MACHINE LEARNING APPROACHES TOWARDS PROTEIN STRUCTURE AND FUNCTION PREDICTION

by

Aashish Jain

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Department of Computer Science

West Lafayette, Indiana

August 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Daisuke Kihara, Chair

Department of Computer Science and Department of Biological Sciences

Dr. Alex Pothen

Department of Computer Science

Dr. Jean F Honorio Carrillo

Department of Computer Science

Dr. Xavier Michel Tricoche

Department of Computer Science

Approved by:

Dr. Kihong Park

To my mother Sunita Jain and brother Anshul Jain for always supporting, encouraging and loving me

ACKNOWLEDGMENTS

The road to my doctorate journey was guided by people who have been central to the last decade of my life.

I want to begin by sincerely thanking my research advisor Dr. Daisuke Kihara for his constant support, guidance, motivation, and for his long-standing counsel that guided me to conduct research with the highest ethos possible. Thank you for giving me the opportunity to learn and make mistakes. I'm also grateful to my committee members Dr. Alex Pothen, Dr. Jean F Honorio Carrillo and Dr. Xavier Michel Tricoche for their support and constructive suggestions.

A person is defined by the company they keep, and I am extremely grateful to have found some of the best people as my friends. I want to thank Abhilash Jindal for his whole-hearted support through my early career and beyond, for instilling in me the courage to believe, and for enlightening me with fruitful discussions. I'm indebted to Sai R. M. V. Subramaniya for lending an ear to my rants and returning wise suggestions. I am also thankful to Sonali Srijan for always being there for me and raising my spirits. Many thanks to Samarth Mathur for his opportune humor. Cheers to our lifelong camaraderie.

I would like to extend my sincere thanks to all members of Kihara lab including Tunde Aderinwale, Eman Alnabati, Charles Christoffer, Ziyun Ding, Lyman Monroe, Daipayan Sarkar, Genki Terashi, Jacob Verburgt, Sean Flannery, and Xiao Wang. Thanks for all the questions, discussions, coding help, encouragement, and so many hours spent together. I am lucky to have shared my work and life with you all.

Last but not least, I am thankful to my family- my mother Sunita Jain who would have been the happiest person in the planet to see me achieve this and my brother Anshul Jain for his constant support and encouragement. I wouldn't have been here without the selfless love and blessings.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Proteins are drivers of almost all biological processes in the cell. The functions of a protein are dependent on their three-dimensional structure and elucidating the structure and function of proteins is key to understanding how a biological system operates. In this research, we developed computational methods using machine learning techniques to predicts the structure and function of proteins. Protein 3D structure prediction has advanced significantly in recent years, largely due to deep learning approaches that predict inter-residue contacts and, more recently, distances using multiple sequence alignments (MSAs). The performance of these models depends on the number of similar protein sequences to the query protein, wherein some cases similar sequences are few but dissimilar sequences with local similarities are more and can be helpful. We have developed a novel deep learning-based approach AttentiveDist which further improves over the previous state of art. We added an attention mechanism where dis-similar sequences are also used (increasing number of sequences) and the model itself determines which information from such sequences it should attend to. We showed that the improvement of distance predictions was successfully transferred to achieve better protein tertiary structure modeling. We also show that structure prediction from a predicted distance map can be further enhanced by using predicted inter-residue sidechain center distances and main-chain hydrogen-bonds. Protein function prediction is another avenue we explored where we want to predict the function that a protein will perform. The crux of the approach is to predict the function of protein based on the function of similar sequences. Here, we developed a method where we use dissimilar sequences to extract additional information and improve performance over the previous approaches. We used phylogenetic analysis to determine if a dissimilar sequence can be close to the query sequence and thus can provide functional information. Our method was ranked highly in worldwide protein function prediction competition CAFA3 (2016-2019). Further, we expanded the method with a neural network to predict protein toxicity that can be used as a safety check for human-designed protein sequences.

CHAPTER 1. INTRODUCTION

1.1 Background

All the biological processes in a cell are performed through proteins. Proteins are the key macromolecules whose interactions carry out key cell functions like maintenance, replication, reproduction and defense. Identifying the functions of individual proteins and their interactions would help us understand how the biological system operates as a whole. Protein's execution of their function is dependent on the three-dimensional structure. Thus, is it important to discover and understand the protein structure to gain a deeper knowledge of how the cell works. Experimentally studying and discovering the attributes of a protein is a slow and expensive process. With the advent of next gen sequencing technologies, proteins sequences are being discovered at a much faster pace than experimental approaches can keep up. Figure 1.1 shows that the growth in size of the standard protein sequence and structure databases. It is evident that GenBank [1], the genetic sequence database is growing much rapidly compared to structure database PDB [2] and functional annotation database SWISS-PROT [3].

Bioinformatics plays a key role in bridging this information gap between the amino acid sequence information and in-depth knowledge about protein's functionality. Computational predictive tools, once modelled, are much faster and relatively inexpensive to run. They can be useful in determining structural and functional properties for new unknown proteins, at a much larger scale than possible just through experimentation. They can also be useful in finding clues and building hypothesis for experimental biologist, accelerating experimental work. Thus, developing computational structure and function prediction methods have been one of the most important area in bioinformatics.

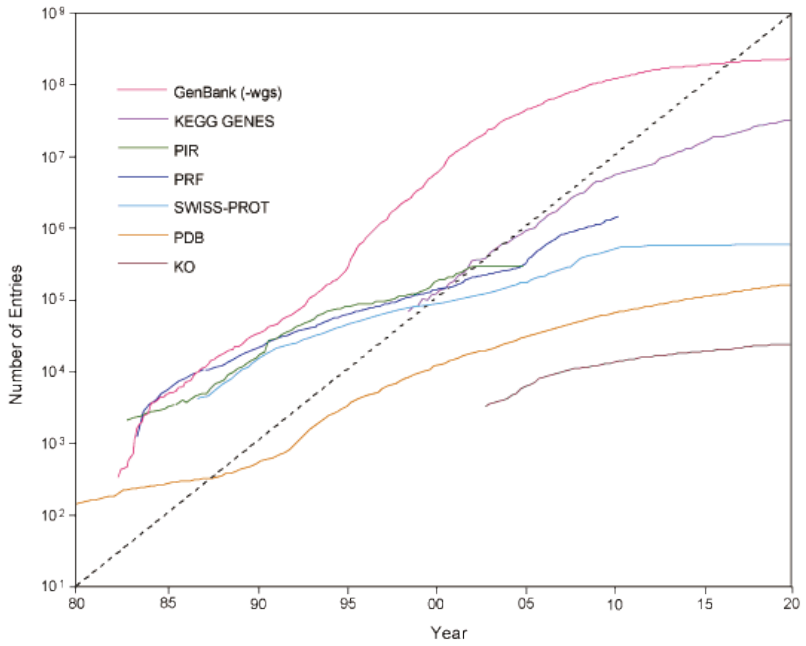


Figure 1.1. Growth of sequence and 3D structure databases. Data shown as of May 2021. Figure was obtained from https://www.kanehisa.jp/en/db_growth.html

1.2 Protein Structure Prediction

There are four levels to define a protein structure as shown in Figure 1.2. The sequence of amino acids in the protein's polypeptide chain constitutes its primary structure. Interactions between various atoms in the polypeptide chain give rise to three types of local conformations called alpha helices, beta sheets, and random coils. These are known as secondary structures of the protein. These secondary structures can also be described by the torsion angles between adjacent amino acids in the protein chain**.** Interactions between the sidechains of amino acids folds the polypeptide chain forming the overall three-dimensional shape called the tertiary structure**.** The quaternary structure of the protein is formed by the combination of multiple single chain protein subunits. The activity of a protein depends on its three-dimensional shape, where the pockets, exposed amino acids and their charges dictates which other macromolecules the protein can interact with. Structure prediction primarily involves predicting the three-dimensional structure, generally from proteins linear amino acid sequence. Accurately predicting structures can help us learn protein's function by identifying structural similarity to known structures, understand consequence of sequence/genetic changes to proteins functionality, virtual drug screening and designing novel protein.

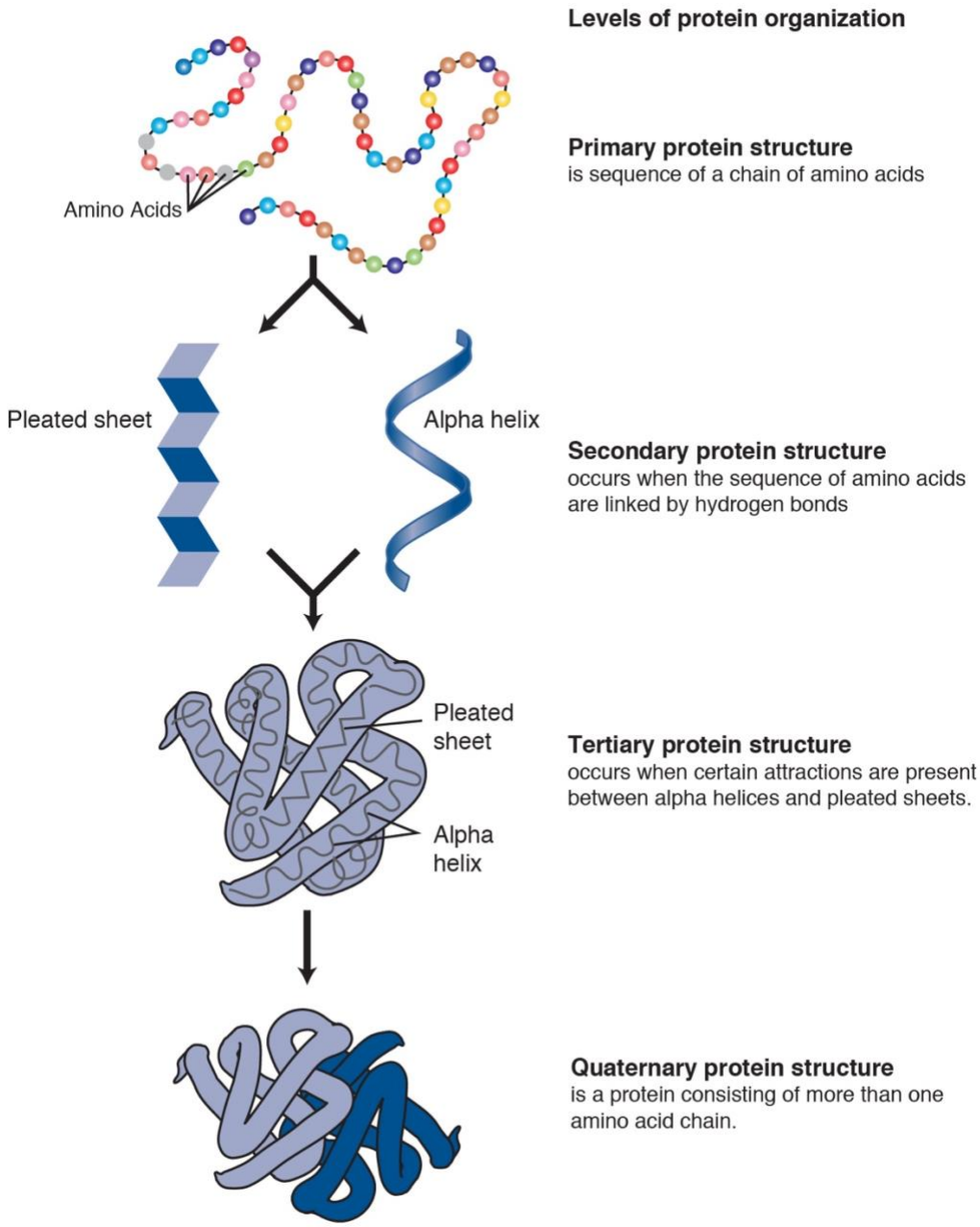


Figure 1.2. Levels of protein structureFigure was obtained from National Human Genome Research Institute, https://www.genome.gov/genetics-glossary/Protein

Traditional structure prediction methods relied on finding similar sequences called templates whose structures are known. The unknown sequence is aligned to the template sequence

and the model is built based on the template 3D structure. Templates can be found using sequence similarity tools like PSI-BLAST [4] and HHSearch [5]. This approach generates good quality protein like structures, however, it is only applicable when homologous structures are available. Approaches where structure prediction is done based on the protein sequence only are called de novo structure prediction/ template free modelling. De novo protein structure prediction is one of the most challenging problem in bioinformatics. The biggest benefit of such approaches it that because they do not rely on similar structures, they can be used to model any protein.

