balance between the two extremes whilst slightly biasing towards smaller feature set sizes that promoted stronger non-domain recall, the inevitably harder to improve metric. This feature set size of 25 percent was then selected to make up the pool of features for the subsequent evolutionary feature selection.

Table 5: Validation Performance Across Correlation-based Feature Inclusion Thresholds. The table compares the feature subsets produced during the first phase of feature reduction by retaining different proportions of the original 2,389 residue features. The inclusion threshold indicates the proportion of the complete feature space retained, while the second column reports the corresponding number of features used as model inputs. A threshold of 1.0 represents the complete feature set, while smaller thresholds represent progressively reduced feature spaces selected according to their correlation based ranking. The column abbreviations depict specific metrics as follows: WD = within-domain; ND = non-domain; Prec. = precision; Re. = recall

Threshold	# Features	Macro F1	WD F1	ND F1	ND Prec.	ND Rec.
0.025	59	0.758	0.935	0.582	0.739	0.425
0.050	119	0.766	0.937	0.596	0.769	0.423
0.100	238	0.776	0.939	0.612	0.805	0.419
0.150	358	0.786	0.942	0.630	0.849	0.411
0.200	477	0.780	0.941	0.620	0.825	0.415
0.300	716	0.783	0.941	0.625	0.833	0.417
0.400	955	0.787	0.942	0.632	0.850	0.415
0.500	1194	0.778	0.940	0.615	0.806	0.424
0.600	1433	0.787	0.942	0.632	0.849	0.416
0.800	1911	0.785	0.941	0.628	0.836	0.419
1.000	2389	0.792	0.943	0.641	0.875	0.407

4.4 Evolutionary Feature Selection Experiments

The evolutionary feature space optimization was initialized with 100 individuals each containing a random number of selected features between 40 and 597 in order to possess size and feature diversity in the starting population. Offspring were produced using two point crossover followed by bit flip mutation that decayed in the number of flips every ten generations to promote more diversity in earlier generations. The starting mutation rate was set to 5 bit flips per individual, then by generation 40 - 50 proceeds with just 1 bit flipped per individual. This allowed the search to explore a broader range of feature combinations in earlier generations whilst gradually shifting toward smaller refinements of stronger candidate feature subsets in later generations. Table 6 illustrates the full set of operators used to perform this step.

Table 6: NSGA-III Operator Settings for Evolutionary Feature Selection. Summary of the evolutionary algorithm settings used during the second phase of feature selection in Seq2DomML. Candidate feature subsets contained between 40 and 597 active features and were evolved within a population of 100 individuals. NSGA-III used 91 reference points to preserve diversity across the three optimization objectives.

GA Operator	Value / Rule
Minimum feature set size	40
Maximum feature set size	597
Population size	100
Reference points	91
Crossover	Two point
Mutation rate	5 bit flips, decaying by 1 every 10 generations
Replacement rule	Replace the bottom 10% by macro F1 score every 10 generations with newly initialized random individuals

The optimization process in feature selection showed gradual improvement across generations, particularly for macro F1-score and non-domain precision. The best macro F1-score increased early in the search and then stabilized for much of the run, suggesting that the feature selection identified relatively strong feature subsets within the first several generations. Non-domain precision showed a larger stepwise improvement, rising substantially during the early generations and improving again near the final generations, indicating that the evolutionary search was able to identify feature combinations that reduced false positive non-domain predictions. Non-domain recall remained comparatively stable and lower than the non-domain precision, reflecting the difficulty of detecting the minority non-domain class which have faced throughout this project. The primary purpose of this stage was therefore to guide the search toward promising feature subsets, which could then be used for retraining on the full dataset at the final population of all 100 surviving individuals.

After optimization and training the final population across the full size training dataset, each feature set was evaluated on the validation set across six different prediction thresholds [0.35, 0.4, 0.45, 0.5, 0.55, 0.6] in order to further expand the possible optimization space. This process produced 600 candidate models for evaluation. Given the notable trade-offs between the optimization objectives, the final model was selected based on its overall validation performance before being evaluated on the held-out test set.

To identify candidate trials for final selection, the 600 post-feature selection trials were first filtered by applying minimum performance requirements to all three optimization objectives and the domain count MAE. Candidate trials were required to satisfy non-domain precision greater than 0.6, non-domain recall greater than 0.45, a macro F1-score greater than 0.74, and a domain count MAE less than 0.94. These thresholds were selected to exclude trials that achieved strong performance in only one metric while performing poorly in another. In application terms, the selected candidates needed to retain sufficient sensitivity to boundary, linker, and terminal residues, avoid excessive over assignment of non-domain regions, preserve strong overall residue classification, and maintain a reasonable estimate of the implied number of domain units. This filtering step yielded 16 candidate trials, each coming from distinct individuals evaluated at unique prediction thresholds, as shown in Table 7. Across all trials, visible tradeoffs between residue level classification metrics and the range of error in the implied domain counts, which made the decision of selecting the final best individual feature set less straightforward.

As shown in the Figure 4, non-domain precision and non-domain recall displayed the expected inverse relationship, where trials with higher precision generally showed lower recall. This tradeoff was important to recognize for final selection because excessive emphasis on precision, would contribute to a higher overall macro F1-score, which would in turn, reduce the detection of true boundary and linker regions. On the other hand, excessive emphasis on recall could increase overprediction of non-domain residues. A similar tradeoff was observed between non-domain recall and macro F1-score, where higher recall generally corresponded to lower macro F1-score, reflecting that non-domain precision tends to make a stronger contribution to the macro F1-score, than the non-domain recall. When considering the effects of non-domain precision, non-domain recall, and the macro F1-score against the domain count MAE, all of the comparison plots demonstrate similar patterns where the strongest objective scores do not correlate with the strongest domain count MAE.

