BBA - Proteins and Proteomics 1872 (2024) 140985

Contents lists available at ScienceDirect

BBA - Proteins and Proteomics

journal homepage: www.elsevier.com/locate/bbapap

Research Paper

FunPredCATH: An ensemble method for predicting protein function using CATH

Joseph Bonello [a,b,*], Christine Orengo [a]

[a] Department of Structural and Molecular Biology, University College London, Gower Street, London WC1E 6BT, United Kingdom
[b] Department of Computer Information Systems, University of Malta, Faculty of ICT, Msida, MSD 2080, Malta

ARTICLE INFO

Keywords:
Protein function prediction
Homology
Ensemble prediction
CAFA3
Gene ontology

ABSTRACT

Motivation: The growth of unannotated proteins in UniProt increases at a very high rate every year due to more efficient sequencing methods. However, the experimental annotation of proteins is a lengthy and expensive process. Using computational techniques to narrow the search can speed up the process by providing highly specific Gene Ontology (GO) terms.
Methodology: We propose an ensemble approach that combines three generic base predictors that predict Gene Ontology (BP, CC and MF) terms from sequences across different species. We train our models on UniProtGOA annotation data and use the CATH domain resources to identify the protein families. We then calculate a score based on the prevalence of individual GO terms in the functional families that is then used as an indicator of confidence when assigning the GO term to an uncharacterised protein.
Methods: In the ensemble, we use a statistics-based method that scores the occurrence of GO terms in a CATH FunFam against a background set of proteins annotated by the same GO term. We also developed a set-based method that uses Set Intersection and Set Union to score the occurrence of GO terms within the same CATH FunFam. Finally, we also use FunFams-Plus, a predictor method developed by the Orengo Group at UCL to predict GO terms for uncharacterised proteins in the CAFA3 challenge.
Evaluation: We evaluated the methods against the CAFA3 benchmark and DomFun. We used the Precision, Recall and F_{max} metrics and the benchmark datasets that are used in CAFA3 to evaluate our models and compare them to the CAFA3 results. Our results show that FunPredCATH compares well with top CAFA methods in the different ontologies and benchmarks.
Contributions: FunPredCATH compares well with other prediction methods on CAFA3, and the ensemble approach outperforms the base methods. We show that non-IEA models obtain higher F_{max} scores than the IEA counterparts, while the models including IEA annotations have higher coverage at the expense of a lower F_{max} score.

1. Introduction

As more powerful sequencing machines become more available, efficient, and affordable, the number of sequences for analysis has also increased rapidly. A 2021 report by The UniProt Consortium shows that between January 2011 and January 2020, the size of UniProtKB/TrEMBL grew from around 13 million to over 189 million sequences, while the number of manually curated sequences in UniProt/SwissProt remained around 500, 000, indicating that numerous proteins remain unannotated [8].

Improving the understanding of proteins has a critical impact on their role as essential building blocks in different organisms and the consequences of mutations that alter their function and role in disease. Computational techniques improve the process of understanding protein function by providing insights into the locations where the protein acts and what activities it might be involved in. The current state-of-the-art protein function prediction methods, however, rely on limited and biased training data that makes it challenging to accurately predict the functions of proteins with novel or rare sequences. Additionally, methods struggle with the complex and context-dependent nature of protein functions, leading to potential inaccuracies in the functional annotations and predictions.

* Corresponding author at: Department of Structural and Molecular Biology, University College London, Gower Street, London WC1E 6BT, United Kingdom.
E-mail address: joseph.bonello.15@ucl.ac.uk (J. Bonello).

https://doi.org/10.1016/j.bbapap.2023.140985
Received 7 August 2023; Received in revised form 5 December 2023; Accepted 6 December 2023
Available online 19 December 2023

Proteins are responsible for various tasks in an organism, such as establishing structure and supporting cells, facilitating biochemical reactions, signalling transmission, repelling invading pathogens, regulating gene transcription, binding, and transporting small molecules within cells.

Rost et al. [29] defined protein function "*as all that happens by means of, and to, a protein*". Other authors define protein function as the variety of possible activities through proteins, from biochemical to biological to phenotypic activities [5,29,33]. From a computational point of view, Friedberg and Radivojac [16] provide a more practical definition to evaluate the prediction of functions since the problem of assigning the best consistent sub-graph (from the Gene Ontology) for the new protein and a score that indicates the level of confidence in the prediction.

The Gene Ontology (GO) is widely used in Biology and Protein Function Prediction (PFP). GO attempts to formalise a "controlled vocabulary of terms" to describe various aspects of biological systems. It is subdivided into three sub-ontologies: the Biological Process ontology, which describes the biological objectives a protein contributes to; the Molecular Function ontology, which contains terms that explain the biochemical activity of a protein; and the Cellular Component ontology, whose terms refer to the place where the protein is active [3].

Given a controlled vocabulary, such as the one provided by GO, that is agreed to by the community and used in popular databases, the computational prediction of protein function can be defined as the process in which models are built to identify patterns in protein properties that are related to function and map these to terms in the controlled vocabulary. These models are then applied to uncharacterised proteins, using the same features to transfer the terms that describe the function.

Homology transfer is one of the most common approaches to PFP, the underlying assumption being that similar sequences have a higher probability of having the same function. Homologous protein sequences descend from a common ancestral sequence, and the observed similarities are characteristics shared with a common ancestor.

Homologues can be further subdivided into orthologues and paralogues. Orthologues arise through a speciation event and can be found in different species. On the other hand, paralogues are the result of a gene duplication event within a species and can exist in different species as the gene duplication event could have occurred before speciation (differentiated by the terms inparalogue and outparalogue to refer to paralogues that occur after and before speciation - see [20] for a review).