Recent approaches for de novo structure prediction focus on accurately predicting long-range contacts in the protein sequence which are then used to assist protein folding. The core principle behind contact prediction is detecting coevolutionary relationships between residues from multiple sequence alignments (MSAs) [6]. Previous contact map prediction approaches used direct coupling analysis to identify these relationships. These methods include CCMPred [7], PSICOV [8], Gremlin [9], EV fold [10], and plmDCA [11]. Currently, deep learning-based methods have improved contact prediction significantly. This is evident from the community-wide assessment for structure prediction, CASP13 [12] (Critical Assessment of Structure Prediction), where top-performing methods in structure prediction including AlphaFold [13] and methods in contact prediction including RaptorX [14], TripletRes [15], and ZHOU Contact [16] are all deep learning-based. Raptor-X and Alphafold also showed that predicting distance distributions instead of binary contacts can further improve the performance. The current approaches, however, are still not accurate enough to consistently achieve structure modeling with high GDT-TS structure evaluation scores [12]. Thus, further improvement is still needed.

One of the keys to accurate distance/contact prediction is the quality of MSAs [17, 18]. Recent works have used a conservative E-value cutoff to generate MSAs because using a large E-value cutoff can lead to noisier and sometimes incorrect co-evolution information in the MSA. On the other hand, a larger E-value cutoff can yield an MSA containing more sequences, which may provide useful information particularly when a query protein does not have many close homologs. The difficulty is that the appropriate level of sequence similarity depends on the protein family [19, 20]. In further chapters, we propose a new deep learning-based approach, AttentiveDist, where the model can use multiple alignment information through an attention mechanism. Attention mechanisms in deep learning models are widely used in natural language processing [21, 22] and computer vision [23, 24] for determining which regions in the sentence or image respectively are

important for a given task. In AttentiveDist, the attention mechanism determines the importance of every MSA at residue pair level, utilizing information from different MSA's to improve the performance.

Recently, DeepMind showcased AlphaFold2 [25], a new deep learning based structure prediction method in CASP14 which performed exceptionally well than all the other methods. For most of the blind targets in the competition AlphaFold2 achieved GDT-TS (Global Distance Test – Total Score) of 0.8 or more. A GDT-TS above 0.5 is generally regarded as correct fold by the community where one can safely assume that the global protein shape is correct in the predicted model. A score of 0.8 or more means that the predicted topology is correct and detailed information like side chain conformation may also match to real structure. This is a great milestone in structure prediction community. Although their paper and code is not available yet, we believe their success is attributed largely due to SE(3) transformer layer [26], a 3D rotation-translation equivariant attention layer, which predicts the 3D protein structure. This allows the deep learning model to output the 3D structure instead of distance map allowing the model to directly learn from the error it makes in structure prediction. The model was trained on 128 TPUv3 GPU's for couple of weeks, which is massively more than 1-2 GPU's used by most other methods. Such resources are generally not available in the academic labs and further development of more resource efficient model is needed.

1.3 Protein Function Prediction

Computational algorithms which can mine the functional genomic and proteomic data to accurately predict protein functions can help assign functions to newly discovered sequences as well as exploring the breadth of different functions a protein can have.

Protein functions are textually described in literature. To computationally predict the functions, they need to be transformed into a vocabulary. This has been done Gene Ontology (GO) Consortium [27] through the introduction of Gene Ontology (GO) terms. GO terms is a hierarchical set of terms that capture functional information. They provide controlled vocabularies of defined terms, where each term corresponds to a specific function of a gene/protein. These cover three domains: Cellular Component (CC), the parts of a cell or its extracellular environment; Molecular Function (MF), the elemental activities of a gene product at the molecular level, such as binding or catalysis; and Biological Process (BP), operations or sets of molecular events with a

defined beginning and end, pertinent to the functioning of integrated living units: cells, tissues, organs, and organisms. It is structured as a directed acyclic graph where each term has defined relationships to one or more other terms in the same domain, and sometimes to other domains. Figure 1.3 shows a small part of GO directed acyclic graph.

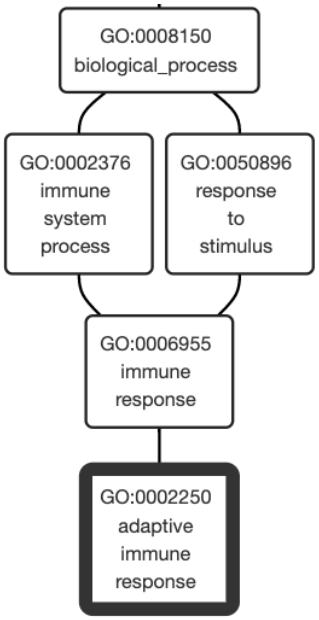


Figure 1.3. Small part of Gene Ontology graph. Figure was obtained from NaviGO tool [28], https://kiharalab.org/web/navigo/views/goparent.php

The traditional approach to predict protein function computationally is to find similar sequences and transfer their annotations. A simple method using this approach retrieves homologous sequence using similarity detection algorithms like PSI-BLAST [4] take the function of the best matching sequences which are above certain thresholds. This approach works well when similarity is very high and starts becoming unreliable with moderate to low similarity levels. This decease the breadth of sequences which can be annotated as many sequences don't have closely similar sequences in the database and can also leads to erroneous predictions for those which have. Besides sequence similarity, other techniques are also used to for predicting functions. Protein-protein interaction network based methods relies on the principle that if two proteins interact with each other, then they might also share the function [29, 30]. But these methods require a defined protein network available for functionally un-annotated proteins. Protein structure-based methods search for similarity in three-dimensional conformation [31, 32]. If two proteins have

similar protein folds, it is highly likely that they also share the function. They perform well, but needs protein structure to be available, which is generally not the case for new proteins. Genomic data based methods uses microarray data and expression pattern to determine proteins function [33]. They depend on finding genes having similar expression patterns, which might indicate co-regulation and common function. Among all these strategies, for large scale function prediction, sequence similarity remains the most efficient approach. Two main reasons for this are availability of huge sequence dataset, which is expanding every day because of high throughput sequencing, and simple requirement of the technique to work for new proteins.

Most of the sequence similarity based methods rely on PSI-BLAST to get a set of similar sequences which is further processed in different way to predict the function. ConFunc splits sequences into sub alignment and use their GO terms to deduce the function [34]. GoFDR tries to identify protein residues which are functionally discriminating, using multiple sequence alignment of homologous hits [35]. ESG uses PSI-BLAST that performs an iterative search, taking in account neighboring sequences, which can capture more general similarities [36]. SIFTER explores the evolutionary relationship among the query protein and gene family it is closely related to [37]. Most of these methods limits the use of BLAST by only considering sequences with low E-value (i.e., sequences with high similarity to query). Although the motivation behind restricting the search space to similar sequences is intuitive, it prevents prediction when closely related sequences are not available. For such 'hard' cases, the algorithm should be able to mine data from less similar sequences as well. This has been the bottleneck in sequence based prediction methods. PFP [38, 39] is one of the pioneer methods, which makes use of sequences with a wide range of similarity to a query ranging from significant hits to very weakly similar ones up to an E-value of 125, far larger than conventionally used thresholds, e.g. 0.001. GO terms are extracted from all the retrieved sequences; however, to reduce the risk of predicting unrelated GO terms taken from weakly similar sequences, sequences are weighted by their E-values. PFP also considers the co-occurrence of GO terms, which is statistics of GO term pairs that frequently co-occur in annotation of the same sequence. PFP was one of the top ranked function prediction methods in Critical Assessment of Functional Annotation CAFA [40] and the top in the Critical Assessment of Protein Structure Prediction (CASP) function prediction category in 2007 [41]. In the following chapters we present a new sequence-based function prediction method, Phylo-PFP, which significantly

improves prediction performance over PFP by incorporating phylogenetic information in defining sequence similarity.

GO term predictions can be further extended to predict specific properties of the protein. In the following chapters we tackle the problem of protein toxicity prediction based on GO terms. Artificially designed proteins can lead to the production of harmful proteins, either un-intentionally or intentionally. Foreseeing such effects can prevent potential harm. One solution is to check for toxicity in lab facilities that synthesis artificial proteins. This can be done through a computational algorithm that take a protein or DNA sequence as input and alerts if the protein is predicted to be toxic. In the following chapters, extending Phylo-PFP, we present a machine learning based protein toxicity prediction method, which can predict the toxicity of a query protein sequence based on the protein's predicted Gene Ontology (GO) annotation.

CHAPTER 2. PROTEIN STRUCTURE PREDICTION USING DEEP LEARNING

Protein 3D structure prediction has advanced significantly in recent years due to improving contact prediction accuracy. This improvement has been largely due to deep learning approaches that predict inter-residue contacts and, more recently, distances using multiple sequence alignments (MSAs). In this chapter I present AttentiveDist, a novel approach that uses different MSAs generated with different E-values in a single model to increase the co-evolutionary information provided to the model. To determine the importance of each MSA's feature at the inter-residue level, we added an attention layer to the deep neural network. We show that combining four MSAs of different E-value cutoffs improved the model prediction performance as compared to single E-value MSA features. A further improvement was observed when an attention layer was used and even more when additional prediction tasks of bond angle predictions were added. The improvement of distance predictions was successfully transferred to achieve better protein tertiary structure modeling.

2.1 Background

Computational protein structure prediction is one of the most important and difficult problems in bioinformatics and structural biology. Understanding protein structure can unlock information about protein function and can aid in the design and development of artificial proteins and drug molecules [42, 43]. Recently, a significant improvement in protein structure prediction has been observed due to improvements in contact and, more recently, distance map prediction [12]. The predicted contacts/distances are used to drive computational protein folding, where the 3D atomic protein structure is predicted without the need for template structures [44].

The core principle behind modern contact prediction is detecting coevolutionary relationships between residues from multiple sequence alignments (MSAs) [6]. Previous contact map prediction approaches used direct coupling analysis to identify these relationships. These methods include CCMPred [7], PSICOV [8], Gremlin [9], EV fold [10], and plmDCA [11]. The next wave of methods, which represents the current state of the art, uses deep learning to predict contacts/distances. Deep learning-based methods have improved contact prediction significantly.

This is evident from the recent community-wide assessment for structure prediction, CASP13 [12] (Critical Assessment of Structure Prediction), where top-performing methods in structure prediction including AlphaFold [13] and methods in contact prediction including RaptorX [14], TripletRes [15], and ZHOU Contact [16] are all deep learning-based. Raptor-X and Alphafold also showed that predicting distance distributions instead of binary contacts can further improve the performance. However, the current approaches are still not accurate enough to consistently achieve structure modeling with high GDT-TS structure evaluation scores [12]. Thus, further improvement is still needed.

One of the keys for accurate distance/contact prediction is the quality of MSAs [17, 18]. Recent works have used a conservative E-value cutoff to generate MSAs because using a large E-value cutoff can lead to noisier and sometimes incorrect co-evolution information in the MSA. On the other hand, a larger E-value cutoff can yield an MSA containing more sequences, which may provide useful information particularly when a query protein does not have many close homologs. The difficulty is that the appropriate level of sequence similarity depends on the protein family [19, 20].

Here, we propose a new deep learning-based approach, AttentiveDist, where the model can use multiple alignment information through an attention mechanism. AttentiveDist uses a set of MSAs that are obtained with different E-value cutoffs, where the deep-learning model determines the importance of every MSA using an attention mechanism. Attention mechanisms in deep learning models are widely used in natural language processing [21, 22] and computer vision[23, 24] for determining which regions in the sentence or image respectively are important for a given task. To better generalize the model, we used a multi-tasking approach, predicting backbone angles and orientation angles [45] together with inter-residue distance. We also show that structure prediction from a predicted distance map using Rosetta [46] can be improved by using predicted inter-residue sidechain center distances and main-chain hydrogen-bonds. The predicted distances and angles are converted into potentials using neural network-predicted background distributions.

We show that the deep learning based inter-residue distance prediction benefits from using multiple MSA's. We compared distance predictions using combinations of individual MSAs of different E-value cutoffs with the attention-based approach, showing that the latter achieved a better precision. We also demonstrate that the attention given to different MSA-based features in

AttentiveDist is correlated to the co-evolutionary information in the MSA. Finally, we show that in structure modelling, additional constrains of predicted inter-residue sidechain center distances and main-chain hydrogen-bonds improves structure prediction.

2.2 Methods

2.2.1 Training, validation, and test datasets

For the training and validation dataset, we took proteins from the PISCES [47] database that consists of a subset of proteins having less than 25% sequence identity and a minimum resolution of 2.5 angstroms, released in October 2019. We further pruned this dataset by removing proteins that contain more than 600 or less than 50 amino acids and those released after 1st May 2018 (i.e. the month of beginning of CASP13). Next, proteins that have intermediate gaps of more than 50 residues, not considering the termini, were removed. Finally, a protein that has 2-letter chain names was removed because PISCES capitalizes chain names making it confusing for cases where the real 2 letter chain name has both mixed lowercase and uppercase alphabets used. This resulted in 11,181 proteins. Out of those, 1,000 proteins were selected randomly as the validation set, and the rest were used to train the models. For each instance of glycine, a pseudo-Cβ atom was built to be able to define Cβ-Cβ distance by converting it to alanine.

CASP13 FM and FM/TBM domains were used as the test set, containing 43 domains (across 35 proteins). The full protein sequence was used in the input instead of the domain to replicate the CASP13 competition.