The result additionally demonstrate that the macro F1-score increased strongly with non-domain precision, but the best macro F1-score values were far from corresponding to the lowest domain count MAE. In contrast, domain count MAE displayed a curved relationship with non-domain precision, non-domain recall, and the macro F1-score. The lowest domain count errors occurred in an intermediate performance region, rather than at the extremes of precision, recall, or macro F1-score. This implies that in order to achieve stronger predictive accuracy for the implied domain counts within a sequence, a balance between the objectives is necessary. Among the 16 filtered candidate trials seen in Table 7, the final selected feature set was composed of 531 features, and evaluated at a prediction threshold of 0.4. This trial was selected

Table 7: The 16 trials of post-feature selection individuals evaluated at different prediction thresholds meeting the criteria of macro F1-score greater than or equal to 0.74, non-domain precision greater than or equal to 0.60, non-domain recall greater than or equal to 0.45, and domain count MAE less than 0.94. The column abbreviations depict specific metrics as follows: WD = within-domain; ND = non-domain; Feat. = features; Pred. Thresh. = prediction threshold; Prec. = precision; DC MAE = domain count mean absolute error.

# Feat.	Pred. Thresh.	WD F1	ND F1	ND Prec.	ND Recall	Macro F1	DC MAE
434	0.450	0.934	0.584	0.717	0.451	0.759	0.923
441	0.450	0.932	0.578	0.692	0.463	0.755	0.914
492	0.450	0.932	0.576	0.685	0.468	0.754	0.887
527	0.450	0.932	0.575	0.686	0.465	0.754	0.907
477	0.450	0.932	0.575	0.685	0.464	0.753	0.902
490	0.400	0.930	0.572	0.671	0.474	0.751	0.916
474	0.450	0.930	0.571	0.666	0.476	0.751	0.919
516	0.400	0.931	0.570	0.680	0.460	0.750	0.923
511	0.450	0.931	0.569	0.676	0.462	0.750	0.926
515	0.400	0.930	0.569	0.668	0.471	0.750	0.883
474	0.400	0.930	0.567	0.667	0.467	0.748	0.928
531	0.400	0.929	0.566	0.660	0.471	0.747	0.893
424	0.450	0.928	0.565	0.645	0.485	0.746	0.923
464	0.400	0.929	0.563	0.654	0.471	0.746	0.929
516	0.400	0.926	0.558	0.631	0.486	0.742	0.907
454	0.400	0.926	0.555	0.634	0.476	0.741	0.898

because on the validation set, it achieved one of the lowest domain count MAEs among the filtered candidates whilst possessing one of the highest non-domain recalls, which has notably been the proposed model’s most difficult to improve aspect. This feature set and prediction threshold were therefore selected for final test set evaluation and comparison across the other tools.

4.5 Comparison with State-of-the-Art Domain Annotation Tools

This experiment involved comparing the evaluation performed on the test set with the selected feature set and hyperparameter knobs described above against the performance of three state of the art tools; Pfam, Interpro, and NCBI CDD, by using these tools to make the predictions on the same test set. Contrary to Seq2DomML, these tools largely rely on homology based approaches for the domain prediction task. These comparisons thereby establish the strengths and weaknesses of each tool’s applicability in distinct biochemical contexts. The predictions from each tool are normalized into a consistent format of binary labels across each sequence, allowing for performance comparisons to remain fair whilst allowing interoperability for the strengths and weaknesses specific to applications.

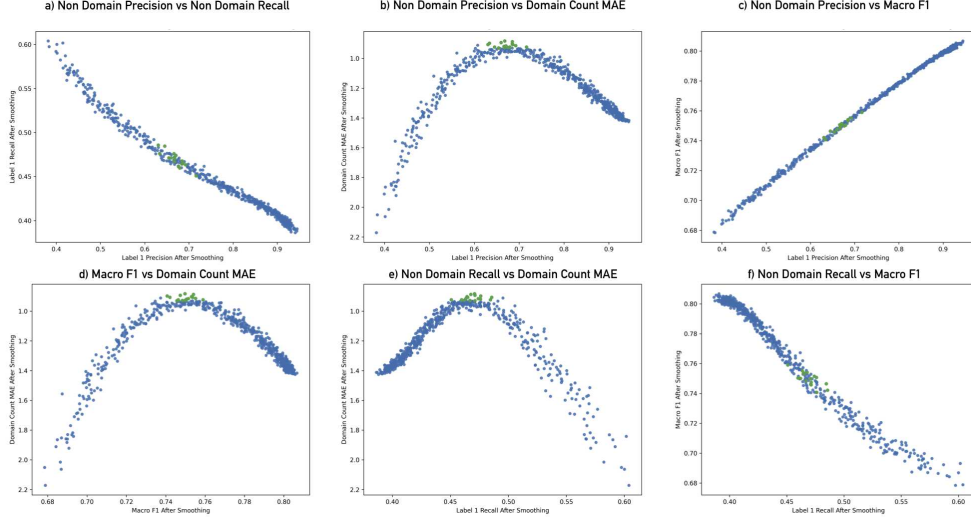


Fig. 4: Pairwise Performance Trade-offs Across Final Feature Selection Trials. The six scatterplots compare the validation performance of the 600 trials. Each blue point represents one feature subset and prediction threshold combination after prediction smoothing, while the green points identify the 16 candidate trials retained after applying the final performance filters. Each subfigure represents the trade-off between two evaluation metrics.

The final selected model was evaluated on the test set and compared against three existing sequence-based domain annotation tools: InterPro, Pfam, and NCBI CDD. For each tool, predicted domain annotations were converted into the same residue level binary label format used for the model, where the within domain class represented residues within annotated domains and non-domain represented residues outside annotated domains, including boundaries, linkers, and terminal flanking regions. The same non-domain stretching rule used in the initial target encoding process was implemented for the other tools so that domain boundaries were stretched out to 9 residues in length thereby enabling a fair evaluation in this comparison.