The significance of these concepts is that orthologues are more likely to maintain the same or similar function in different species, while paralogues may evolve new functions even though they retain the sequence similarity to their parents [1,21]. From a PFP perspective, methods that rely on homology search for the degree of identity of the query protein with known proteins. A high degree of similarity (above 60%) indicates that homologues have a high probability of sharing the same function [10,15,27,30].

The Critical Assessment of protein Function Annotation (CAFA) challenge [35] is a timed community assessment that assesses the current state of protein function prediction methods. It presents an opportunity to understand the different methods and the data that the methods used to predict protein function. CAFA also provides a good benchmark against which to compare novel algorithms and enhancements to existing ones.

CAFA is organised over three phases. The first phase consists of a prediction stage, where the organisers release numerous proteins and predictors apply their methods to compute the putative function of the proteins. A waiting phase follows, where biological curators deposit annotations for proteins in UniProt/SwissProt and UniProtGOA [9]. In the third phase, the scores for the methods (based on the annotations accrued for the proteins in stage 2) are calculated. The proteins used in the assessment in the third phase can be used outside the challenge to evaluate novel prediction methods.

CAFA has two types of benchmarks, a No Knowledge (NK) and a Limited Knowledge (LK) benchmark in two evaluation modes, full-mode and partial-mode. In the NK benchmark, proteins are selected if they do not have experimentally verified annotations during the prediction phase but would have accumulated annotations during the waiting phase. The Limited Knowledge benchmark, introduced in CAFA2, includes proteins with experimental annotations in one or two ontologies at release and accumulated experimental annotations during the waiting phase. The full mode is used for general-purpose methods that predict function for all proteins, whereas partial mode is useful for methods that predict specific targets selected by the method developers [35].

The CAFA challenge highlights the plurality of data models and approaches involved in Protein Function Prediction (PFP). These methods attempt to discover a mapping from a protein's sequence to GO terms describing the function and where it acts.

We have developed a homology-based method using CATH FunFams, described further below. We integrated our approach with another homology-based method, domain-centric Gene Ontology (dcGO) predictor pioneered by Fang and Gough [14]. We compared our method to some of the methods that rank in the top-10 in CAFA and which are described below, to compare our method against the results they obtained (referred to as CAFA3 Rank 1 and CAFA3 Rank 10 in Table 3).

Our approach is a generic ensemble that allows base predictors to be added to the ensemble, which uses a 'mixture of experts' approach. We use Set Intersection and Set Union to provide precision-based and recall-based models to combine the predictions from the underlying base models. This method allows diverse predictors to be combined to improve the overall predictions and increase coverage.

One approach to predict protein function involves using the k-Nearest Neighbours (kNN) algorithm. kNN is a supervised Machine Learning (ML) algorithm used for estimating the probability of a data point joining a cluster or another, depending on its proximity to the cluster.

Koskinen et al. [18] developed a high-throughput GO annotation tool named *Protein ANNotation with Z-score (PANNZER)* as a high-throughput annotation tool to reliably annotate proteins. It uses statistical testing in addition to kNN to maximise the evidence when annotating proteins. The process scans the target sequence against a database to collect a list of homologues (called the Sequence Similarity Result List, SSRL). The list is partitioned into clusters of similar proteins, which are evaluated using a regression model. The prediction process results from enrichment analysis of the GO classes in the SSRL.

Another PFP method that ranks in the top 10 and uses the kNN algorithm is MS-kNN [19]. This uses three data sources for the prediction method: protein sequence data, microarray expression data and protein-protein interaction data. In the kNN algorithm, the similarity between two proteins in the three data sources is scored based on the data composition of the data source, which formed the baseline. A second classifier (called *Lin-sim kNN* considers proteins annotated with similar functions in addition to neighbourhood proteins with the same function. The scores from the baseline and the *Lin-sim kNN* classifiers were finally integrated using various strategies, and the result was used for PFP.

You et al. [34] developed *GOLabeler*, a Machine Learning method based on five sequence-derived methods. These features include GO term frequency, homology-based inference (through BLAST-kNN), the frequency of amino acid trigrams, InterPro [17] features and ProFET [23] features. GOLabeler uses a Learn-To-Rank (LTR) method to predict protein function, where predicted GO terms from the base methods are combined, and the top terms are selected for ranking by LTR.

INGA is a predictor that uses a consensus among predictors and uses Protein-Protein Interaction (PPI) networks (using the STRING database [32]), sequence similarity (using BLAST [2]) and domain assignments (from the PFAM database [7]) to train the methods [26]. These sources generated 36 models and applied them to a training set of 10,000 experimentally annotated SwissProt proteins. The consensus score in

INGA maximises the F-score, representing a given prediction's quality.

The methods reviewed above are methods that ranked in the top 10 in the CAFA3 challenge. Another recent approach exploiting CATH-FunFams is *DomFun*, a novel tool that predicts function using a tripartite network of domains, proteins, and functions [28]. DomFun uses CATH data but different techniques for this study, making it a suitable comparison method. The tripartite network is analysed to generate association values between domains and functions to generate a list of mappings between domains and functions. Protein functions are predicted by searching for the domains associated with the protein in the domain-function list. DomFun uses non-IEA SwissProt data besides CATH FunFams as the basis of the model.

The evaluation of our FunPredCATH method shows that it performed very well in partial mode on the different benchmarks, although coverage was lower than the methods we compared against. On the Biological Process ontology, FunPredCATH outperformed the top-ranked method on the NK benchmark in partial mode and obtained the same F_{max} as the top method on the LK benchmark in partial mode.

On the Cellular Component ontology, FunPredCATH obtained the same F_{max} as the 10th method in the NK benchmark in partial mode, while it ranked higher than the 10th ranked method on the LK benchmark in partial mode.

On the Molecular Function ontology, FunPredCATH outperformed the 10th ranked method on the NK benchmark in partial mode, while it outperformed the top-ranked method on the LK benchmark in partial mode.