2.2.2 MSA generation

To generate the MSA we used the DeepMSA [17] pipeline. This pipeline consists of three stages where three different databases are searched to obtain similar sequences, which produces better MSAs compared to a single database search. The packages used for DeepMSA were HHsuite [48] version 3.2.0 and HMMER [49] version 3.3. The sequence databases we used were released before the CASP13 competition began for the sake of fair comparison, and were: Uniclust30[50] database dated October 2017, Uniref90 [51] dated April 2018, and Metaclust_NR [52] database dated January 2018. We generated 4 different MSAs with E-value 0.001, 0.1, 1, and 10 used in HHsuite [48] and HMMER [23, 49].

2.2.3 Network parameters and training

In AttentiveDist the convolution filter (kernel) size is 5x5 for the first 3 blocks and then 3x3 for the rest of the network, and the channels were kept constant to 64. We also added dilation to increase the receptive field of the network, with dilation cycling through 1,2 and 4.

The loss function used during training is the weighted combination of individual objective loss. For each objective cross-entropy loss was used and the weights were manually tuned. Distance and orientation angles losses were given weight of 1 while the backbone φ and ψ angle losses were given weight of 0.05 each. The Adam [53] optimizer with a learning rate of 0.0001 was used. Dropout probability was 0.1. Dilations were cycled between 1,2 and 4. The learning rate, dropout and loss weights were tuned on the validation dataset. We trained the model for 80 epochs. Batch size was set to 1 because of GPU memory constraints.

2.2.4 Sidechain center distance and backbone hydrogen-bond (N-O) prediction

For the tertiary structure modeling, we tested the inclusion of two additional predicted distance constraints, distances between Side-Chain cEnters (SCE) and distances between the nitrogen and the oxygen (N-O) in peptide bonds. These distances were binned similarly to the Cβ – Cβ distances. The first bin was for a distance between 0 to 2 Å, bins up to 20 Å were of a width of 0.5 Å, followed by a bin of size 20 Å to infinite. A bin for residue pairs with missing information was also added. For prediction, we used networks with 25 ResNet blocks, which is smaller than the one in Figure 2.1**.** The model was trained on the E-value 0.001 MSA data (Figure 2.2). The prediction performance for SCE distances and N-O distances are is shown in Table 2.1.

2.2.5 Protein 3D structure generation from distance prediction

We performed protein structure modeling similar to the work by Yang et al. [45] We used Rosetta's protein folding and energy minimization protocols with customized constraints. The constraints were computed from our predictions of distance distributions (Cβ-Cβ, SCE-SCE, and backbone N-O) and angle distributions (backbone φ−ψ and the three residue-pair orientation angles) by normalizing the predicted values with predicted reference distributions. For both distance and angle constraints, the predicted distributions were converted to an energy potential as follows:

$$ene(i) = -log\left(\frac{P_i}{REF_i}\right), i = 1, 2, \ldots N, \qquad \text{(Eq. 2.1)}$$

where Pi and REFi are the predicted probability and the reference probability of i-th bin, respectively. N is the number of bins in the predicted distribution.

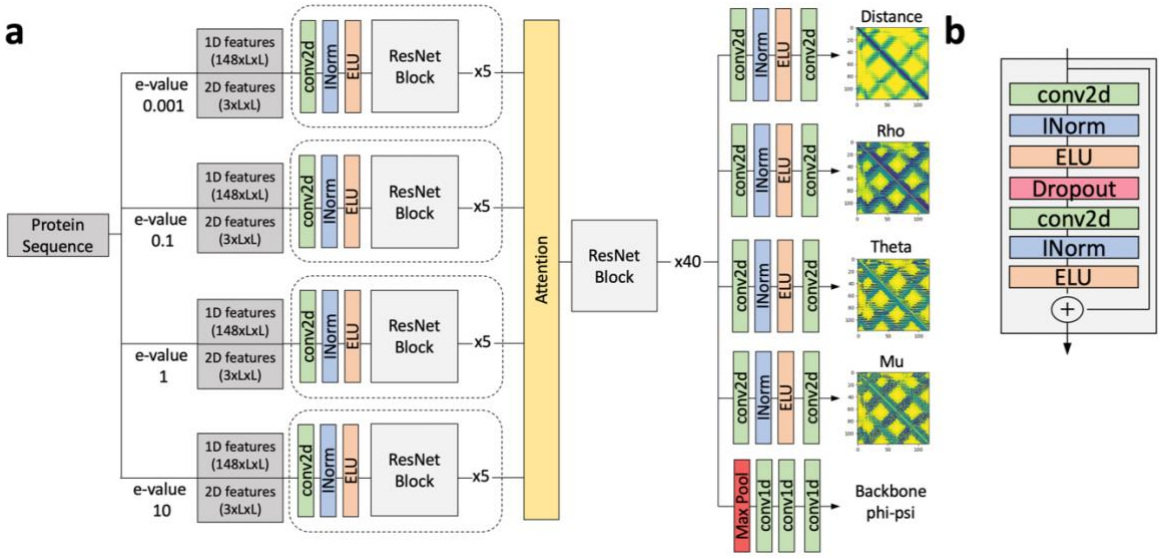


Figure 2.1. The network architecture of AttentiveDist. a. The overall architecture. From sequence-based features computed from a set of MSA's of different E-values and 2D features, AttentiveDist uses ResNet with attention mechanism to predict Cb-Cb distances, three side-chain orientation angles, and backbone j, ψ angles.. Dotted box represents weights are shared. b. Layers in a single ResNet Block. conv2d (green), 2d convolution layer; INorm (blue), instance normalization; ELU (orange), Exponential Linear Unit.

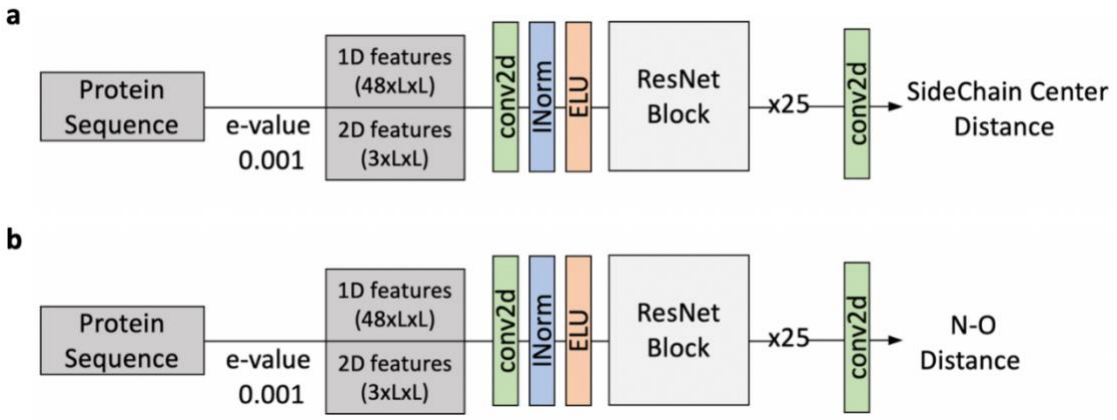


Figure 2.2. ResNet model architecture for a, Sidechain Center (SCE) distances and b, backbone peptide N-O pairwise distance prediction. The ResNet Block is the same as described in Figure 2.1b. conv2d (green) is 2d convolution layer, INorm (blue) is instance normalization, ELU (orange) is Exponential Linear Unit.

Table 2.1. Long range precision of prediction made for Side-Chain cEnters (SCE) contact and contact between the nitrogen and the oxygen (N-O) in peptide bonds. The contact is defined if as pairs within 8 Å for SCE-SCE, and 4 Å for N-O. The 43 CASP13 FM and FM/TBM targets were considered.

Prediction	L/5	L/2	L/1
SCE	0.688	0.530	0.410
N-O	0.856	0.744	0.545

The reference probability distributions of three distances, backbone angles, and the side-chain orientation angles were predicted with a five-layer fully-connected neural networks. A network of the same architecture was trained for each type of constraints. For a distance type, the features used were the positions *i* and *j* of the two amino acids, the length of the protein, and a binary feature of whether a residue is glycine or not [13]. For angle predict we also included the one-hot encoding of the amino acid type.

All energy potentials were smoothed by the spline function in Rosetta, and then used as constraints in the energy minimization protocol. The energy potentials of distances (Cβ-Cβ, SCE-SCE and backbone N-O) and inter-residue orientations were split into L/10*L/10 blocks. To explore a diverse conformational space, the blocks of the potentials were randomly added to the energy function in the minimization steps. We generated 4,000 decoy models with different folding paths (i.e. additions of the blocks of potentials) and weight parameters that balance the energy terms. All decoy models were ranked by ranksum [54], a sum of the ranks of three scoring functions, GOAP [55], DFire [56], and ITScore [57]. The best scoring model was selected as the predicted structure.

2.3 Results

2.3.1 AttentiveDist architecture

AttentiveDist predicts the distribution of Cβ - Cβ distance and three side-chain orientation angles for each amino acid residue pair, as well as backbone dihedral angles. Its uses a deep learning framework, ResNet [58], with an attention mechanism that identifies important regions in MSAs.

Figure 2.1 shows the network structure of AttentiveDist. The network is derived from ResNets [58], where each residual block consists of convolution layers followed by instance

normalization [59] and exponential linear unit [60] (ELU) as the activation function. This set is repeated twice with a dropout layer in between to form one residual block. The first 5 residual blocks are feature encoding layers and the weights are shared for the different inputs generated by 4 MSAs of E-values 0.001, 0.1, 1, and 10. For multiple different MSA feature encoding, we use soft attention to automatically determine the most relevant MSA for each pair of residues. An attention weight vector a of size k is computed for every i, j pair of residues, where k is the number of different MSAs used. Let X_m be the encoded feature matrix for MSA m. a_m is a scalar value that represents the "attention" or importance given to encoded feature $X_{m(i,j)}$, which is computed using Equation 1. The matrix W in Equation 1 is chosen such that e_m is scalar, and it is learned during training along with the other parameters of the network. The attended feature matrix Y is computed as the weighted sum of different MSA encoded features where the weight is attention given as shown in Equation 2. The intuition is that Y captures the relevant information from multiple different MSAs.

$$a_m = \frac{\exp e_m}{\sum_{k=1}^{M} \exp e_k}, \text{ where } e_m = W^T X_{m(i,j)} \quad \text{(Eq. 2.2)}$$

$$Y_{i,j} = \sum_{k=1}^{M} a_k X_{k(i,j)} \quad \text{(Eq. 2.3)}$$

The attended features are then passed through 40 residual blocks. The model branches into 5 different paths with different outputs after the last residual block. In each path there is an additional convolution layer followed by normalization and activation which learn task-specific representations. To improve the generalization, we used a multi-task learning approach where the model is trained on six related tasks, namely, distance prediction, three side-chain orientation angles (Figure 2.3), and the φ, ψ backbone angles. The paths for distance and orientation angles contain a final convolution layer to obtain the proper output dimension, followed by softmax activation. In the backbone φ, ψ angles path, a max pooling layer is added to reduce the dimensionality from LxLx64 to Lx64 where L is the size of the protein, followed by 1D convolution and softmax activation. The whole network is trained end-to-end. The final model is an ensemble of 5 models, where the prediction is the average of individual E-value models and the attention-based model that combines the four MSAs.

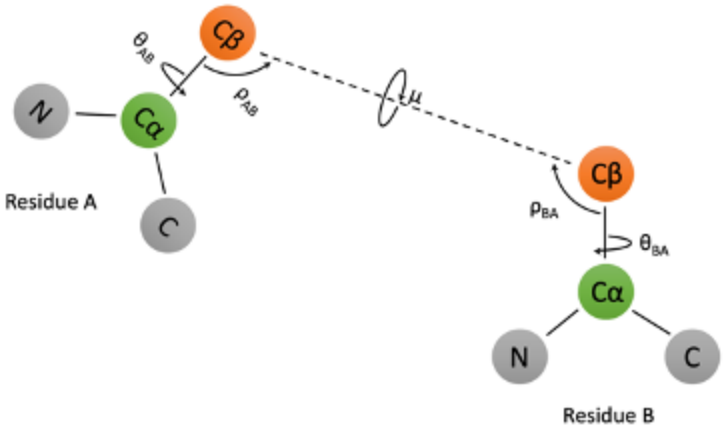


Figure 2.3. Orientation angles. The three orientation angles mu(μ), theta(θ) and rho(ρ) between any pair of residues in a protein. In the 3D structure of the protein, considering any two residues A and B, θAB represents the dihedral angle between the vectors NA -> CαA and CβA -> CβB along the axis of CαA -> CβA. ρAB represents the angle between the vectors CβA -> CαA and CβA -> CβB. θ and ρ depends on the order of residue and thus are asymmetric. μ represents the dihedral angle between the vectors CαA -> CβA and CβA -> CαB along the axis of CβA -> CβB. These orientation angles help in representing the direction of residue A to residue B and vice-versa. The orientation angles were originally described in Yang et al.23 We used different notations of angles from them to prevent confusion with conventionally used angle notation.