As shown in Table 8, Seq2DomML achieved the highest macro F1-score score of 0.744 among the evaluated methods. The model also achieved the strongest non-domain precision, with a precision of 0.654, indicating that its non-domain predictions were less prone to false positive non-domain assignments than the other tools. NCBI CDD achieved the second highest non-domain precision at 0.596, whereas Pfam and InterPro produced substantially lower non-domain precision values of 0.387 and 0.344, respectively.

The main tradeoff for Seq2DomML was observed in non-domain recall. InterPro and Pfam achieved higher non-domain recall values, with recalls of 0.708 and 0.717, respectively, while Seq2DomML model achieved a lower non-domain recall of 0.466.

Table 8: Test Set Performance Comparison Across Domain Annotation Tools. The table compares the performance of InterPro, Pfam, NCBI CDD, and Seq2DomML on the held out test set using residue classification metrics and the domain count mean absolute error. Higher values indicate stronger performance for all classification metrics, whereas lower values indicate stronger performance for domain count MAE. The column abbreviations depict specific metrics as follows: WD = within-domain; ND = non-domain; Prec. = precision; DC MAE = domain count mean absolute error.

Tool	WD F1	ND Prec.	ND Recall	ND F1	Macro F1	DC MAE
InterPro	0.826	0.345	0.709	0.464	0.645	1.065
Pfam	0.853	0.387	0.717	0.503	0.678	0.860
NCBI CDD	0.921	0.597	0.482	0.533	0.727	1.390
Seq2DomML	0.928	0.655	0.467	0.561	0.745	0.916

This indicates that InterPro and Pfam detected more true non-domain residues, but did so with considerably lower precision, indicating that many true domains were simply not detected, and predicted as non-domains. In contrast, Seq2DomML’s model produced fewer non-domain predictions overall, but those predictions were much more confident. Despite the inflated recall metrics seen in the other tools, the true recollection of non-domain residues amongst these tools should be interpreted with caution. Any undetected domains across these sequences, were simply classified as non-domain, meaning the recall may not truly be as strong as the classification metric suggests in these tools.

The residue level heat maps in Fig. 5 provide a visual comparison of these prediction patterns across a sample set of 50 sequences. InterPro and Pfam show broader and more fragmented non-domain predictions across many sequences, consistent with their higher recall but lower precision. NCBI CDD produces more conservative domain region predictions, but visibly fails to identify a substantial proportion of non-domain regions relative to the ground-truth annotations. Seq2DomML shows a more selective non-domain prediction pattern, with fewer scattered non-domain predictions than InterPro and Pfam, while still capturing a decent amount of boundary associated regions. The longer stretches of non-domain fragments observable in interprot and Pfam, additionally reflect the homology-based focus of identifying domains, which leads to unrecognized domains with few known homologs being misrepresented as non-domain regions.

In addition to residue level classification metrics, each method was additionally evaluated according to its implied number of domains per sequence, where a domain is defined as a stretch of predicted within domain residues greater than ten residues in length. This analysis was included because the intended use of the model is not only to identify individual residue labels, but also to support domain partitioning decisions for construct design. For each sequence, predicted label patterns were converted into

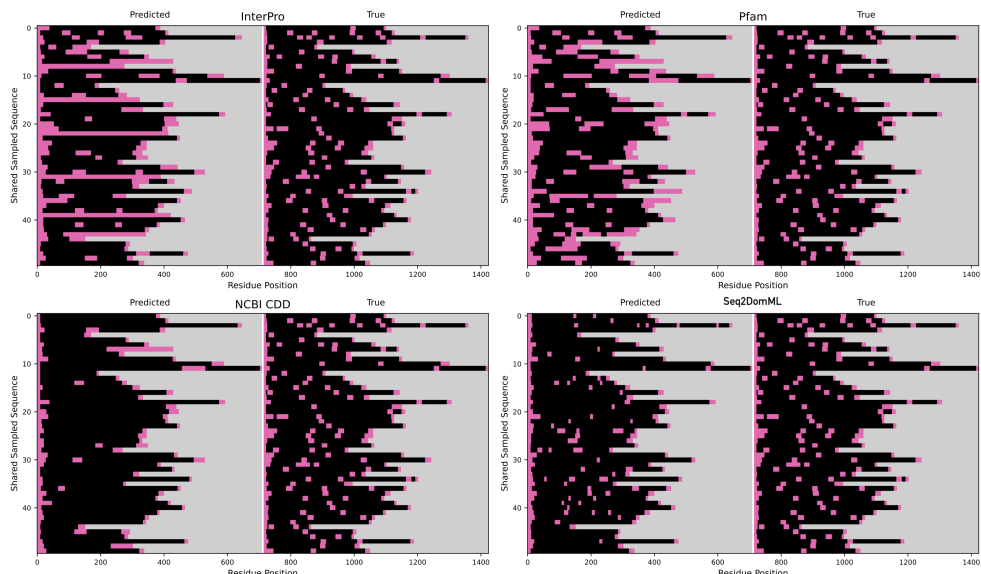


Fig. 5: Comparison Heatmaps of Residue Level Predictions Across a Sample Set of Test Sequences for Each Domain Annotation Method. The heatmaps compare the predicted residue labels produced by InterPro, Pfam, NCBI CDD, and Seq2DomML with the corresponding true ECOD derived labels for the same 50 randomly selected test sequences. Each row represents one protein sequence, and the horizontal axis represents residue position. Within each subfigure, the predicted labels are shown on the left and the true labels are shown on the right, allowing the spatial agreement between predicted and reference domain architectures to be examined across methods. Black represents within domain residues, whilst pink represents non-domain residues including domain boundaries, linkers, and terminal flanking regions. Grey regions indicate positions beyond the end of shorter sequences after alignment to a shared plotting width. The repeated true label heatmap in each subfigure provides the same reference for visually comparing how closely each method reproduces the location and extent of within domain and non-domain regions.

an implied domain count, and the mean absolute error between predicted and true domain counts was calculated.