2. Materials and method

The methods described in this paper were written in Python 3.7 with a MariaDB backend. Scalability was achieved by adopting a parallel architecture. Moreover, to improve the prediction speed, several preprocessing steps were performed (such as mapping proteins to FunFams) to improve the prediction performance. The predictor runs on Ubuntu 22.04.

2.1. Data sources

Our approach uses the CATH database [24] to identify protein domains. CATH is a hierarchical protein structure organisation maintained by the Orengo Group at UCL [31]. CATH stands for **C**lass, **A**rchitecture, **T**opology, and **H**omologous superfamilies and uses the protein structure to detect evolutionary relationships and improve the understanding of the relationship between sequence and function [31]. CATH data can be searched through the portal available at http://www.cathdb.info/ or by using an API [31]. CATH provides a hierarchical classification of protein domains for proteins obtained from the PDB database. Superfamilies in CATH group together domains sharing a clear common ancestor [31].

Homology searches involving the whole protein may not always return a characterisation to a known protein family. In such cases, analysing the domain components of the uncharacterised protein and finding functionally characterised homologues for each domain might improve the correct identification of the protein's function. This approach can be characterised by constructing a 'domain grammar' of function [6,13].

Function prediction is possible using CATH Functional Families (FunFams). CATH FunFams comprise evolutionary-related domains classified into functionally consistent sets using an entropy-based approach that segregates sets of relatives with differentially conserved residue position [11]. They have been used to identify protein functions associated with particular diseases, such as in [4].

For each FunFam, a Hidden Markov Model (HMM) is derived from the Multiple Sequence Alignment (MSA) of relatives using HMMER3. Query sequences are then scanned against the library of CATH FunFam HMMs and assigned to a FunFam if the match to the HMM is within the *E*-value threshold for the FunFam. GO terms are mapped to the FunFam, which enables the inheritance of these GO terms once a FunFam has been identified.

CATH FunFams facilitate predicting protein function by representing structural and functional units within proteins conserved across various protein sequences and families. By identifying and characterising these domains in a protein, it is possible to infer potential functions based on the known functions of similar domains.

Besides CATH FunFams, the models are trained using UniProtGOA protein annotations. The proteins from UniProtGOA that were considered consisted of those proteins for which a CATH FunFam could be identified. Moreover, the models were separately trained with experimental annotations (which we refer to as non-IEA annotations) and both experimental and electronic annotations (which we refer to as IEA annotations).

2.2. Overview of the base predictors

The prediction process uses three base predictors, two of which were developed as part of this research. The first predictor (dcGO4CATH) is an implementation of domain-centric Gene Ontology (dcGO) for CATH. dcGO is a method developed by Fang and Gough [14] for PFP using the SCOP database. The second predictor in FunPredCATH consists of set-based models applied to the CATH database.

The third method is FunFams-Plus, a PFP method developed by the Orengo Group based on CATH and one of the modules in the Orengo-FunFams method that competed in CAFA3 [12]; the Orengo-FunFams method ranked among the top-10 methods in the CAFA3 challenge [35]. FunFams-Plus is also the baseline that FunPredCATH aims to improve on.

Each predictor is given protein sequences as input. For each sequence in the input, the corresponding CATH FunFams are identified. Using the identified CATH FunFams, the base methods retrieve GO terms that exceed a preset, empirically derived threshold. The threshold is a lower-bound cut-off to identify the most promising GO terms.

2.2.1. dcGO4CATH

A homology-based approach used in PFP involves statistical techniques to predict protein function. dcGO [14] is an example of this class of methods. dcGO infers GO terms associated with SCOP [22] domains and supra-domains (accessible through the SUPERFAMILY database [25]) from annotations in UniProtGOA. The method relies on a matrix representing the frequency of GO term annotations for SCOP domains and supra-domains. A statistical test measures the overall, relative and significance inference derived from these frequencies to obtain a score used to evaluate a prediction's strength.

The underlying principle under which dcGO operates is that if a GO term annotates a set of proteins linked to a domain, it is possible to infer that the GO term can be assigned to that domain. dcGO uses the Directed-Acyclic Graph structure of GO and domain composition in SUPERFAMILY to generate GO associations for SCOP family and superfamily domains.

Two main considerations at the centre of the dcGO algorithm are:

1. structural domains generally correspond to a functional unit of a protein, and therefore, GO terms are more likely to correspond to a domain than a whole protein;
2. if a domain has more proteins annotated with a particular GO term than one would expect by chance, then it is possible to infer functional GO associations.

We have adapted the dcGO algorithm to use the CATH FunFam data and used it as one of our base prediction algorithms. Since CATH FunFams are similarly hierarchically structured as their SCOP equivalent, it was possible to adapt the algorithm using CATH FunFams. The main challenge was that, in some cases, CATH FunFams had relatively few proteins. Due to the insufficient number of proteins, it was not possible

to generate a confidence score between the CATH FunFam and the GO term in these cases.

2.2.2. Set-based methods

The set-based approach involves calculating an association score based on the prevalence of proteins with some associated GO term within a given FunFam.

Fig. 1 describes the notation used to express the equations of the methods, using a Venn diagram as an aid. A set, A or B, is a collection of elements. $A \cap B$ (verbally A intersection B), represents the common members of the sets A and B, while $A \cup B$ (verbally A union B) represents the combined members of the sets A and B. $A \backslash B$ (verbally A minus B), and $B \backslash A$ (verbally B minus A) represent the unique elements in the set A and the set B respectively, that is, the members that are in one set and not in the other.

Let sets A and B represent proteins in a specific FunFam and proteins annotated by a particular GO term, respectively. Therefore, $A \cap B$ would be all proteins within a FunFam and annotated by the considered GO term. $A \cup B$ would contain all proteins in a FunFam or annotated by a particular GO term. A would be all the proteins within the FunFam, both the proteins annotated by the considered GO term and those that are not. $A \backslash B$ would be the set of proteins in the FunFam that are not annotated by the GO term.