We used eight sequence-based input features. The 1D features are one hot encoding of amino acid type (20 features), PSI-BLAST [4] position specific scoring matrix (20 features), HMM [61] profile (30 features), SPOT-1D [62] predicted secondary structure (3 features) and solvent accessible surface area (1 feature), making a total of 74 1D features. MSAs, from which the 1D features were computed, were generated using the DeepMSA [17] pipeline. 1D features were converted into 2D features by combining features of two residues into one feature vector. We also used three 2D features, which were a predicted contact map by CCMPRED [7] (1 feature), mutual information (1 feature), and statistical pairwise contact potential [63] (1 feature). Thus, in total we used (2 x 74) + 3 = 151 L x L features, where L is the length of the protein.

The AttentiveDist network predicts the Cβ – Cβ distance of every pair of residues in a target protein as a vector of probabilities assigned to 20 distance bins. The first bin is for 0 to 4 Å, the next bins up to 8Å are of a size 0.5 Å and then bins of a 1 Å size follow up to 18 Å. The last bin added is for no-contact, i.e. for 18 Å to an infinite distance. Similarly, the backbone φ, ψ angles were binned to 36 ranges, each of which has a 10-degree range. Three side-chain orientation angles, ρ, θ, and μ (Figure 2.3) were binned into 24, 24, and 15 bins, each with a size of 15 degrees, respectively. The side-chain orientation angles were only considered between residue pairs that

are closer than 20 Å, and for the rest of the residue pairs a no contact bin was considered as the correct answer. For target values for training, the real distances and angles were converted into vectors where the bin containing the real distance/angle has value 1 and while the rest were set to 0.

The network was trained on a dataset of 11,181 non-redundant proteins, which were selected from the PISCES [47] database. Sequences released after 1st May 2018 (i.e. the month of beginning of CASP13) were removed. Every pair has less than 25% sequence identity. Out of these, 1,000 proteins were selected randomly as the validation set, while the rest were used to train the models. More details are provided in the Method section.

2.3.2 Contact prediction performance

We compared the performance of AttentiveDist with several different input MSA settings on 43 FM (Free Modeling) and FM/TBM (Template-Based Modeling) domains from CASP13. FM and FM/TBM are harder targets compared to template-based modeling because they do not have any appropriate template protein available, necessitating de-novo prediction. We used the standard metric of top L/n predicted long range contacts precision and F1 score as used in other works, where L is length of the protein and n is 1, 2, and 5. Long range contacts are defined as contacts between residues that are 24 or more residues away. Since AttentiveDist predicts residue-residue distances instead of binary contact, we converted this to contact prediction by summing the probabilities of distance bins from minimum distance to 8 Å.

We performed an ablation study of our model to understand how much different additions contribute to the performance (Table 2.2). The baseline model shown at the top of the table is a single model that predicts only Cβ-Cβ distance using an E-value of 0.001 for feature generation. 0.001 was used for E-value because it gave the overall the highest precision among the other E-values used in AttentiveDist. Next, we added multitask learning, where the model predicts the distance, 2D side-chain orientation angles, and the backbone dihedral angles together, but without attention. The multi-task learning improved the L/1 precision from 0.451 to 0.468.

The next three rows compare multi-task learning results with four different E-values (0.001, 0.1, 1, and 10). The results show that on average an E-value of 0.001 performed the best. The sixth row, “No attention, 4 E-values” shows the results of using MSAs with the four E-values to compute four different 1D features but without the attention mechanism. In this model we

concatenated the features of the 4 E-values and passed them to the network. This increased the L/1 precision to 0.472; however, the L/2 and L/5 decrease by 0.006 and 0.011, respectively. The reason of the decrease could be because the 4 MSAs were input in parallel without any weighting mechanism. We also compared with contact map probability based a MSA selection strategy [45] where for each target one prediction out of the 4 MSAs was selected based on the sum of L/1 contact probability values. Interestingly, the MSA selection performance was similar to the "No attention, 4 E-values" strategy. The next strategy, the AttentiveDist (single) model, which used the attention mechanism, improved L/1 precision further to 0.479. We also computed the average probabilities from 4 single E-value models (4 E-value (average)), which yielded L/1 precision of 0.479. Finally, we averaged the outputs from the 5 models (4 single E-value models and the model with attention), the full AttentiveDist, which resulted in a 0.14 gain to achieve 0.493 in L/1 precision. We show the L/1 precision comparison of the 43 individual targets between No attention, 4 E-values and E-value 0.001 model in Figure 2.4a. In Figure 2.4b we compare the E-value 0.001 model and AttentiveDist and in Figure 2.4c we compare No attention, 4 E-values and AttentiveDist. Overall, we show that using four different E-value MSA's improves the performance in all L/1, L/2 and L/5 precision. A similar trend was observed when F1 score was considered, where AttentiveDist (single) improved the L/1 F1 score from 0.427 to 0.442 compared to the E-value 0.001 model.

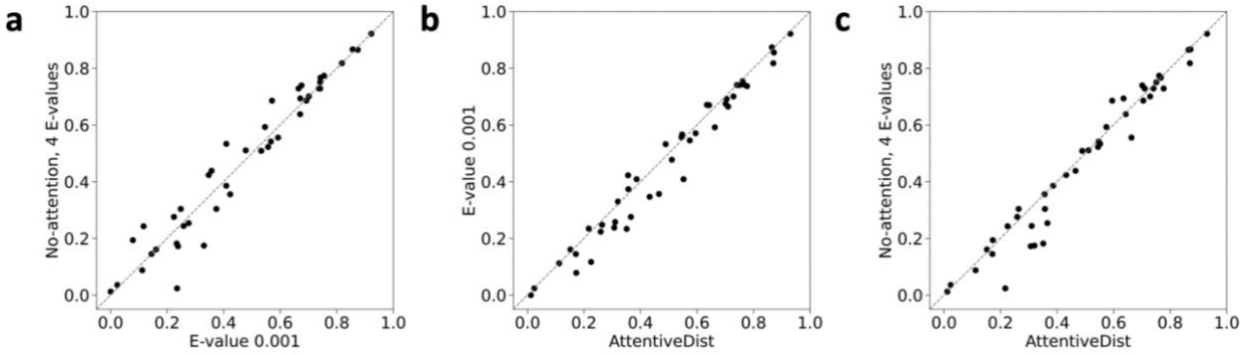


Figure 2.4. Individual target L/1 precision comparison between a, 4 E-value model without attention and E-value 0.001 model b, E-value 0.001 model and AttentiveDist c, 4 E-value model without attention and AttentiveDist. E-value 0.001 model represents the model trained with E-value 0.001 MSA features in multi-task fashion.

In Table 2.2 we also compare the performance with TripletRes [15], the second best server method in CASP13, because it used the same MSA generation pipeline, DeepMSA, with the same sequence datasets. Comparison with the same MSAs makes the comparison more informative because the performance highly depends on the input MSA. There was a significant improvement in L/1 precision of 9.3% and F1 score of 9.4% when compared to TripletRes. When compared for individual targets (structure domains), AttentiveDist had a higher L/1 precision than TripletRes for 27 domains, tied for 2 domains out of the 43 domains (Figure 2.5a). AttentiveDist had higher average precisions than RaptorX-Contact [14], the top server methods in CASP13, as shown at the bottom of Table 2.2. RaptorX has a new development after CASP14 [64], but here we compared with their results in CASP13. Comparisons of individual targets (Figure 2.5b) shows AttentiveDist showed a higher L/1 precision than Raptor-X for 23 domains and tied for 2 domains out of the 43 domains.

Table 2.2. CASP13 FM and FM/TBM 43 targets long range precision and F1 score. L/5, L/2 and L/1 shows values when top L/5, L/2 or L/1 contact predictions with the highest probabilities were considered where L is the length of the protein.

Model	Precision			F1 score		
	L/5	L/2	L/1	L/5	L/2	L/1
Distance only (E: 10^{-3})	0.700	0.586	0.451	0.224	0.359	0.411
E-value 10^{-3}	0.716	0.608	0.468	0.228	0.373	0.427
E-value 10^{-1}	0.693	0.587	0.452	0.216	0.363	0.415
E-value 1	0.724	0.589	0.455	0.230	0.362	0.414
E-value 10	0.696	0.580	0.452	0.217	0.354	0.411
No-attention, 4 E-values	0.705	0.602	0.472	0.223	0.371	0.432
MSA selection	0.713	0.604	0.472	0.226	0.373	0.433
AttentiveDist (single)	0.716	0.613	0.479	0.230	0.385	0.442
4 E-values (average)	0.744	0.619	0.479	0.238	0.383	0.445
AttentiveDist (Ensemble)	**0.746**	**0.624**	**0.493**	0.241	**0.387**	**0.454**
TripletRes	0.701	0.587	0.451	0.230	0.363	0.415
RaptorX-Contact	0.744	0.612	0.481	**0.248**	0.381	0.441

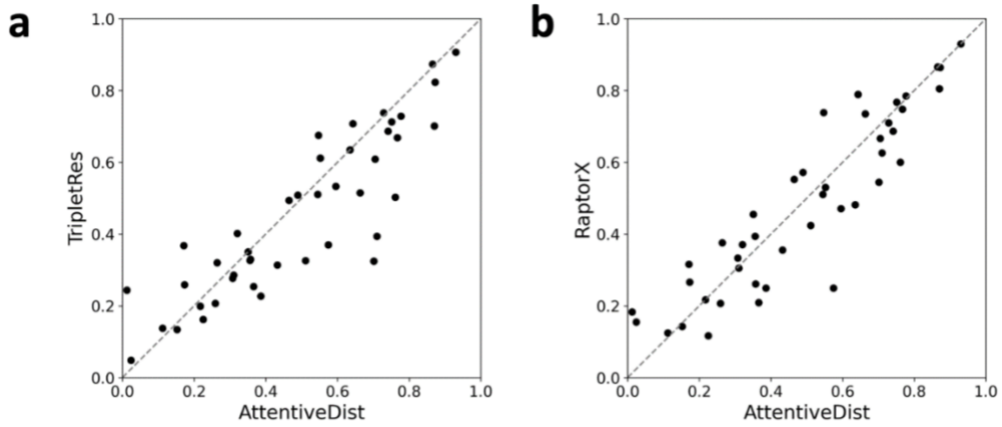


Figure 2.5. Long L/1 precision comparison of the 43 CASP13 FM and FM/TBM domains between a, TripletRes and AttentiveDist. AttentiveDist showed a higher L/1 precision than TripletRes for 27 domains and tied for 2 domains out of the 43 domains. b, Raptor-X and AttentiveDist. AttentiveDist showed a higher L/1 precision than Raptor-X for 23 domains and tied for 2 domains out of the 43 domains.

2.3.3 Prediction performance relative to the size of MSAs

As observed by previous works [14, 15], we also observed correlation between the size of MSAs, i.e. the number of sequences in the MSAs and the contact prediction accuracy. In Figure 2.6a, the L/1 long range contact precisions were shown for two methods, AttentiveDist and the model using only MSAs of E-value 0.001, relative to the number of sequences in the MSAs. The number of sequences in the MSAs is shown in the log scale. A positive correlation was observed, as shown in the figure, and particularly, there is clear distinction of the performance at the sequence count of 100. When the sequence count was less than 100, L/1 precision was always below 0.4. Oppositely, when the sequence count is very high, over 10,000, high precisions of over 0.75 were observed. Although the high precision was observed with a large number of sequences, observed precisions had a large range of values when the sequence counts was over 100.

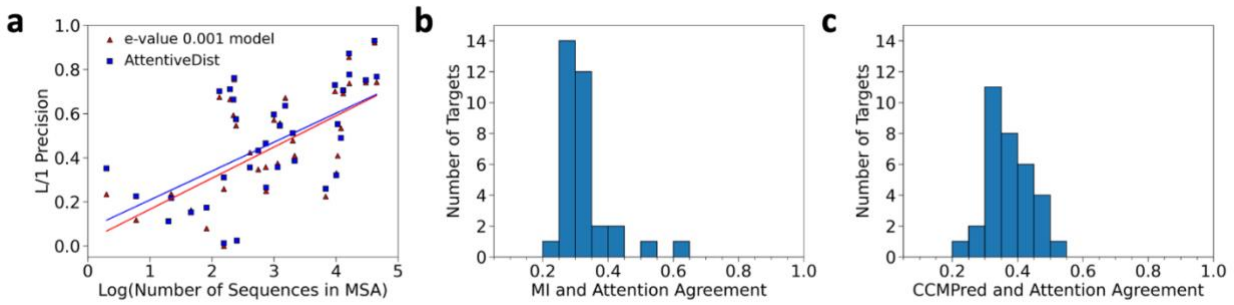


Figure 2.6. Analysis of the MSA size and the attention. a, Relationship between log of the sequence counts in MSAs and long-range L/1 contact precision for the 43 CASP13 targets. AttentionDist (blue) and the E-value 0.001 model (red), where E-value 0.001 was used as a cutoff for generating MSAs. The lines represent the regression. b, the fraction of residue pairs where the MSA with the highest attention agreed with the MSA with the highest mutual information (MI). The number of targets among the 35 CASP13 target proteins that have the particular fraction of agreed residue pairs were counted for each bin. 43 FM and FM/TBM CASP13 target domains belong to 35 proteins. Out of the 35 proteins, two proteins were discarded from this analysis because the four MSAs with different E-value cutoffs of these proteins were identical. c, the agreement is compared with the contact probability computed from the four MSAs with CCMPred.