As shown in Table 8, Pfam achieved the strongest domain count MAE with it’s score being the lowest at 0.860, which was followed by Seq2DomML’s model at 0.915 InterPro at 1.065, and NCBI CDD at 1.390. This indicates that Pfam produced the closest overall domain count estimates, while Seq2DomML’s model achieved the second best domain count MAE. Although NCBI CDD performed relatively well in residue level classification, its domain count MAE was the highest, suggesting that residue level F1-score metrics alone do not fully capture performance metrics to the best extent.

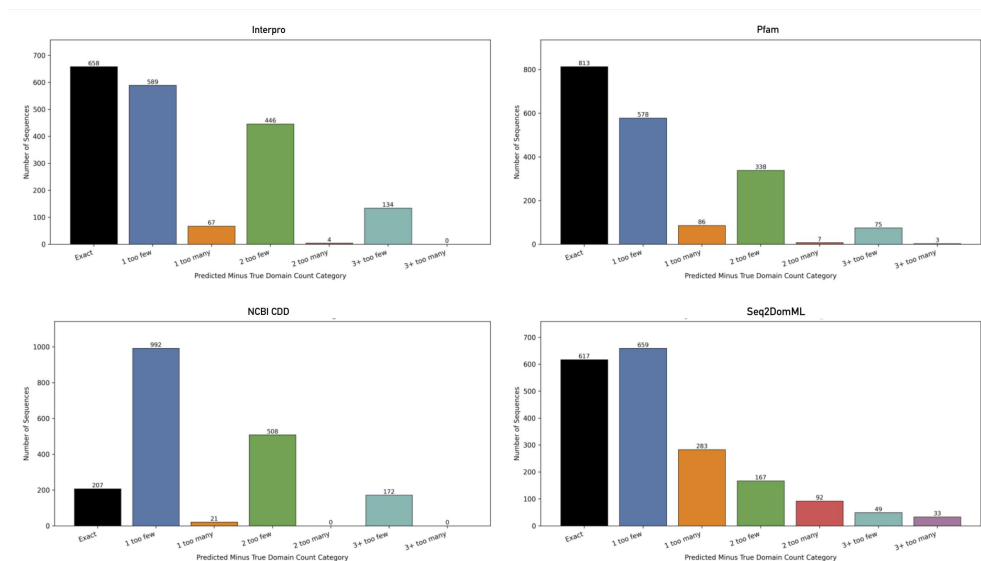


Fig. 6: Comparison of Domain Error Category Counts Across Annotation Tools. The plots summarize the difference between the number of domain units predicted by InterPro, Pfam, NCBI CDD, and Seq2DomML and the corresponding true number derived from ECOD annotations. The horizontal axis groups sequences according to the predicted domain count relative to the true count, while the vertical axis shows the number of sequences assigned to each category. Exact indicates that the predicted and true domain counts were equal. The categories 1 too few, 2 too few, and 3 or more too few indicate increasing underprediction of the true number of domains, whereas 1 too many, 2 too many, and 3 or more too many indicate increasing overprediction. The number displayed above each bar gives the total number of sequences within that category. Together, the four panels show the distribution and direction of domain count errors produced by each annotation method.

The domain count error category plots seen in Fig. 6 provide a more detailed view of how each method arrived at its overall domain count MAE. Pfam produced the largest number of exact domain count predictions, with 813 sequences predicted exactly, followed by InterPro with 658, Seq2DomML with 617, and NCBI CDD with 207. However, the broader error distributions reveal an important distinction between exact-count matching and overall prediction behavior. NCBI CDD was dominated by underprediction, with 992 sequences predicted as one domain too few and 508 predicted as two domains too few, indicating that its residue-level performance did not consistently preserve the correct number of domain units across full sequences. InterPro and Pfam also tended to underpredict domain number more often than they overpredicted it, although this tendency was less pronounced than for NCBI CDD.

Seq2DomML showed a more balanced distribution of underprediction and overprediction errors, with 659 sequences predicted as one domain too few and 283 predicted as one domain too many. While Pfam performed best for exact domain count recovery, Seq2DomML achieved the second lowest domain count MAE whilst avoiding the severe underprediction pattern observed for NCBI CDD, demonstrating a key strength of the proposed framework. Taken together with its strongest macro F1-score, non-domain precision, and non-domain F1-score, this indicates that the final model offers a comparatively balanced prediction profile: it remains competitive for domain count recovery while providing stronger residue level discrimination of non-domain regions.

Looking beyond residue level metrics, a visual comparator of each these tools predictions was produced by mapping the assigned labels onto the existing solved structures for each. The visualizations represented in Figures 7-9 were chosen from a batch of 100 randomly selected comparison structures in order to summarize the strengths and applicability of Seq2DomML. In these rendered images, residues assigned to non-domain are shown in white, representing predicted non-domain regions, including linkers, terminal flanking regions, and boundary associated residues, and structured domain regions are shown in color, where the red accounts for the first domain, orange for the second, and yellow for the third. It is however important to note that for Pfam, NCBI CDD, and Interpro, excessively long non-domain regions do not completely indicate a non-domain region being predicted, yet specifically allude to an absence of any domain detection within that sequence’s region. This pattern can be observed by the large portions of some predicted labels colored in white across interpro and Pfam most notably.

It is also important to note that because non-domain regions were deliberately stretched to nine residues during target label encoding, which can produce white boundary regions that extend into structured secondary elements in both the ECOD derived true labels and the predicted labels. Therefore, the presence of white residues within otherwise structured regions should not automatically imply a non-domain region, yet instead should imply a region where a domain boundary is predicted to be present within.