The set-based method employs three similarity measures to quantify the size of the intersection, namely the Jaccard, the Sørensen-Dice and the Overlap similarity measures. The equations Eq. (1), Eq. (2) and Eq. (3), respectively, show the calculations for each of the similarity measures.

$$J_{AB} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

$$SD_{AB} = \frac{2|A \cap B|}{|A| + |B|} \tag{2}$$

$$O_{AB} = \frac{|A \cap B|}{min(|A|, |B|)} \tag{3}$$

The Jaccard Similarity Index provides a comparative score of two sets by examining the distinct proteins in the two sets (represented by the set $A \cup B$) and the proteins that exist in *both* sets (represented by the set $A \cap B$).

The Sørensen-Dice Similarity Coefficient gives a more general similarity metric on vectors. The Sørensen-Dice index differs from the Jaccard index because it does not satisfy the triangle inequality (which states that the sum of the lengths of any two sides of a triangle must be greater than or equal to the remaining side). This makes the Sørensen-Dice index less susceptible to outliers, thus maintaining sensitivity in heterogeneous data sets.

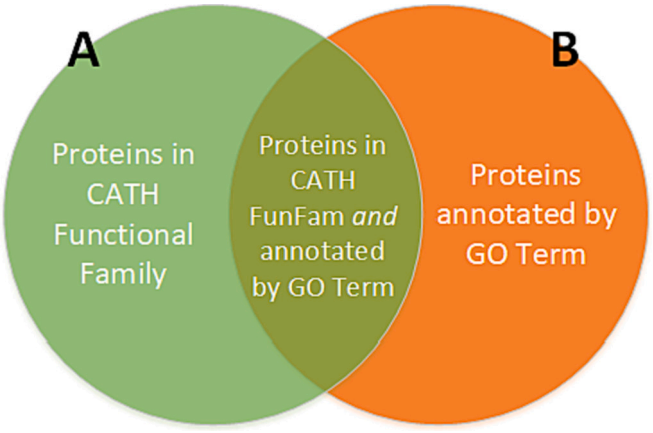

Fig. 1. Venn Diagram showing the overlap between the sets of proteins in the CATH Functional Family and the proteins annotated by the particular GO Term.

The Overlap Similarity Coefficient, also known as the Szymkiewicz-Simpson coefficient, measures the overlap between two sets. It is calculated by dividing the size of the intersection by the smaller of the two sets.

Fig. 2 shows how the confidence changes with the size of the intersection of the two sets. If only a few of the proteins in a FunFam are annotated with a GO term compared to the other proteins in the FunFam, the indexes give a low score. As the number of proteins annotated by the GO term increases, so will the score. The score, therefore, indicates confidence level when assigning the GO term to a query protein that maps to the FunFam during prediction.

The protein annotations used in this study were extracted from UniProtGOA (version 157 of June 2016, at the time of CAFA3). These data are stored in a relational database, simplifying data retrieval and processing. There were 308,273 proteins with non-IEA annotations having an identifiable FunFam associated with them, which were used for training. Similarly, 7,649,853 proteins were used for training the IEA models.

The basic CATH-FunFams algorithm is a Protein Function Prediction method developed and maintained by the Orengo Group. Given an unannotated protein sequence, the method identifies domains related to the protein and assigns them to CATH Superfamilies. After a CATH FunFam HMM search, the domains are assigned to the specific FunFams obtained from the search. Lastly, experimental GO terms are inherited from FunFams, where confidence scores are calculated and GO term assignments are up-propagated [11].

For CAFA3, CATH versions 4.0 and 4.1 were used in the predictions. Fig. 3 shows the process used to generate the predictions. In addition to the CATH-FunFams prediction algorithm described previously, FunFams-Plus also uses Pfam FunFams, InterPro data and BLAST against UniProt to predict functions when a CATH FunFam cannot be identified.

The FunFams-Plus method differs from the Orengo-FunFams method reported for CAFA3 [35] since the Orengo-FunFams method represents a modified version of FunFams-Plus that also uses Machine Learning techniques to improve the prediction process. In this study, we wanted to see how FunFams-Plus could be extended and improved by developing new algorithms.

2.3. Overview of the ensemble prediction

FunPredCATH uses a 'mixture of experts' approach based on the base predictors previously described – dcGO4CATH, the set-based methods and FunFams-Plus predictions – to create different prediction models. FunPredCATH combines the models described above and applies them to predict protein function. The resulting models attempt to consolidate the predictions to improve coverage and provide more relevant predictions.

Predictions from the individual base predictors are combined using two set-based operations: union and intersection. The union operator is a recall-oriented metric that returns all the unique GO terms predicted by the different methods. The intersection operator is a precision-oriented metric that only returns the common GO terms predicted by the different individual predictors.

FunPredCATH generates new models (see Table 1) by combining models from the previously described base methods that provide different levels of predictions. It should be noted that the base models (the set-based and dcGO4CATH) are based only on CATH data without incorporating any other data sources. If a protein's sequence cannot be matched to a CATH functional family (i.e. no homologue is identified from the HMMs), then no prediction will be made for that protein.