2.3.4 Analyses of attention weights

In AttentiveDist, for each residue pair, attention values are distributed across four MSA-based features each computed with the four different E-value cutoffs, which sum up to 1.0. To understand what the attention mechanism captures, in Figure 2.6b and 2.6c we examined how the attention corresponds to co-evolution signals. We compared with local and global co-evolutionary signals. The local co-evolutionary signal used is mutual information (MI), which uses pairwise residue profile information. The global signal considers effects from other residues as well, which can be computed by pseudo-likelihood maximization (PLM). We used CCMPred [7], which is an implementation of PLM. For each residue pair in a protein target, we counted the number of times the MSA with the highest attention weight assigned by AttentiveDist agrees with the MSA with the highest co-evolutionary signal. As reference, we computed random-level agreement, where the MSA assignment for each residue pair was shuffled while keeping the fraction of times that each MSA had the highest attention weight in the original computation the same. The average agreement for MI was 0.329 compared to a random agreement of 0.298, and for CCMPred it was 0.376 compared to 0.277 random agreement. In both cases the agreement was higher than random. The histogram shifted to higher values when compared with CCMPred than MI. The average

agreement for the 33 proteins were higher for CCMPred than MI. Thus, overall, the attention is a mechanism to select MSAs with higher co-evolutionary signals.

We also analyzed attention weights assigned to each MSA features in targets. First, for each target, we summed attention values given to each MSA over all residue pairs in the target and selected the one with the highest sum as most informative. We found that out of 43 targets, in 32 targets E-value 0.001 received the most attention, while E-value 0.1, 1, and 10 received the most attention for 1, 1, and 8 targets, respectively.

Next, we analyzed attention values given to residue pairs in a target. Figure 2.7a, b, c, show the maximum, minimum, and standard deviation of attention weights given to four MSAs in each target. The average statistics for the CASP13 targets are shown in Table 2.3. We can observe that the attention weight values vary for different targets. Figure 2.7d shows the percentage of residue pairs that had the largest attention weight for each MSA feature. We can see that E-value 0.001 shared the largest fraction of residue pairs for most of the targets. This is understandable considering that E-value of 0.001 showed the highest prediction performance (Table 2.2) among the four MSA features.

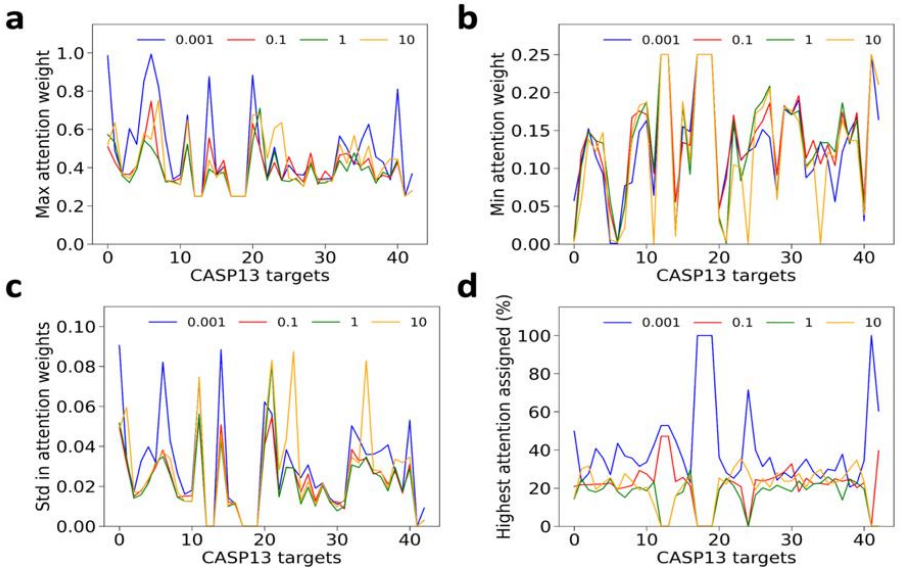


Figure 2.7. Statistics of pairwise attention weights given to the 43 CASP13 targets. a, the maximum attention weight given to each MSA among values for all the residue pairs. b, the minimum attention weight given to each MSA. c, standard deviation given to each MSA. d, Percentage of residue pairs in a target where each MSA had the largest attention weight. In all figures the x-axis represents the 43 CASP13 targets. Four MSAs with E-value of 0.001, 0.1, 1, and 10 are shown in blue, red, green, and yellow lines.

Table 2.3. Statistics of attention weights given to different E-value based features averaged over 43 CASP13 FM and FM/TBM domain targets.

MSA E-value	Max	Min	Std
0.001	0.486	0.129	0.030
0.1	0.397	0.141	0.022
1	0.382	0.138	0.021
10	0.424	0.124	0.027

2.3.5 Angle prediction

Accuracy of angle prediction are provided in Table 2.4. The results show the fraction of times that an angle is predicted at the exact correct bin or at a bin off by 1 or 2 bins. Within 2 bins, about 70% of the angles are predicted correctly.

Table 2.4. Accuracy of backbone phi-psi and orientation angles for the 43 CASP13 FM and FM/TBM domain targets. The bin size of torsional angles was set to 10° while the bin for the orientation angles was 15°. Bin slack of 0 represents that the predicted bin of the highest probability and the real bin were the same. Bin slack of 1(or 2) denotes that the predicted bin was 1(or 2) bin(s) away from the correct bin.

Angle	Bin Slack		
	0	1 (±10°)	2 (±20°)
φ	0.277	0.590	0.722
ϕ	0.242	0.548	0.700
	0	1 (±15°)	2 (±30°)
μ	0.332	0.582	0.634
θ	0.377	0.656	0.703
ρ	0.394	0.692	0.748

2.3.6 Protein structure modeling

Finally, we built the tertiary structure models of the CASP13 domains and compared with the top CASP13 server models. For the structure modeling, in addition to the predictions of Cβ-Cβ distance, main-chain φ, ψ angles, and the three, ρ, θ, and μ, side-chain orientation angles, we

tested the inclusion of two additional distance constraints, which were Side-chain CEnter (SCE)-SCE distances and peptide-bond nitrogen (N)-oxygen (O) atom distances. These distances help in proper secondary structure formation and side-chain packing. All the constraints were converted into a potential function by normalizing predicted probability values in bins by predicted reference probability values. The folding was performed using Rosetta [46] by adding the predicted potentials into the Rosetta energy function. Out of a few thousand models generated, the best scoring model for each target are reported in this section. Details are provided in Methods.

We compare the average TM scores of the predicted structures with three top CASP13 servers in Figure 2.8a. For AttentiveDist, we showed results by two versions, one with the predicted SCE-SCE distances and the backbone N-O distances, which is denoted as AttentiveDist (Full), and the one without these two distance constraints (AttentiveDist w/o SCE and N-O). AttentiveDist w/o SCE and N-O improved the TMscore from 0.552 to 0.568 compared to the single E-value 0.001 multi-task trained model, demonstrating the effectiveness of using four MSA's in structure modeling. Comparing two versions of AttentiveDist, the two distance constraints further improved the TMscore by 2.1% from 0.568 to 0.579. In Figure 2.9, TM-scores of individual domain targets by the two versions of AttentiveDist are shown. For 19 domains the multi-task AttentiveDist showed a higher GDT-TS and tied for 4 domains out of 43 domains in total.

AttentiveDist showed higher average TM scores than the top-three CASP13 severs, Zhang-Server (0.517), RaptorX-DeepModeller (0.508), and BAKER-ROSETTASERVER (0.450), which are shown in Figure 2.8a as well. As we used the same MSA extraction strategy as Zhang-Server, in Figure 2.8b, we further show the TMscores of the 43 individual targets by AttentiveDist (Full) and Zhang-Server. AttentiveDist (Full) showed a higher TM-Score than Zhang-Server for 29 cases and tied for 3 cases. We also compared the residue-residue contact area difference (CAD) score [65] in Table 2.5. CAD score determines the structure similarity by comparing the inter-atomic contact area between the reference and predicted structure. AttentiveDist improved both the AA (all residues) and SS (only sidechain residues) CAD score compared to the server models.

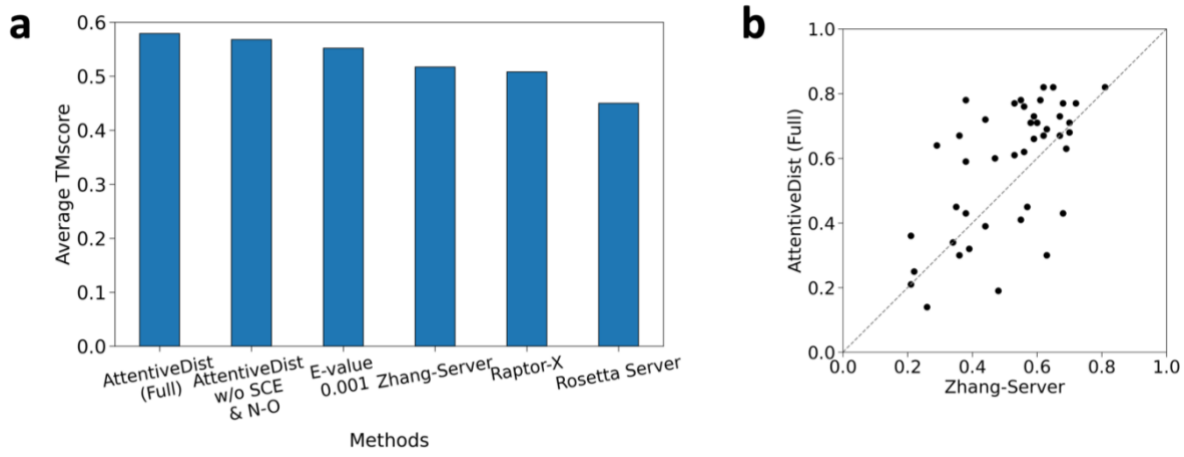


Figure 2.8. Performance in structure modelling. a, TM-score for AttentiveDist, AttentiveDist without using predicted sidechain center distance and backbone N-O distance and the top 3 server methods in CASP13 for 43 FM and FM/TBM targets. b, Individual target TM-score comparison between our method and the Zhang-Server. The registered name of Raptor-X in CASP13 was RaptorX-DeepModeller and BAKER-ROSETTASERVER for Rosetta Server.

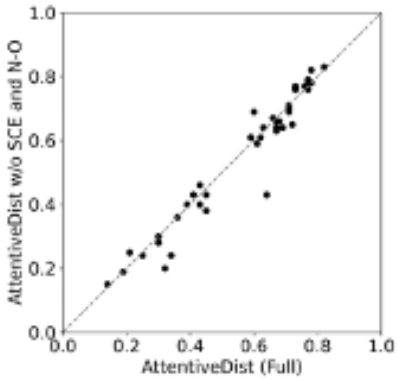


Figure 2.9. TM-score of AttentiveDist (Full) and AttentiveDist without using predicted SCE-SCE and N-O distances on the 43 CASP13 domains. AttentiveDist (Full) showed higher TM-Score for 19 targets, tied on 6 targets.

Table 2.5. Average CAD score of top 1 predicted PDB for the 43 CASP13 FM and FM/TBM domain targets. In AA all residue atoms are taken into consideration, while in SS only sidechain atoms are taken into consideration.

Model	CAD	
	AA	SS
Zhang-Server	0.512	0.237
RaptorX-DeepModeller	0.500	0.231
BAKER-ROSETTASERVER	0.509	0.225
AttentiveDist (Full)	**0.545**	**0.283**

Figure 2.10 provides four examples of models computed with distance prediction by AttentiveDist (Full) in comparison with Zhang-Server, RaptorX-DeepModeller, and BAKER-ROSETTASERVER. The first panel, Figure 2.10a, is a 180-residue long domain with two α-helices and two β-sheets, T0957s1-D1. While our model has a TM-score of 0.78, indicating that the overall conformation is almost correct, the models by the other three methods have some substantial differences from the native. The Zhang-Server model missed one β-sheet, the RaptorX-DeepModeller did not predict any β-sheets, and the BAKER-ROSETTASERVER placed the β-sheet at the top of the structure and a α-helix in substantially different orientations. The second example, T0980s1-D1 (Figure 2.10b**)** is another αβ class protein with a long loop region, which is placed on the right-hand side of the figures. The loop is difficult to correctly model, as the three top CASP13 servers did not fold it well. The incorrect modeling of the loop also affected to the placement of the α-helix in the right orientation in their models. Our AttentiveDist model managed to have the overall fold almost correct, as shown by a higher TM-score of 0.64. For the next target, T0986s2-D1 (Figure 2.10c), the Zhang-Server has almost all the architecture correct, but slight shifts of α helices cost it in the TM-score, which was 0.59. Our model had the conformation almost correct even in the loop regions, resulting in a high score of 0.78. The BAKER-ROSETTASERVER model did not assemble the large β-sheet correctly. The last target shown has an α-helical structure, which consists of two long α-helices with multiple small α-helices. (T0950-D1, Figure 2.10d). While our model identified correct orientations for the two long helices, the

other methods built them incorrectly which caused other incorrect helix arrangements at the top of the structure in the figure, resulting in lower scores.