In the PDB structure 1V8C Chain D as seen in Fig. 7, the ECOD derived labels show two major structured domains separated by a short non-domain boundary region. Pfam and InterPro visually recover a similar two-domain organization, with the left beta rich domain and the right helical domain separated by a white linker appearing interval. NCBI CDD also broadly captures this organization, although its segmentation appears more conservative and less sharply separated in the boundary region. While Seq2DomML produces a visually comparable two domain partition, it additionally assigns a small internal region within the left domain as non-domain. This extra white region may represent a false positive boundary prediction, but it is not structurally absurd as it occurs near an exposed looping edge region rather than being randomly distributed across the buried domain core. This suggests that some apparent false positive boundary predictions may instead reflect structurally plausible local flexibility or subdomain like partitioning not explicitly defined from the ECOD annotation, although such cases should be interpreted as hypothesis generating rather than as evidence that the ECOD assignment is incorrect.

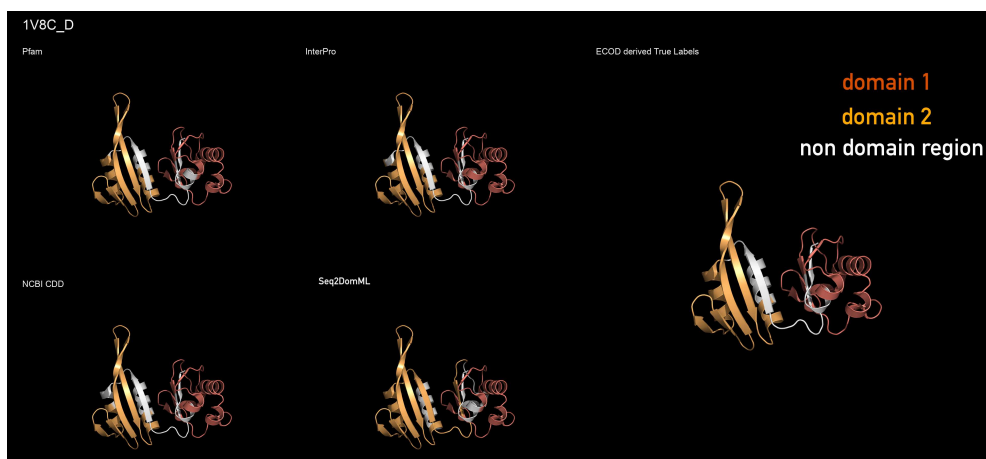


Fig. 7: Structural Comparison of Predicted Domain Partitions for PDB 1V8C Chain D. Residue labels predicted by Pfam, InterPro, NCBI CDD, and Seq2DomML are mapped onto the experimentally determined protein structure and compared with the ECOD derived true labels. White regions represent residues assigned to the non-domain class, including terminal regions and boundaries separating domain units. Coloured regions represent residues assigned to the within domain class. Distinct colours are used to separate consecutive predicted domain units. The ECOD structure on the right provides the reference domain partition against which the four prediction methods are visually compared.

In the PDB structure 2YQB Chain A as seen in Fig. 8, the ECOD derived structural annotations contain three visually distinct domain regions. Pfam underdetects part of this structure, leaving a large central region annotated white as a non-domain, which suggests that no domain was recognized for a substantial structured portion of the protein. InterPro performs better, recovering a multi domain pattern that more closely resembles the ECOD derived labels, although some boundary placement still differs. NCBI CDD identifies the large right hand domain but misses a crucial domain boundary therefore merging two distinct domains into a single domain. In contrast, Seq2DomML recovers a three domain organization that, alike the Interpro predictions, are visually similar to the ECOD segmentation. This can be seen through the left helical domain, the central beta rich domain, and the larger right hand beta rich domain. Although some boundary regions are expanded into structured elements, this is expected under the label stretching strategy. Visually, the segmentations of domains from Seq2DomML’s model and Interpro appear to be the most reliable for this specific structure, highlighting the competitive nature that the proposed framework possesses.

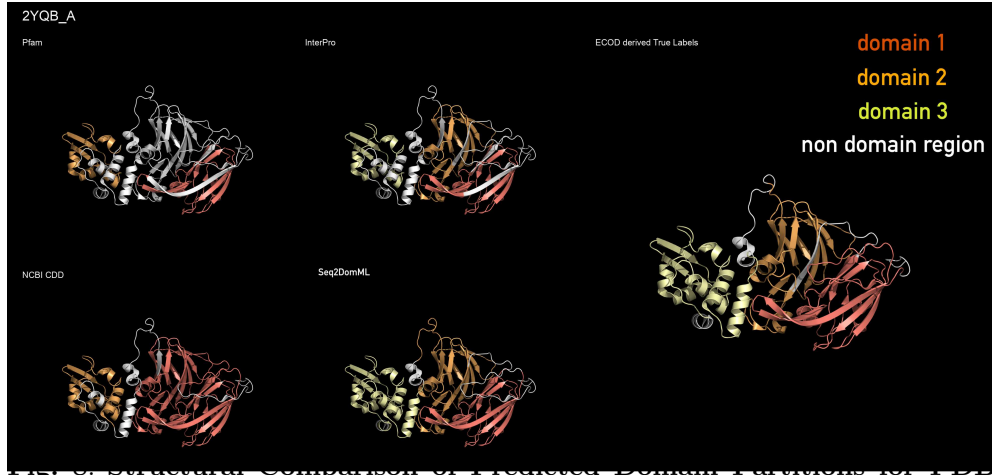


Fig. 8. Structural Comparison of Predicted Domain Partitions for PDB 2YQB Chain A. Residue labels predicted by Pfam, InterPro, NCBI CDD, and Seq2DomML are mapped onto the experimentally determined protein structure and compared with the ECOD derived true labels. White regions represent residues assigned to the non-domain class, including terminal regions and boundaries separating domain units. Coloured regions represent residues assigned to the within domain class. Consecutive domain units are shown using distinct colours. The ECOD structure on the right provides the reference partition for evaluating how closely each method reproduces the three domain architecture of 2YQB Chain A.