FunPredCATH's base models were trained using non-IEA and IEA annotations from UniProtGOA with CATH FunFams version 4.3. We wanted to investigate whether non-IEA annotations would provide improved predictions over the IEA annotations. The evaluation carried

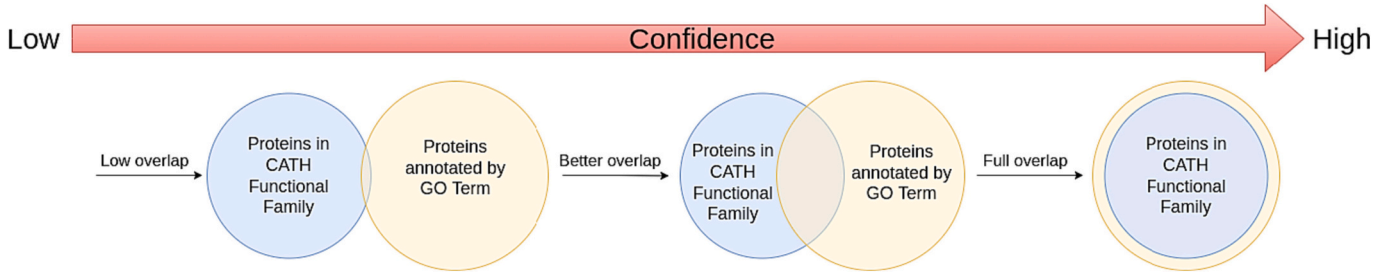

Fig. 2. The overlap between the different sizes of the intersection provides the level of confidence that is used in the predictions.

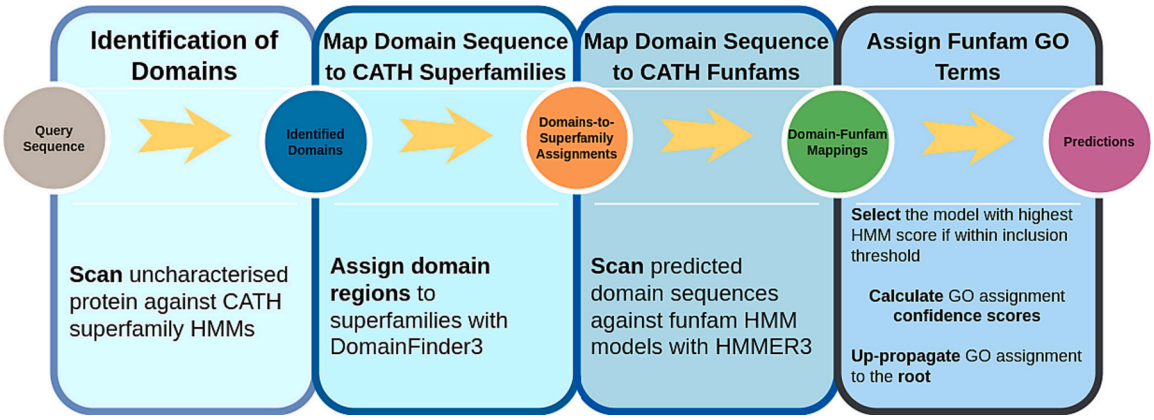

Fig. 3. FunFams-Plus Workflow.

Table 1
FunPredCATH Models.

Method Name	Methods Included in FunPredCATH Models		
	Set-Based Methods	dcGO4CATH	FunFams-Plus
Intersection Methods			
Intersected Base Predictors	✓	✓	×
Intersected Base Predictors with FunFams-Plus Intersected	✓	✓	✓
Unioned Base Predictors with FunFams-Plus Intersected	✓	✓	✓
Union Methods			
Unioned Base Predictors	✓	✓	×
Intersected Base Predictors with FunFams-Plus Unioned	✓	✓	✓
Unioned Base Predictors with FunFams-Plus Unioned	✓	✓	✓

out was intended to assess how well the models predict proteins by using only CATH FunFams; that is, if a FunFam is not identified, then the models did not attempt to find homologues (for example, by using BLAST or other means, although FunFams-Plus provides these) that could then be used to attempt or improve prediction. dcGO4CATH was trained with non-IEA data only to ensure high-quality predictions.

The IEA trained base methods (Jaccard, Sørensen and Overlap) were combined using Set Intersection and Set Union. In a separate model, the non-IEA trained models were also combined with dcGO4CATH by using the intersection and union operators similarly. Finally, Set Intersection and Set Union were applied to the models generated from the two strategies and combined with the FunFams-Plus results. FunPredCATH generates and exploits six models using intersection and union: two with just the base predictors and another four when each of the two base predictor models is unioned and intersected with FunFams-Plus.

The Set Intersection model produces predictions common across all the methods and will represent the smallest set of GO terms across the three methods: a conservative method. Although this method may not always achieve high coverage, it should be precise since it represents the agreement between the different predictors.

The Set Union model returns all the unique predictions from the different methods and represents the widest set of GO terms predicted by the three methods. This method increases the coverage and should have better recall since all the predictions from the different methods are considered equally valid.

In evaluating prediction methods and providing a reliable comparison, a common standard benchmark is required. By applying computational methods to a community-agreed standard dataset, it is possible to compare the performance of the methods and assess the overall state of the community's ability to meet its targets.

The CAFA challenge benchmark datasets provide this opportunity through an assessment for the global target of predicting protein function irrespective of the organism and secondly as a means of assessing the performance of the different methods on a species-by-species basis.

In this study, the CAFA benchmark provides the ideal independent dataset with which to compare the performance of the methods. The evaluation of the methods can be performed against an independent dataset accepted by the community with correspondingly accepted metrics.

The benchmark for evaluating FunPredCATH is taken from the CAFA3 challenge. CAFA scores prediction methods based on their general performance and specific subsets of proteins taken from benchmark organisms. CAFA evaluates predictions as follows: if T is a set of experimentally determined annotations, and P a non-empty set of predicted annotations from an ontology for a particular protein, then precision and recall are defined by Eq. (4) and Eq. (5) below:

$$pr(P,T) = \frac{|P \cap T|}{|P|} \tag{4}$$

$$rc(P,T) = \frac{|P \cap T|}{|T|} \quad (5)$$

where $|P|$ is the number of predicted items, $|T|$ is the number of terms determined experimentally and $|P \cap T|$ is the number of terms predicted but also determined experimentally. Prediction methods generally have a threshold t that is used to determine the confidence of a prediction. By varying the threshold, it is possible to plot a precision/recall curve. To obtain a single value, CAFA calculates a metric called the *maximum harmonic mean* (F_{max}), which is computed as shown in Eq. (6).

$$F_{max} = \max_t \left\{ 2 \times \frac{pr(t) \times rc(t)}{pr(t) + rc(t)} \right\} \quad (6)$$

The "No knowledge" (NK) benchmark contains proteins with no prior experimental annotations, while the "Limited knowledge" (LK) benchmark consists of proteins with partial prior experimental annotations. In the full evaluation mode, the assessors evaluate all the benchmark proteins, penalising methods for missed predictions. In the partial evaluation mode, predictions are evaluated only on the benchmark sets where at least one prediction is made.