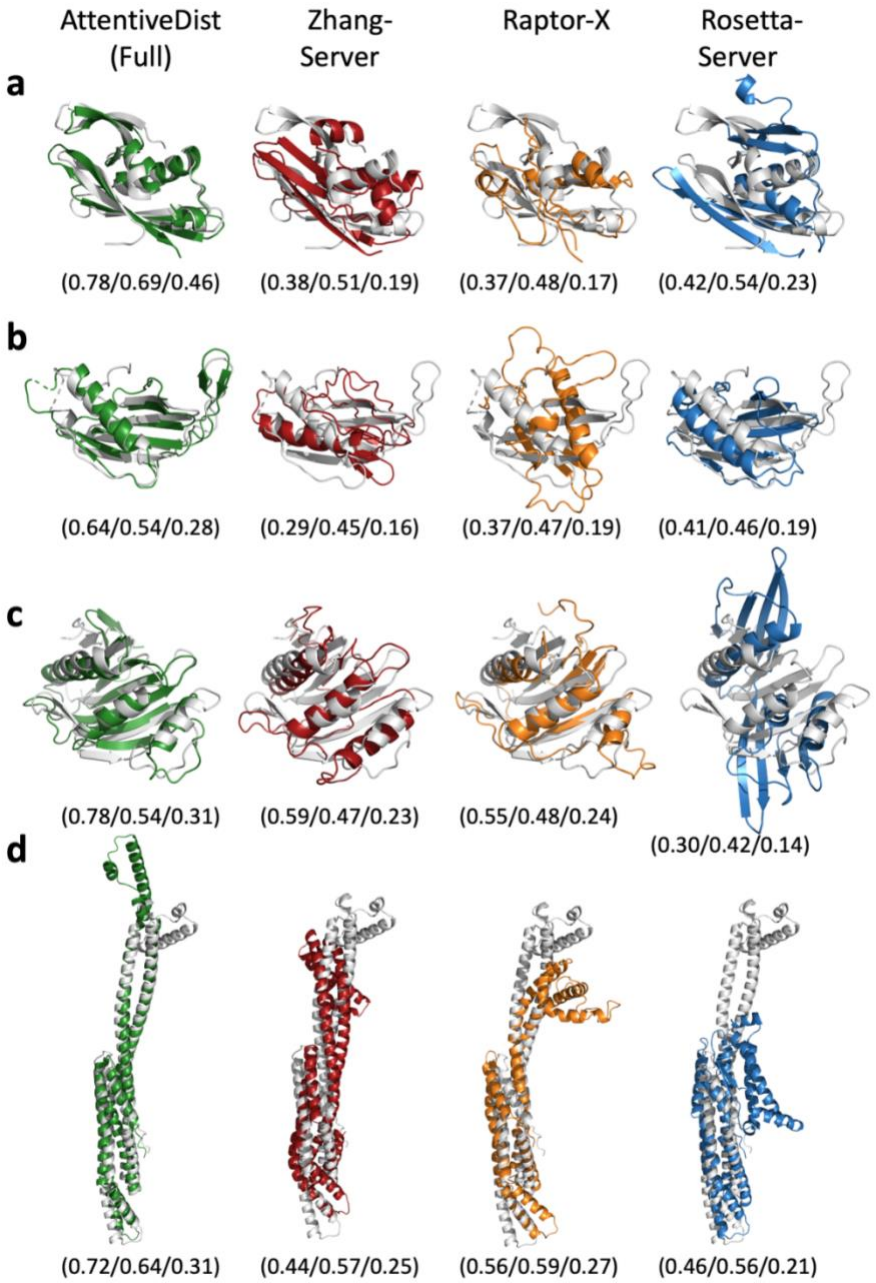


Figure 2.10. Examples of structure models by AttentiveDist (Full) in comparison with the top-1 model by the three top servers. AttentiveDist (Full), green; Zhang-Server, red; RaptorX-DeepModeller, orange; and BAKER-ROSETTASERVER, blue. The native structures are shown in gray. TM-scores, CAD AA, and CAD SS are shown in parentheses, respectively, separated by /. Targets are a, T0957s1-D1 (PDB ID: 6cp8; length: 180 amino acids); b, T0980s1-D1 (PDB ID: 6gnx; 104 aa); c, T0986s2-D1 (PDB ID: 6d7y; 155 aa); d, T0950-D1 (PDB ID: 6ek4; 331 aa).

2.3.7 Performance in CASP14

CASP, the worldwide protein structure modelling competition is an excellent platform to access the progress of structure prediction methods as well as the progress of the field in the community. We participated with AttentiveDist model in CASP14, with the results released in December 2020. For contact prediction category ranking of the models were computed based on L/5 long range contacts for free modelling targets. Our group was ranked 13th, with the server model ranked 32nd [66]. The model rank is lower because each group generally submits multiple variations of their model.

The L/5 long range precision by our group was 0.371 which is better than average but much lower that the best method achieving 0.665 precision. From analyzing the methods with higher performance than ours, there were two key reasons for this huge different in performance. First, we did not use the template model as input feature. A good template can provide significant amount of information about the protein fold leading to higher contact prediction precision. Second is the databases used for MSA search. The top groups used metagenomic databases like BFD [67], JGI [68] and MGnify [69] which contains millions to billions of protein sequences extracted from environmental genomics. A larger search database can increase the size of MSA, which as shown in Figure 2.6 leads to a better performance. Even without using metagenomics database and templates our model was still competitive for certain targets. For instance, for target T1029-D1 our server model prediction was ranked the best with TMscore 0.53, even higher than AlphaFold2 model having TMscore 0.47.

2.4 Discussion

We presented AttentiveDist, a deep learning-based method for predicting residue distances and angles from four MSAs with four different E-value cutoffs. By adding an attention layer to the network, useful features from MSAs were selectively extracted, which led to higher predictive performance. In AttentiveDist, the attention layer as well as multi-tasking strategy boosted the prediction accuracy. In the context of the recent intensive efforts for developing residue distance/contact prediction methods by the community, this work shows another strong demonstration of how protein structure information can be further squeezed by exploiting modern deep learning technologies. Although our approach showed higher precision for free modelling

targets, an improvement is still needed especially when the available sequences are sparse for input MSAs, which remains as an important future work.

2.5 Code availability

Code is made available at http://github.com/kiharalab/AttentiveDist.

CHAPTER 3. PROTEIN FUNCTION PREDICTION USING PHYLOGENETIC DISTANCE OF DISTANTLY RELATED SEQUENCES

Function annotation of proteins is fundamental in contemporary biology across fields including genomics, molecular biology, biochemistry, systems biology, and bioinformatics. Function prediction is indispensable in providing clues for interpreting omics-scale data as well as in assisting biologists to build hypotheses for designing experiments. As sequencing genomes is now routine due to the rapid advancement of sequencing technologies, computational protein function prediction methods have become increasingly important. A conventional method of annotating a protein sequence is to transfer functions from top hits of a homology search; however, this approach has substantial short comings including a low coverage in genome annotation. In this chapter I present Phylo-PFP, a new sequence-based protein function prediction method, which mines functional information from a broad range of similar sequences, including those with a low sequence similarity identified by a PSI-BLAST search. To evaluate functional similarity between identified sequences and the query protein more accurately, Phylo-PFP re-ranks retrieved sequences by considering their phylogenetic distance. Compared to the Phylo-PFP's predecessor, PFP, which was among the top ranked methods in the second round of the Critical Assessment of Functional Annotation (CAFA2), Phylo-PFP demonstrated substantial improvement in prediction accuracy.

3.1 Background

Proteins are drivers of almost all biological processes in the cell. Therefore, elucidating function of an individual protein is key to understanding how a biological system operates through functional interactions of component proteins. Ultimately, the biological function of a protein needs to be determined experimentally; however, a hypothesis is needed to design an assay that determines whether a target protein has a particular function. Computational function prediction can provide valuable information when biologists build such hypotheses. As genome sequencing has become routine due to the rapid advancement of sequencing technologies [70], function prediction has become increasingly important. Computational function prediction methods are also useful for analyzing omics data including gene expression and protein-protein interaction data.

In addition to function prediction methods that use protein sequence information, there are other types of methods that consider gene co-expression patterns, phylogenetic profiles, three dimensional (3D) structures of proteins, as well as protein-protein interaction networks [71]. These non-sequence-based methods can often identify functional relationships of proteins that are not obvious from sequence similarity. However, non-sequence information is not always available and thus has limited applicability.

Recently there is an increasing momentum for developing function prediction methods driven by successful organization of a community-wide objective assessment of protein function prediction, the Critical Assessment of Function Annotation (CAFA) [72, 73]. In CAFA, participants predict function (GO terms or other ontology terms specified by the organizers) of many target proteins (48,298, and 100,816 proteins in CAFA1 and CAFA2, respectively). Then, predictions are evaluated only for newly annotated GO terms to the target proteins after a waiting period of over six months from the prediction submission. This process is designed for assessing methods' capability of predicting new functions rather than retrieving known functions from existing data sources. Three rounds of CAFA have been held so far, CAFA1 in 2010-2011, CAFA2 in 2013-2014, and CAFA3 in 2016-2017, for which the official evaluations were reported for the first two.

PFP [38, 39] is one of the pioneer methods, which makes use of sequences with a wide range of similarity to a query ranging from significant hits to very weakly similar ones up to an E-value of 125, far larger than conventionally used thresholds, e.g. 0.001. GO terms are extracted from all the retrieved sequences; however, to reduce the risk of predicting unrelated GO terms taken from weakly similar sequences, sequences are weighted by their E-values. PFP also considers the co-occurrence of GO terms, which is statistics of GO term pairs that frequently co-occur in annotation of the same sequence. PFP was one of the top ranked function prediction methods in CAFA and the top in the Critical Assessment of Protein Structure Prediction (CASP) function prediction category in 2007 [41].

Here, we present a new method, Phylo-PFP, which significantly improves prediction performance over PFP by incorporating phylogenetic information in defining sequence similarity. We first show that the E-values of the sequences do not largely agree with the distances defined by phylogenetic trees to a surprising extent. Then, we show that weighting sequence by considering the phylogenetic distance can substantially improve GO term prediction accuracy.

Predictions by Phylo-PFP were evaluated on a dataset of 1702 non-redundant protein sequences and showed better performance than the original PFP as well as several other existing methods. To compare its performance among the best programs available to date, Phylo-PFP was used to predict functions of target sequences in CAFA2. We show that Phylo-PFP outperforms all the top methods used in CAFA2, having the highest score in all three GO categories, Molecular Function (MF), Biological Process (BP), and Cellular Component (CC).

3.2 Methods

3.2.1 Overview of the Phylo-PFP method

Figure 3.1 illustrates the workflow of Phylo-PFP. For a query protein sequence, Phylo-PFP searches similar sequences from a reference sequence database with PSI-BLAST (maximum iteration set to 3). In this retrieval, top 500 sequences are retrieved or until an E-value of up to 125 is reached. Collecting diverse sequences with a large E-value has two advantages: First, as demonstrated in the original version of PFP, the E-value cutoff will capture a larger breadth of sequences, which is particularly effective when closely annotated homologs to the query do not exist in the database. Also, for Phylo-PFP, having many sequences help in constructing meaningful phylogenetic trees, which is a key new feature of the Phylo-PFP algorithm.