On the other hand, the PDB structure of 5B7W Chain B as seen in Fig. 9 highlights one of the clearer cases to vouch for the Seq2DomML, where existing tools under detect domain presences completely within a structure. The ECOD derived labels show only two major colored regions separated by a white boundary associated interval. Pfam and NCBI CDD largely assign the right hand region as a single domain while leaving the left hand folded region white, implying that no domain was detected across a structurally ordered portion of the protein. InterPro is even more conservative in this case leaving the entirety of the structure annotated as a non-domain, which implies a lack of domain recognition completely. Seq2DomML, however, recovers a two domain pattern that more closely resembles the ECOD derived labels, assigning the left and right folded regions as separate domain units. The white linker between them is also retained, which is important for construct design because this region would be an important region for splicing selection guidance between independently folded units.

This example therefore provides one of the strongest visual demonstrations of the proposed model’s practical value as a domain partitioning tool, as it identifies structured domain content in regions where the other tools were unable to do so. Notably, this structure is annotated in the PDB with an *unknown function* label, which supports the notion that under-characterized structural domains within protein databases will be much trickier for existing homology-based tools to recognize in input sequences. This emphasizes the potential utility of Seq2DomML for cases where domains are

uncharacterized, unrecognized, or poorly studied by existing homology-based tools, as demonstrated by this protein.

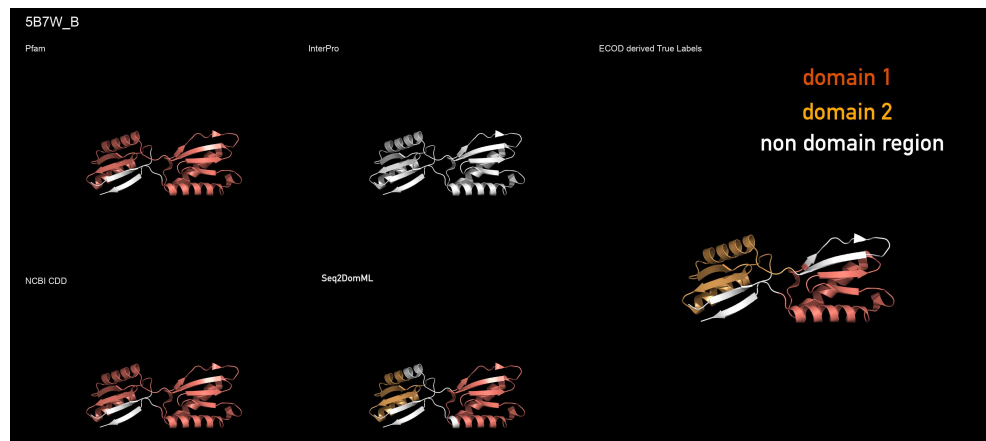


Fig. 9: Structural Comparison of Predicted Domain Partitions for PDB 5B7W Chain B. Residue labels predicted by Pfam, InterPro, NCBI CDD, and Seq2DomML are mapped onto the experimentally determined protein structure and compared with the ECOD derived true labels. White regions represent residues assigned to the non-domain class, including terminal regions and boundaries separating domain units, while coloured regions represent residues assigned to the within domain class. Consecutive domain units are displayed using distinct colours. The ECOD structure on the right provides the reference partition for assessing how closely each method reproduces the two domain architecture of 5B7W Chain B.

Overall, the structure mapped comparisons show that the quantitative metrics only partially capture each tool’s capabilities of the prediction task. Pfam, InterPro, and NCBI CDD often perform well when the queried sequence matches existing domain models, but their visual outputs can contain long white regions where no domain is detected, even when those residues clearly participate in folded secondary structure. Conversely, Seq2DomML sometimes introduces additional domain partitions or boundary associated regions absent in the true ECOD labels, but these predictions are frequently localized near regions that have debatable structural transitional characteristics such as exposed loops, or separable folded lobes rather than being randomly distributed. This distinction is important for the intended application of the model. For construct design, a sequence only predictor does not need to replace curated domain databases; rather, it can provide an additional residue level hypothesis layer for identifying candidate splice regions, especially in cases where a domain is largely uncharacterized, leading to a complete absence of domain recognition.

5 Conclusion

Domain boundary prediction across protein sequences remains a crucial step in construct design for structural solution, particularly when selecting splice sites that improve the isolation, expression, and crystallization of stable single-domain protein units. In this study, ECOD-derived structural domain annotations were used to train a sequence-based bidirectional LSTM model capable of predicting residue-level domain and non-domain regions without requiring structural coordinates as input, producing the proposed framework of Seq2DomML. This approach was designed to support the identification of boundary, linker, and terminal flanking regions that may guide domain partitioning during construct design.

Current state-of-the-art tools including Pfam, InterPro, and NCBI CDD use homology-based annotation to predict domain regions, and therefore show strengths when a domain has many well-studied homologs, yet commonly fall short when homologs are absent or poorly represented in their databases. Seq2DomML demonstrated competitive performance alongside these tools and highlighted advantageous prediction tendencies for poorly studied domain families. The model achieved the strongest macro F1-score, non-domain residue precision, and non-domain residue F1-score, suggesting that its predicted boundary and linker regions were more selective and less prone to excessive non-domain assignment associated with unrecognized structured domains in homology-based outputs.

Overall, these results suggest that sequence-based learning from ECOD annotations can provide a complementary prediction layer for protein domain partitioning and structural solution workflows, particularly when homology-based tools fail to recognize structured regions with limited annotation coverage. Future work should focus on improving boundary and linker recall, experimentally validating predicted splice sites in expression and crystallization pipelines, and testing whether Seq2DomML generalizes beyond bacterial X-ray structures to broader structural biology applications.

Declarations

Ethics approval and consent to participate. Not applicable.

Consent for publication. Not applicable.

Data Availability. All protein sequences used in this study came from the Protein Data Bank’s Public library at <https://www.rcsb.org/>. The exact list of the proteins used in training and evaluation are provided as Entry IDs with their specific protein chains in the supplementary information.

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

Funding. Funding was granted by the Natural Sciences and Engineering Research Council of Canada (NSERC).