3. Results and discussion

Table 2 shows results drawn from the evaluation of FunPredCATH in the three GO sub-ontologies, Biological Process, Cellular Component, and Molecular Function on the Limited Knowledge (NK) benchmark in partial mode.

The methods were evaluated against the benchmark Naive and BLAST methods in CAFA and DomFun, a method developed since CAFA3 that also uses the CATH database for training the models. The results for the base FunFams-Plus model that was used in FunPredCATH are also shown for comparison.

FunPredCATH outperforms the FunFams-Plus method in our CAFA3 benchmarks. The results in Table 3 show that FunPredCATH achieves higher precision rates than FunFams-Plus and higher F_{max}.

FunPredCATH also outperforms DomFun, which uses the same CATH data as our method regarding the F_{max} and the coverage across all ontologies and benchmarks. The full results can be seen in Table 3 in the supplementary information.

The comparison of the models trained with IEA and non-IEA annotations showed that the models trained with the IEA annotations generally obtained better F_{max} scores. A comparison between Table 2 and Table 2 in the supplementary material show that the IEA models outperform the non-IEA models in Molecular Function and Biological Process. In the Cellular Component ontology, the non-IEA methods achieved the same as the IEA models. However, the coverage obtained by the IEA models across the three ontologies was higher than the non-IEA models.

Concerning other CAFA methods, FunPredCATH (trained on IEA annotations) outperforms the top-ranked method in the Biological Process ontology, No Knowledge (partial mode) benchmark (0.43 and 0.40 respectively) and in the Limited Knowledge (partial mode) benchmark obtained a similar F_{max} (0.64).

In the Cellular Component ontology, No Knowledge (partial mode) benchmark, FunPredCATH obtained the same F_{max} as the tenth-ranked method (0.60). FunPredCATH outperformed the tenth-ranked method in the Limited Knowledge (partial mode) benchmark (with an F_{max} of 0.62).

In the Molecular Function ontology, FunPredCATH scored better than the tenth-ranked method in the No Knowledge (partial mode) benchmark (with an F_{max} of 0.57 and 0.54, respectively). FunPredCATH outperformed the top method in CAFA3 in the Limited Knowledge (partial mode) benchmark, with an F_{max} of 0.66 while the top method achieved an F_{max} of 0.62.

The lower coverage achieved by FunPredCATH without FunFams-Plus is due to the instances where FunPredCATH could not map proteins to a CATH FunFam since CATH only covers a subset of UniProt. Consequently, proteins do not receive annotations from the base predictors and, therefore, are not annotated.

Low F_{max} scores could also result from proteins mapping to FunFams, which are not very populated. Therefore, the base method scores are lower than the threshold prioritising predicted GO terms, resulting in missed predictions. This can be mitigated by allowing lower FunFam-to-GO term scores to be considered when the number of proteins in the identified FunFams is low, with the resulting GO terms having a lower confidence score as a result. This should improve the coverage and provide probable GO terms for the target proteins.

When the base predictor models were added to predictions from FunFams-Plus, which uses a cascade method that includes matches from PFAM-FunFams, InterPRO and BLAST to obtain predictions where no obvious CATH homologues can be found, an increase in coverage can be observed over the basic FunFams-Plus method. This indicates that using other means to find possible functional associations improves the prediction process.

4. Conclusion

The Protein Function Prediction problem remains, at present, a difficult computational problem that can have a significant impact on our understanding of the roles of proteins, especially when related to disease. We presented FunPredCATH, a set-based ensemble method used to predict protein function from sequence using a statistical-based method, set-based methods and a homology-based method (FunFams-Plus). Our results confirm that FunPredCATH compares well with similar methods while providing reasonable coverage.

Table 2
Evaluation of IEA FunPredCATH models on the CAFA3 Limited Knowledge benchmark in Partial mode.

Method	Biological Process				Cellular Component				Molecular Function			
	F_{max}	Precision	Recall	Coverage	F_{max}	Precision	Recall	Coverage	F_{max}	Precision	Recall	Coverage
Intersected Base Predictors	0.57	0.49	0.68	0.75	0.59	0.53	0.63	0.73	0.61	0.56	0.64	0.75
Unioned Base Predictors	0.50	0.45	0.57	0.75	0.55	0.48	0.62	0.73	0.50	0.51	0.49	0.75
Intersected Base Predictors With FunFams-Plus Intersected	**0.64**	**0.60**	**0.41**	0.61	**0.62**	**0.62**	**0.62**	0.47	**0.66**	**0.67**	**0.63**	0.63
Intersected Base Predictors With FunFams-Plus Unioned	0.50	0.52	0.41	0.85	0.55	0.46	0.65	0.85	0.54	0.54	0.51	0.87
Unioned Base Predictors With FunFams-Plus Intersected	0.57	0.42	0.75	0.62	0.58	0.51	0.65	0.48	0.61	0.54	0.66	0.64
Unioned Base Predictors With FunFams-Plus Unioned	0.47	0.46	0.47	0.85	0.53	0.47	0.59	0.85	0.49	0.47	0.48	0.87
Naive	0.37	0.48	0.30	**0.99**	0.53	0.57	0.49	**0.93**	0.25	0.34	0.20	**0.98**
BLAST	0.25	0.17	0.38	**0.99**	0.45	0.39	0.52	**0.93**	0.40	0.33	0.53	**0.98**
DomFun	0.49	–	–	0.55	0.60	–	–	0.51	0.62	–	–	0.49
FunFams-Plus	0.51	0.46	0.56	0.73	0.58	0.53	0.65	0.60	0.50	0.40	0.68	0.76

Table 3
Evaluation on the CAFA3 benchmarks comparing FunPredCATH (using IEA annotations) against Rank 1 and Rank 10 of the CAFA3 Top 10 methods and the Naive and BLAST benchmarks. ☆ indicates that FunPredCATH outperforms the Rank 1 method in CAFA3 while † indicates that FunPredCATH scores are at least as good as the 10th ranked method in CAFA3.