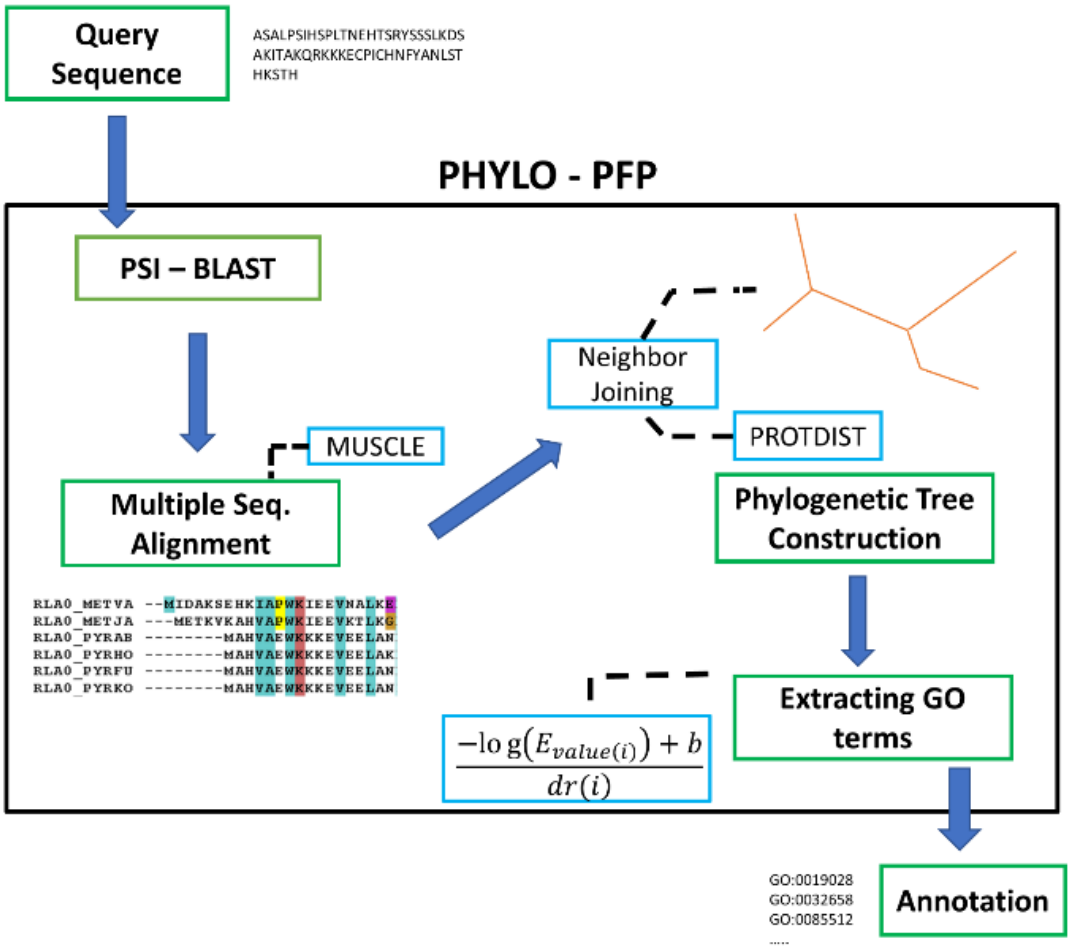


Figure 3.1. Overview of Phylo-PFP algorithm

The next step of Phylo-PFP is to rank retrieved sequences using a weighting factor that considers the phylogenetic distance among them. This step is the key difference from the original PFP, which simply uses the raw E-value to rank sequences. There are three steps in constructing a phylogenetic tree: 1) A multiple sequence alignment (MSA) is computed for the retrieved sequences using MUSCLE [74]. 2) From the MSA, a pairwise sequence alignment for each sequence pair is extracted, from which a distance matrix is computed using PROTDIST [75] in the PHYLIP package with the Jones-Taylor-Thornton model. 3) With the set of computed distances, a phylogenetic tree is constructed using the neighbor joining (NJ) method implemented in PHYLIP. Following the tree construction, a distance dr is defined between the query protein and each protein k as the sum of the branch lengths between them on the tree, which is scaled to a value between 0 and 100 as

$$dr = \frac{distance(k) - \min_i distance(i)}{\max_i distance(i) - \min_i distance(i)} * 100 \qquad (1)$$

Using the phylogenetic distance dr and the E-value, a retrieved sequence *i* is ranked with a weight named the Evolutionary distance-normalized Log E-value (ELE) in the descending order:

$$ELE(i) = \frac{-log_{10}(E-value(i)) + b}{dr(i)} \qquad (2)$$

where *E-value(i)* is the E-value of the sequence *i*, *b* is the constant, $\log_{10}(125)$, which is an offset added to make the numerator of the equation a non-negative value up to an E-value of 125, and $dr(i)$ is the phylogenetic distance of the sequence *i*. The numerator is the weight used in the original PFP. In Phylo-PFP, the numerator is normalized by $dr(i)$, i.e. a sequence that has a large distance on the phylogenetic tree receives a discounted weight, which brings the contribution of the sequence lower when the score for predicted GO terms are computed. Using ELE, a function (GO term) f_a is scored for a query sequence as

$$s(f_a) = \sum_{i=1}^{N} \sum_{j=1}^{Nfunc(i)} \big(ELE(i) P(f_a|f_j)\big) \qquad (3)$$

where *N* is the number of sequences retrieved from the sequence database within an E-value of 125, *Nfunc(i)* is he number of GO terms annotating the sequence *i*, *ELE(i)* is the weight defined in Eq. 2, and $P(f_a|f_j)$ is the functional association [39], a conditional probability that GO term f_a is in annotation of a sequence that is also annotated with GO term f_j. The function association allows predicting GO terms that do not appear in annotations of retrieved sequences. Associations are also computed between terms across different categories, e.g. terms in MF and BP. Associations with a probability of 0.9 or higher were considered. Each GO term in the final prediction is also given a confidence score, which is computed by normalizing ELE for all GO terms belonging to the same category. The Eq. 2 is an update from the original PFP score [39]. In PFP, the score is

$$s(f_a) = \sum_{i=1}^{N} \sum_{j=1}^{Nfunc(i)} ((-\log(E\text{-value(i)}) + b) P(f_a|f_j) \qquad (4)$$

where *E-value(i)* is the E-value of sequence *i*.

3.2.2 Constructing the annotation database

For any function prediction method, it is crucial to have a comprehensive annotation database that keeps known GO terms for sequences, as the method depends on it in extracting GO terms from PSI-BLAST hits. We integrated several data sources to form our annotation database for Phylo-

PFP. The primary database used was the UniProtKB/Swiss-Prot including Non-IEA (Inferred from Electronic Annotation) annotations [76]. In addition we integrated annotations from UniPathway [77], TIGRFAMs [78], SMART [79], Reactome [80], PROSITE [81], ProDom [82], PRINTS [83], PIRSF [84], Pfam [85], InterPro [86], and HAMAP [87].

3.2.3 Non-redundant benchmark dataset

Target sequences for the benchmark dataset were selected from UniProt Reference Clusters (UniRef), which provides a clustered set of sequences from UniProt Knowledgebase [88]. We used the UniRef50 clusters of 8/25/2016, in which sequences with more than 50% identity to each other are clustered. We selected a representative sequence from each cluster which fulfills two conditions: a cluster must include more than 1500 sequences, and the representative protein is annotated in UniProt. Representative sequences were removed if it had more than 500 hits with an E-value 0.0 in the third round of PSI-BLAST as these sequences have many highly similar sequences which makes their function prediction easy. This procedure yielded 1702 sequences for the benchmark dataset. We also constructed another benchmark dataset by clustering these 1702 sequences with 30% sequence identity cutoff.

3.2.4 CAFA2 dataset

We also tested Phylo-PFP on the dataset from CAFA2 [73]. CAFA2 released 100,816 target protein sequences but predictions were evaluated only for 1776 sequences which newly accumulated GO terms during the waiting period. Among the 1776 sequences, 419 sequences had MF GO terms, 860 sequences had BP GO terms, and 1259 sequences had CC GO terms. For replicating participation in CAFA2 with Phylo-PFP, the benchmark sequence dataset as well as the ground truth of the annotation were obtained from the supplementary data at https://figshare.com/articles/Supplementary_Data_for_CAFA2/2059944/1. When we ran Phylo-PFP, we used the UniProt database of August 2013 (a version released before the CAFA2 target sequences were released to participants), so that annotations newly added after the release were not included.

3.2.5 Other methods compared

For PSI-BLAST, we extracted GO terms from the top 10 hits in the third iteration of a PSI-BLAST run. As for Pfam [89], GO terms were extracted from profile hits using Pfam2go mapping available from http://www.geneontology.org. For both PSI-BLAST and Pfam a confidence level of a predicted GO term was assigned by the E-value of the most significant sequence hit from which that term was extracted (in case multiple sequence hits have the same GO term): A confidence score of 1.0 was assigned if the E-value of the sequence was 0.01 or smaller; 0.5 if the E-value was between 0.01 to 10; and 0.2 when the E-value was 10 or larger. As for SIFTER [90], we used the webserver at http://sifter.berkeley.edu.

3.2.6 Prediction evaluation score

For PSI-BLAST, we extracted Prediction accuracy was evaluated with the F_{max} score following the evaluation in evaluation in CAFA. For each protein sequence, it compares a set of GO terms predicted by a method with the true annotation of the protein and calculates precision P, recall R and F_{max} as

$$P(t) = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_i(t)}{TP_i(t)+FP_i(t)} \quad (5)$$

$$R(t) = \frac{1}{n}\sum_{i=1}^{n}\frac{TP_i(t)}{TP_i(t)+FN_i(t)} \quad (6)$$

$$F_{max} = \max_t\{\text{F1}-\text{score}\} = \max_t\left\{\frac{2*P(t)*R(t)}{P(t)+R(t)}\right\} \quad (7)$$

where t is the score threshold, $P(t)$ is the precision at threshold t, $R(t)$ is the recall at threshold t, $TP_i(t)$ is the total number of GO terms that have predicted score greater than or equal to t and are present in true annotation set for protein i (i.e. true positive prediction considering that predicted GO terms with a score t or higher are predicted), $FP_i(t)$ is the total number of GO terms that have a predicted score greater than or equal to t and are not present in true annotation set for protein i (false positive at score t), $FN_i(t)$ is the total number of proteins that are present in true annotation set but do not have predicted score greater than or equal to t for protein i (false negative at score t), n is the number of proteins predicted used for evaluation. Considering that each method may assign its prediction confidence score of a different distribution, F1-score is computed at different confidence score cutoff, t, and the maximum of among the computed F1-score was taken as the prediction accuracy for the method (Eq. 7).

3.2.7 FunSim score

The funSim [91, 92] score was used to compute similarity of GO annotations of proteins. For GO annotations of two proteins, funSim is defined from similarity of GO term pairs as follows:

The similarity of two individual GO terms c_1 and c_2 is

$$sim(c_1, c_2) = \max_{c \in S(c_1,c_2)} \left(\frac{2 \cdot logp(c)}{logp(c_1) + logp(c_2)} \cdot (1 - p(c)) \right) \qquad (8)$$

where $p(c)$ is the annotation frequency of term c relative to the frequency of the ontology root, and $S(c_1, c_2)$ is the set of common ancestor terms between terms c_1 and c_2. The similarity of two sets of terms, GO_i^A and GO_j^B, of respective sizes *N* and *M* is calculated by constructing an all-by-all similarity matrix s_{ij}.

$$s_{ij} = sim\left(GO_i^A, GO_j^B\right), \forall i \epsilon \{1, \dots, N\}, \forall j \epsilon \{1, \dots, M\} \qquad (9)$$

Row vectors compare the similarity of set *A* (protein 1) to set *B* (protein 2), while column vectors compare the similarity of set *B* (protein 2) to set *A* (protein 1).

$$Sim(A, B) = \frac{1}{N} \sum_{i=1}^{N} \max_{1 \le i \le M} s_{ij} \qquad (10)$$

$$Sim(B, A) = \frac{1}{M} \sum_{j=1}^{M} \max_{1 \le j \le N} s_{ij} \qquad (11)$$

To calculate an overall similarity score for the two term sets, we combined these two terms for each GO category:

$$GO_{score} = max\ (\ Sim(A, B),\ Sim(B, A)\ ) \qquad (12)$$

where GO_{score} is any of the three category scores (*MF-score*, *BP-score*, *CC-score*). If annotations of the two query proteins contain terms for all three categories, funSim is defined as

$$funSim = \frac{1}{3}\left[\left(\frac{MF_{score}}{max(MF_{score})} \right)^2 + \left(\frac{BP_{score}}{max(BP_{score})} \right)^2 + \left(\frac{CC_{score}}{max(CC_{score})} \right)^2 \right] \qquad (13)$$

$max(GO_{score}) = 1$ (maximum possible GO_{score}) and the range of the *funSim* score is [0,1]. If query protein(s) do not have GO annotations for all three categories, funSim is computed only for categories that commonly exist in the two proteins compared.

3.3 Results

We first discuss that relationship between the E-value and phylogenetic distance of sequences. Then, we present prediction results of Phylo-PFP on the two datasets.

3.3.1 New sequence weight and sequence similarity

First, we examined to what extent the new sequence weight ELE (Eq. 2) correlates sequence similarity score computed with E-value used in the original PFP. In phylogenetic studies, difference between sequence similarity scores and the phylogenetic distance has been a focus of interest [93, 94]. Smith and Pease discussed cases when a sequence similarity score, *-log(E-value)*, which is also used in the original PFP, does not capture evolutionary related sequences [94]. Eisen showed examples when the phylogenetic distance is expected to perform better than a sequence similarity-based score in predicting gene function [95].