Authors’ contributions. Alana Monks, contributed to the conceptualization of the study, methodological developments, software implementation, data collection, data

preprocessing, model development, formal analysis and visualizations, as well as the preparation of this manuscript.

Azam Asilian Bidgoli co-supervised this study, contributed to the to the conceptualization of the study and methodological developments, and the revision of the manuscript.

Author Mike Suits contributed to the conception of the study, co-supervision of the project, funding acquisition, and revised the manuscript. All authors reviewed and approved the final manuscript.

Acknowledgements. Not applicable.

Supplementary information. The supplementary materials are provided in the zip file archive named `supplementary_materials.zip`. See the `supplementary_files_readme.pdf` file for the outline of the supplementary materials contents.

References

- [1] Cooper, C.D.O., Marsden, B.D.: N- and c-terminal truncations to enhance protein solubility and crystallization: Predicting protein domain boundaries with bioinformatics tools. *Methods Mol. Biol.* **1586**, 11–31 (2017)
- [2] Edich, M., Gao, Y., Briggs, D.C., Thorn, A.: Unlocking the secrets of SARS-CoV-2 nsp3 by combining experiments with AlphaFold2 domain prediction. *Life Sci. Alliance* **9**(4), 202503247 (2026)
- [3] Malawski, G.A., Hillig, R.C., Montecarlo, F., Eberspaecher, U., Schmitz, A.A.P., Crusius, K., Huber, M., Egner, U., Donner, P., Müller-Tiemann, B.: Identifying protein construct variants with increased crystallization propensity—a case study. *Protein Sci.* **15**(12), 2718–2728 (2006)
- [4] Paysan-Lafosse, T., Andreeva, A., Blum, M., Chuguransky, S.R., Grego, T., Pinto, B.L., Salazar, G.A., Bileschi, M.L., Llinares-López, F., Meng-Papaxanthos, L., Colwell, L.J., Grishin, N.V., Schaeffer, R.D., Clementel, D., Tosatto, S.C.E., Sonnhammer, E., Wood, V., Bateman, A.: The pfam protein families database: embracing AI/ML. *Nucleic Acids Res.* **53**(D1), 523–534 (2025)
- [5] Blum, M., Andreeva, A., Florentino, L.C., Chuguransky, S.R., Grego, T., Hobbs, E., Pinto, B.L., Orr, A., Paysan-Lafosse, T., Ponamareva, I., Salazar, G.A., Bordin, N., Bork, P., Bridge, A., Colwell, L., Gough, J., Haft, D.H., Letunic, I., Llinares-López, F., Marchler-Bauer, A., Meng-Papaxanthos, L., Mi, H., Natale, D.A., Orengo, C.A., Pandurangan, A.P., Piovesan, D., Rivoire, C., Sigrist, C.J.A., Thanki, N., Thibaud-Nissen, F., Thomas, P.D., Tosatto, S.C.E., Wu, C.H., Bateman, A.: InterPro: the protein sequence classification resource in 2025. *Nucleic Acids Res.* **53**(D1), 444–456 (2025)

- [6] Hart, D.J., Waldo, G.S.: Library methods for structural biology of challenging proteins and their complexes. *Curr. Opin. Struct. Biol.* **23**(3), 403–408 (2013)
- [7] Cheng, H., Schaeffer, R.D., Liao, Y., Kinch, L.N., Pei, J., Shi, S., Kim, B.-H., Grishin, N.V.: ECOD: an evolutionary classification of protein domains. *PLoS Comput. Biol.* **10**(12), 1003926 (2014)
- [8] Burley, S.K., *et al.*: Updated resources for exploring experimentally-determined PDB structures and computed structure models at the RCSB protein data bank. *Nucleic Acids Research* **53**(D1), 564–574 (2025) <https://doi.org/10.1093/nar/gkae1091>
- [9] Kawashima, S., Pokarowski, P., Pokarowska, M., Kolinski, A., Katayama, T., Kanehisa, M.: AAindex: amino acid index database, progress report 2008. *Nucleic Acids Research* **36**(Database issue), 202–205 (2008) <https://doi.org/10.1093/nar/gkm998>
- [10] Jaenicke, R.: Stability and folding of domain proteins. *Prog. Biophys. Mol. Biol.* **71**(2), 155–241 (1999)
- [11] Bhaskara, R.M., Brevern, A.G., Srinivasan, N.: Understanding the role of domain-domain linkers in the spatial orientation of domains in multi-domain proteins. *J. Biomol. Struct. Dyn.* **31**(12), 1467–1480 (2013)
- [12] Holcomb, J., Spellmon, N., Zhang, Y., Doughan, M., Li, C., Yang, Z.: Protein crystallization: Eluding the bottleneck of x-ray crystallography. *AIMS Biophys.* **4**(4), 557–575 (2017)
- [13] Reddy Chichili, V.P., Kumar, V., Sivaraman, J.: Linkers in the structural biology of protein-protein interactions. *Protein Sci.* **22**(2), 153–167 (2013)
- [14] Papaleo, E., Saladino, G., Lambrugh, M., Lindorff-Larsen, K., Gervasio, F.L., Nussinov, R.: The role of protein loops and linkers in conformational dynamics and allostery. *Chem. Rev.* **116**(11), 6391–6423 (2016)
- [15] Ito, S., Nishino, T.: Rational engineering in protein crystallization: Integrating physicochemical principles, molecular scaffolds, and computational design. *Crystals (Basel)* **16**(1), 36 (2025)
- [16] Bhaskara, R.M., Srinivasan, N.: Stability of domain structures in multi-domain proteins. *Sci. Rep.* **1**(1), 40 (2011)
- [17] Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
- [18] Li, S., Chen, J., Liu, B.: Protein remote homology detection based on bidirectional long short-term memory. *BMC Bioinformatics* **18**(1), 443 (2017)