Ontology	Type	Mode	FunPredCATH Top F_{max}	FunPredCATH Coverage	CAFA3 Rank 1 F_{max}	CAFA3 Rank 1Coverage	CAFA3 Rank 10 F_{max}	CAFA3 Rank 10Coverage	Naive F_{max}	Naive Coverage	BLAST F_{max}	BLAST Coverage
BP	NK	Full	0.33	0.76	0.40	0.98	0.37	0.96	0.26	0.97	0.26	0.97
BP ★	**NK**	**Partial**	**0.43**	0.45	**0.40**	0.98	0.39	0.98	0.26	0.97	0.27	0.97
BP	LK	Full	0.49	0.61	0.60	1.00	0.54	0.82	0.36	0.99	0.25	0.99
BP ★	**LK**	**Partial**	**0.64**	0.61	**0.64**	0.83	0.56	1.00	0.37	0.99	0.25	0.99
CC	NK	Full	0.50	0.80	0.61	1.00	0.58	1.00	0.54	0.97	0.46	0.97
CC †	**NK**	**Partial**	**0.60**	0.48	0.63	0.85	**0.60**	0.81	0.55	0.97	0.46	0.97
CC	LK	Full	0.51	0.85	0.62	1.00	0.59	0.93	0.50	0.93	0.43	0.93
CC †	**LK**	**Partial**	**0.62**	0.47	0.65	0.60	**0.61**	0.94	0.53	0.93	0.45	0.93
MF	NK	Full	0.48	0.84	0.62	1.00	0.51	0.88	0.33	0.93	0.42	0.93
MF †	**NK**	**Partial**	**0.57**	0.53	0.62	1.00	**0.54**	0.34	0.34	0.93	0.43	0.93
MF	LK	Full	0.51	0.63	0.62	1.00	0.56	0.91	0.25	0.98	0.40	0.98
MF ★	**LK**	**Partial**	**0.66**	0.63	**0.62**	1.00	0.59	0.83	0.25	0.98	0.40	0.98

The main limitation of FunPredCATH lies in the statistics-based and set-based methods, which are only based on CATH, which limits the prediction of protein function if no CATH FunFam is identified. Future work in this area involves a cascade process that uses BLAST to find additional homologues in the base methods. Moreover, the base models can also be trained using other sources, such as Pfam FunFams and InterPRO, which should extend the capabilities of the models.

The scores generated by the FunPredCATH models can be used in Machine Learning models such as kNN since they can act as distance functions for proteins within the same FunFam based on the strength of the GO term annotating the protein. The expectation is that proteins whose GO terms are more represented in the CATH FunFam will form tighter sub-clusters than those whose annotations are sparser.

CRediT authorship contribution statement

Joseph Bonello: Conceptualization, Investigation, Methodology, Software, Writing – original draft, Writing – review & editing. **Christine Orengo:** Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The source code of the project, along with instructions to set up and use the tool, is available on GitHub at http://www.github.com/bonej079/PrediCATH

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.bbapap.2023.140985.

References

[1] A.M. Altenhoff, R.A. Studer, M. Robinson-Rechavi, C. Dessimoz, Resolving the ortholog conjecture: orthologs tend to be weakly, but significantly, more similar in function than paralogs, PLoS Comput. Biol. 8 (5) (2012), e1002514.

[2] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, D.J. Lipman, Basic local alignment search tool, J. Mol. Biol. 215 (3) (1990) 403–410.

[3] M. Ashburner, C.A. Ball, J.A. Blake, D. Botstein, H. Butler, et al., Gene ontology: tool for the unification of biology, Nat. Genet. 25 (1) (2000) 25–29.

[4] Paul Ashford, Camilla S.M. Pang, Aurelio A. Moya-Garcıa, Tolulope Adeyelu, Christine A. Orengo, A CATH domain functional family based approach to identify putative cancer driver genes and driver mutations, Sci. Rep. 9 (1) (2019) 1–15.

[5] G.J. Bartlett, A.E. Todd, J.M. Thornton, Inferring protein function from structure, in: P.E. Bourne, H. Weissig (Eds.), Structural Bioinformatics, John Wiley & Sons, Inc, 2005, pp. 387–407.

[6] M. Bashton, C. Chothia, The generation of new protein functions by the combination of domains, Structure 15 (1) (2007) 85–99.

[7] A. Bateman, L. Coin, R. Durbin, R.D. Finn, V. Hollich, S. Griffiths-Jones, A. Khanna, et al., The Pfam protein families database, Nucleic Acids Res. 32 (suppl_1) (2004) D138–D141.

[8] A. Bateman, M. Martin, S. Orchard, M. Magrane, R. Agivetova, S. Ahmad, E. Alpi, E.H. Bowler-Barnett, R. Britto, et al., UniProt: the universal protein knowledgebase in 2021, Nucleic Acids Res. 49 (D1) (2020) D480–D489.

[9] M. Courtot, A. Shypitsyna, E. Speretta, A. Holmes, T. Sawford, T. Wardell, M. J. Martin, C. O'Donovan, UniProt-GOA: a central resource for data integration and GO annotation, SWAT4LS 2015 (2015) 227–228.

[10] L.M. Cruz, S. Trefflich, V.A. Weiss, M.A.A. Castro, Protein function prediction, Methods Mol. Biol. 1654 (2017) 55–75.

[11] S. Das, D. Lee, I. Sillitoe, N.L. Dawson, J.G. Lees, et al., Functional classification of CATH superfamilies: a domain-based approach for protein function annotation, Bioinformatics 31 (21) (2015) 3460–3467.

[12] S. Das, I. Sillitoe, D. Lee, J.G. Lees, N.L. Dawson, et al., CATH FunFHMMer web server: protein functional annotations using functional family assignments, Nucleic Acids Res. 43 (W1) (2015) W148–W153.

[13] B.H. Dessailly, O.C. Redfern, A. Cuff, C.A. Orengo, Exploiting structural classifications for function prediction: towards a domain grammar for protein function, Curr. Opin. Struct. Biol. 19 (3) (2009) 349–356.

[14] H. Fang, J. Gough, A domain-centric solution to functional genomics via dcGO predictor, BMC Bioinforma. 14 (Suppl. 3) (2013) S9.

[15] I. Friedberg, Automated protein function prediction–the genomic challenge, Brief. Bioinform. 7 (2006) 225–242.

[16] I. Friedberg, P. Radivojac, Community-wide evaluation of computational function prediction, in: C. Dessimoz, N. Škunca (Eds.), The Gene Ontology Handbook, Springer, New York, 2017, pp. 133–146.

[17] S. Hunter, R. Apweiler, T.K. Attwood, A. Bairoch, A. Bateman, D. Binns, P. Bork, et al., InterPro: the integrative protein signature database, Nucleic Acids Res. 37 (suppl_1) (2009) D211–D215.

[18] P. Koskinen, P. Törönen, J. Nokso-Koivisto, L. Holm, PANNZER: high-throughput functional annotation of uncharacterized proteins in an error-prone environment, Bioinformatics 31 (10) (2015) 1544–1552.

[19] L. Lan, N. Djuric, Y. Guo, S. Vucetic, MS-k NN: Protein function prediction by integrating multiple data sources, in: BMC Bioinformatics 14, BioMed Central, 2013, pp. 1–10.

[20] D. Lee, O. Redfern, C. Orengo, Predicting protein function from sequence and structure, Nat. Rev. Mol. Cell Biol. 8 (12) (2007) 995–1005.

[21] Benjamin Linard, Ingo Ebersberger, Shawn E. McGlynn, Natasha Glover, Tomohiro Mochizuki, Mateus Patricio, Odile Lecompte, et al., Ten years of collaborative progress in the quest for orthologs, Mol. Biol. Evol. 38 (8) (2021) 3033–3045.

[22] A.G. Murzin, S.E. Brenner, T. Hubbard, C. Chothia, SCOP: a structural classification of proteins database for the investigation of sequences and structures, J. Mol. Biol. 247 (4) (1995) 536–540.

[23] D. Ofer, M. Linial, ProFET: feature engineering captures high-level protein functions, Bioinformatics 31 (21) (2015) 3429–3436.

[24] C.A. Orengo, A.D. Michie, S. Jones, D.T. Jones, M.B. Swindells, et al., CATH – a hierarchic classification of protein domain structures, Structure 5 (8) (1997) 1093–1109.

[25] A.P. Pandurangan, J. Stahlhacke, M.E. Oates, B. Smithers, J. Gough, The SUPERFAMILY 2.0 database: a significant proteome update and a new webserver, Nucleic Acids Res. 47 (D1) (2019) D490–D494.

[26] D. Piovesan, M. Giollo, E. Leonardi, C. Ferrari, S.C.E. Tosatto, INGA: protein function prediction combining interaction networks, domain assignments and sequence similarity, Nucleic Acids Res. 43 (W1) (2015) W134–W140.

[27] R. Rentzsch, C.A. Orengo, Protein function prediction using domain families, in: BMC Bioinformatics 14, BioMed Central, 2013, pp. 1–14, 3.
[28] E. Rojano, F.M. Jabato, J.R. Perkins, J. Córdoba-Caballero, F. Garcıa-Criado, I. Sillitoe, C. Orengo, J.A.G. Ranea, P. Seoane-Zonjic, Assigning protein function from domain-function associations using DomFun, BMC Bioinforma. 23 (1) (2022) 1–19.
[29] B. Rost, J. Liu, R. Nair, K.O. Wrzeszczynski, Y. Ofran, Automatic prediction of protein function, Cell. Mol. Life Sci. 60 (2003) 2637–2650.
[30] Burkhard Rost, Twilight zone of protein sequence alignments, Protein Eng. Des. Sel. 12 (2) (1999) 85–94.
[31] I. Sillitoe, N. Bordin, N. Dawson, V.P. Waman, P. Ashford, et al., CATH: increased structural coverage of functional space, Nucleic Acids Res. 49 (D1) (2020) D266–d273.
[32] C. Von Mering, L.J. Jensen, B. Snel, S.D. Hooper, M. Krupp, M. Foglierini, N. Jouffre, M.A. Huynen, P. Bork, STRING: known and predicted protein–protein associations, integrated and transferred across organisms, Nucleic Acids Res. 33 (suppl_1) (2005) D433–D437.
[33] J.D. Watson, R.A. Laskowski, J.M. Thornton, Predicting protein function from sequence and structural data, Curr. Opin. Struct. Biol. 15 (3) (2005) 275–284.
[34] R. You, Z. Zhang, Y. Xiong, F. Sun, H. Mamitsuka, S. Zhu, GOLabeler: improving sequence-based large-scale protein function prediction by learning to rank, Bioinformatics 34 (14) (2018) 2465–2473.
[35] N. Zhou, Y. Jiang, et al., The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens, Genome Biol. 20 (1) (2019) 244.