- [19] Boadu, F., Lee, A., Cheng, J.: Deep learning methods for protein function prediction. *Proteomics* **25**(1-2), 2300471 (2025)
- [20] Xuan, W., Liu, N., Huang, N., Li, Y., Wang, J.: CLPred: a sequence-based protein crystallization predictor using BLSTM neural network. *Bioinformatics* **36**(Suppl.2), 709–717 (2020)
- [21] Liu, J., Tang, X., Guan, X.: Grain protein function prediction based on self-attention mechanism and bidirectional LSTM. *Brief. Bioinform.* **24**(1), 493 (2023)
- [22] Gong, Y., Li, R., Fu, B., Liu, Y., Wang, J., Li, R., Chen, D.Z.: A CNN-LSTM ensemble model for predicting protein-protein interaction binding sites. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **20**(6), 3588–3599 (2023)
- [23] Coello Coello, C.A., Lamont, G.B., Veldhuizen, D.A.: *Evolutionary Algorithms for Solving Multi-objective Problems*, 2nd edn. Genetic and Evolutionary Computation. Springer, New York, NY (2007)
- [24] Wang, Y., Fan, R., Cheng, L., Gong, B., Liu, J.: An improved adaptive NSGA-II with multiple filtering for high-dimensional feature selection. *Electronics (Basel)* **15**(1), 236 (2026)
- [25] Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Trans. Evol. Comput.* **18**(4), 577–601 (2014)
- [26] Wang, J., Chitsaz, F., Derbyshire, M.K., Gonzales, N.R., Gwadz, M., Lu, S., Marchler, G.H., Song, J.S., Thanki, N., Yamashita, R.A., Yang, M., Zhang, D., Zheng, C., Lanczycki, C.J., Marchler-Bauer, A.: The conserved domain database in 2023. *Nucleic Acids Res.* **51**(D1), 384–388 (2023)
- [27] Varadi, M., *et al.*: AlphaFold protein structure database in 2024: providing structure coverage for over 214 million protein sequences. *Nucleic Acids Research* **52**(D1), 368–375 (2024) <https://doi.org/10.1093/nar/gkad1011>
- [28] The UniProt Consortium: UniProt: the universal protein knowledgebase in 2025. *Nucleic Acids Research* **53**(D1), 609–617 (2025) <https://doi.org/10.1093/nar/gkae1010>
- [29] Letunic, I., Bork, P.: 20 years of the SMART protein domain annotation resource. *Nucleic Acids Research* **46**(D1), 493–496 (2018) <https://doi.org/10.1093/nar/gkx922>
- [30] Mi, H., Lazareva-Ulitsky, B., Loo, R., Kejariwal, A., Vandergriff, J., Rabkin, S., Guo, N., Muruganujan, A., Doremiex, O., Campbell, M.J., Kitano, H., Thomas, P.D.: PANTHER database of protein families, subfamilies, functions

- and pathways. *Nucleic Acids Research* **33**(Database issue), 284–288 (2005) <https://doi.org/10.1093/nar/gki078>
- [31] Hulo, N., Bairoch, A., Bulliard, V., Cerutti, L., De Castro, E., Langendijk-Genevaux, P.S., Pagni, M., Sigrist, C.J.A.: The PROSITE database. *Nucleic Acids Research* **34**(Database issue), 227–230 (2006) <https://doi.org/10.1093/nar/gkj063>
 - [32] Durham, J., Zhang, J., Schaeffer, R.D., Cong, Q.: DPAM-AI: a domain parser for AlphaFold models powered by artificial intelligence. *Bioinformatics* **41**(1), 740 (2024)
 - [33] Schaeffer, R.D., Zhang, J., Medvedev, K.E., Kinch, L.N., Cong, Q., Grishin, N.V.: ECOD domain classification of 48 whole proteomes from AlphaFold structure database using DPAM2. *PLoS Comput. Biol.* **20**(2), 1011586 (2024)
 - [34] Triant, D.A., Pearson, W.R.: Most partial domains in proteins are alignment and annotation artifacts. *Genome Biol.* **16**(1), 99 (2015)
 - [35] Nogales, E., Louder, R.K., He, Y.: Cryo-EM in the study of challenging systems: the human transcription pre-initiation complex. *Curr. Opin. Struct. Biol.* **40**, 120–127 (2016)
 - [36] Sharma, S., Zheng, H., Huang, Y.J., Ertekin, A., Hamuro, Y., Rossi, P., Tejero, R., Acton, T.B., Xiao, R., Jiang, M., Zhao, L., Ma, L.-C., Swapna, G.V.T., Aramini, J.M., Montelione, G.T.: Construct optimization for protein NMR structure analysis using amide hydrogen/deuterium exchange mass spectrometry. *Proteins* **76**(4), 882–894 (2009)
 - [37] Kermani, A.A.: A guide to membrane protein x-ray crystallography. *FEBS J.* **288**(20), 5788–5804 (2021)
 - [38] Grabowicz, M.: Lipoproteins and their trafficking to the outer membrane. *EcoSal Plus* **8**(2), 67–76 (2019)
 - [39] Bushuiev, A., Bushuiev, R., Sedlar, J., Pluskal, T., Damborsky, J., Mazurenko, S., Sivic, J.: Revealing data leakage in protein interaction benchmarks (2024) [arXiv:2404.10457](https://arxiv.org/abs/2404.10457) [cs.LG]
 - [40] Kawashima, S., Kanehisa, M.: Aaindex: Amino acid index database. *Nucleic Acids Research* **28**(1), 374 (2000) <https://doi.org/10.1093/nar/28.1.374>
 - [41] Cui, Y., Jia, M., Lin, T.-Y., Song, Y., Belongie, S.: Class-balanced loss based on effective number of samples. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019)
 - [42] Asilian Bidgoli, A., Rahnamayan, S.: Multi-phase evolutionary search space

shrinking for large-scale multi-objective feature selection. Appl. Soft Comput.
184(113755), 113755 (2025)

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [supplementarymaterials.zip](#)