Figure 3.2a shows the distribution of the Pearson's correlation coefficients between ELE and *-log(E-value)* for PSI-BLAST hits for 1702 sequences in the benchmark dataset. For each query sequence in the benchmark dataset, similar sequences to the query were retrieved from the database with PSI-BLAST up to an E-value of 125, and correlation between the sequence similarity score and ELE was computed and summarized in a histogram. For cases of very similar sequences to the query with an E-value of 0, a very small number (1e-1000) was assigned.

Although there is a small peak at the highest correlation bin of 1.0, the highest peak in the plot was observed at a very weak correlation of around 0.1. 53.58% were less than 0.2. Due to these very weak correlation, the mean correlations values were modest, 0.234. The same trend was

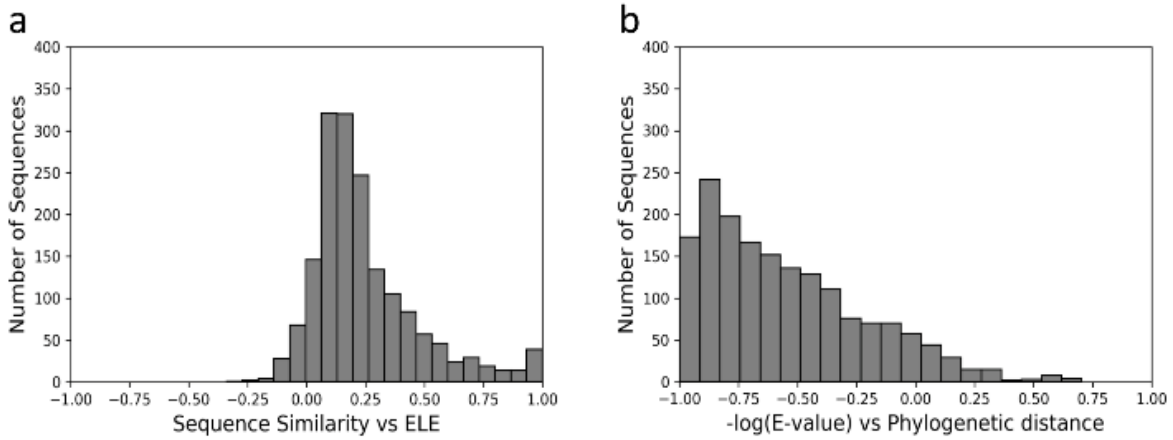


Figure 3.2. (a) Histogram of Pearson's correlation coefficients computed between -log(E-value) and ELE of PSI-BLAST hits for the dataset of 1702 sequences. (b) Histogram of Pearson's correlation coefficients between -log(E-value) and the phylogenetic distance.

observed in Figure 3.3, which is a histogram of correlation between the BLAST Bit score and ELE. Smith & Pease showed similar plots of correlation between the evolutionary distance and *–log(E-*

value) for simulated protein sequences [94], which corresponds to Figure 3.2b in our analysis. Compared with their results (Fig. 3A and B in their paper), which showed high correlations between the two values, our results on real sequences show a diverse distribution of correlations including many cases that had almost no correlation. The average correlation in Figure 3.2b was -0.546. The different result between the plots by Smith & Pease and Figure 3.2b in this work is probably due to the different ways that the sequence datasets were constructed. The sequence dataset in the former work was simulated based on a molecular model [94] while in the current work sequences were collected from real database searches up to a very weak similarity of an E-value of 125. Another difference is that while the plots by Smith & Pease are computed only for two query proteins, the current work summaries 1702 proteins showing that there are sequences of a high correlation but with a larger number of sequences with weak correlations.

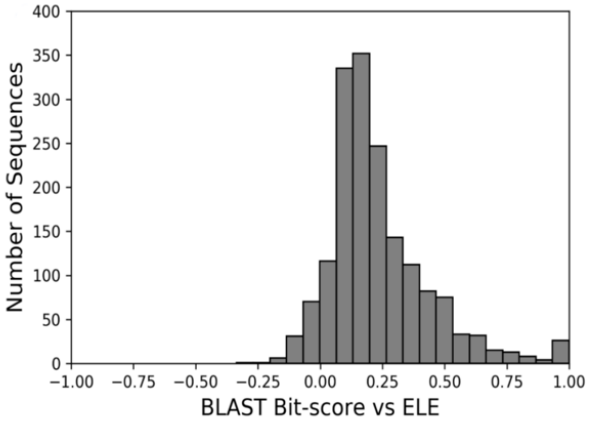


Figure 3.3. Correlation between BLAST Bit score and ELE. Histogram of Pearson's correlation coefficients between the BLAST Bit alignment score and the ELE score of PSI-BLAST hits for the 1702 sequences in the benchmark dataset. The average correlation was 0.247 and 53.70% of the correlation values were less than 0.2.

In Figure 3.4, three examples of proteins with a high and low correlations are shown. Figure 4a is from alpha-ketoglutarate-dependent dioxygenase AlkB (UniProt ID: P05050), an example of sequences with a high correlation between ELE and the sequence similarity, -log(E-value). The correlation of this protein was 0.999. Figure 4b and Figure 4c are opposite cases where there was no correlation between ELE and the sequence similarity. Figure 3.4b is from alpha-ketoglutaric semialdehyde dehydrogenase (UniProt ID: Q6FFQ0). The correlation was 0.125. For this protein, the score distribution was split into a large number of high scoring proteins (upper right) and a

small number of low scoring proteins (bottom left) and the high scoring proteins were a mixture of different dehydrogenase families dominated by N-succinylglutamate 5-semialdehyde dehydrogenase and NAD/NADP-dependent betaine aldehyde dehydrogenase, which had high similarity but inconsistent score rankings between ELE and the sequence similarity. ELE was higher for semialdehyde proteins and lower for dehydrogenases, while the sequence similarity was almost the same among them. Figure 3.4c is a plot for peptide chain release factor 2 (UniProt ID: Q8ZHK4), which had a correlation value of 0.179. Sequence hits consisted of peptide chain release factor 1 and peptide chain release factor 2 from different organisms. As shown in the figure, ELE distinguished factor 1 (triangles) and factor 2 (circles) better than the sequence similarity giving higher scores to factor 2 homologs, while the sequence similarity did not separate these two groups.

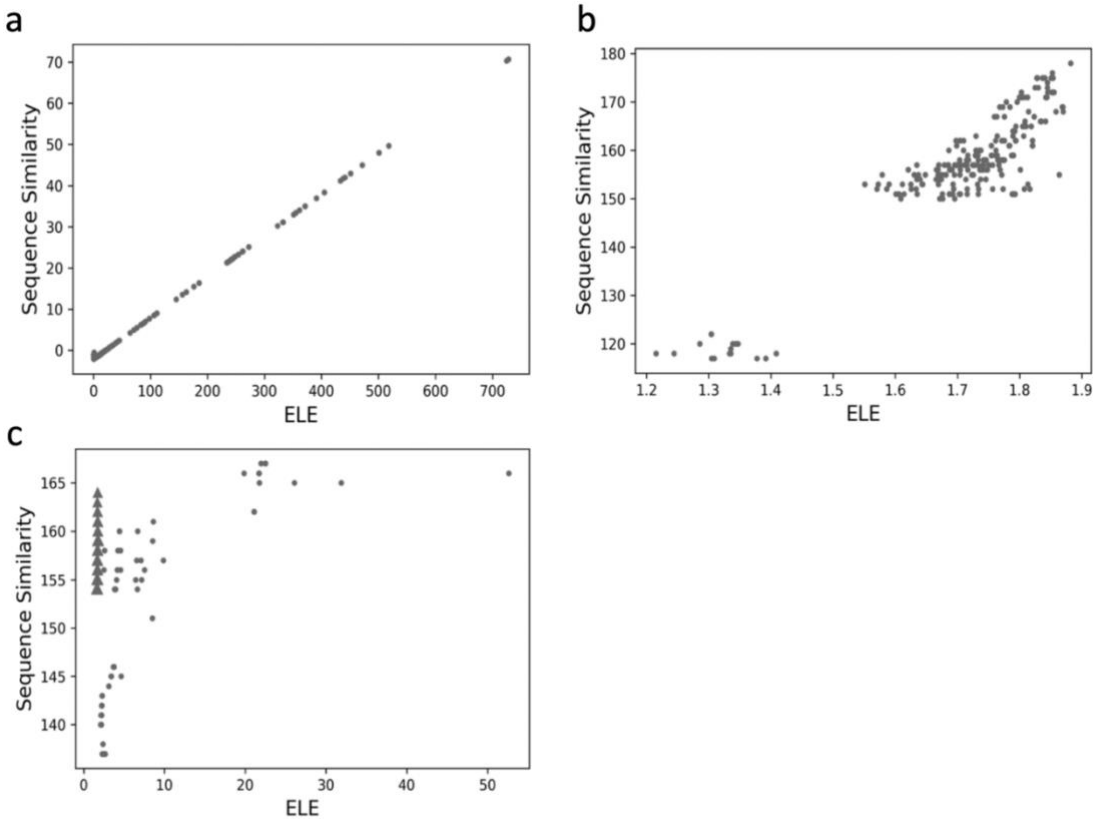


Figure 3.4. Examples of score correlation of individual proteins. (a) Score distribution of ELE and the sequence identity for alpha-ketoglutarate-dependent dioxygenase AlkB (UniProt ID: P05050). (b) Score distribution of ELE and the sequence identity for alpha-ketoglutaric semialdehyde dehydrogenase (UniProt ID: Q6FFQ0). (c) Score distribution of ELE and the sequence identity for peptide chain release factor 2 (UniProt ID: Q8ZHK4). The sequence hits include factor 2 homologs (circles) and factor 1 homologs (triangles).

3.3.2 Performance of Phylo-PFP on the non-redundant benchmark dataset

Next, we evaluated prediction performance of Phylo-PFP on the benchmark dataset. The prediction performance was compared with the original PFP and three other existing methods as reference, PSI-BLAST, Pfam, and SIFTER [90]. SIFTER was chosen because it considers a phylogenetic tree to transfer function from similar proteins to the query. When we ran Phylo-PFP and PFP, we removed the query sequence itself and sequence hits with an E-value of 0 from the PSI-BLAST run. The performance of the methods were evaluated with the Fmax score, which is the average F1-score at a method's score cutoff that gives the maximum F1-score to the entire set of target proteins (i.e. the method's score cutoff was not optimized differently for each target. Fmax score was used because it is a main evaluation metric used in CAFA.

Table 3.1 summarizes the Fmax score of the five methods. Phylo-PFP showed the highest Fmax score, 0.812 followed by PSI-BLAST with a score of 0.785. The rest of the methods were ranked in the order of PFP, Pfam, and SIFTER. To test the statistical significance, we ran a two-sided hypothesis test for each method against Phylo-PFP, using paired t-test. In Table 3.2 P-value is shown from hypothesis testing of Fmax score of each method compared to Phylo-PFP, showing that the performance difference between Phylo-PFP and the other methods was statistically significant. For further comparison, in Figure 3.5A we removed sequence hits up to a certain E-value, 1e-2, 1e-1, 1, 10, and 100 from the PSI-BLAST search for the three methods, Phylo-PFP, PFP, and PSI-BLAST, and predicted GO terms from remaining sequence hits. This is to simulate situations when a query protein does not find any significant hits. Pfam and SIFTER results do not change by E-value cutoffs, because PSI-BLAST is not used in their algorithms. It is apparent that Phylo-PFP and PFP performed substantially better than PSI-BLAST. At an E-value cutoff of 1e-2, the Fmax scores of the three methods were 0.465, 0.463, and 0.353, respectively (Table 3.1). It caught our attention that the Fmax score of PFP was worse than PSI-BLAST with no cutoff, which is probably due to the nature of this particular benchmark dataset, where query sequences have a sufficient number of highly similar sequences because they are collected from clusters of Uniref50. However, when sequence hits were limited to an E-value of 1e-2 or lower, PFP showed its superior ability to PSI-BLAST as consistent with the earlier benchmark studies of PFP [38, 39, 96]. Comparing Phylo-PFP and PFP, Phylo-PFP performed better with no cutoff (0 in the plot) and cutoffs of 1e-2, 1e-1, 1, and 10. The margin between the two methods was largest when no E-value cutoff was applied. This implies that the sequence hits reranking with the ELE weight was more

effective when closely similar sequences were more correctly ranked. At the cutoff of 100, Phylo-PFP and PFP showed almost identical Fmax score, 0.400, and 0.401, respectively.

Table 3.1.The Fmax score of the five methods on the benchmark dataset. Fmax scores were computed at two E-value cutoffs of PSI-BLAST search, with no cutoff and 1e-2. Only one score was provided for Pfam and SIFTER since they do not use a database search results from PSI-BLAST.

Method	Fmax (no cutoff)	Fmax 1e-2 cutoff
Phylo-PFP	0.812	0.465
PFP	0.747	0.463
PSI-BLAST	0.785	0.353
PFam	0.500	-
SIFTER	0.288	-

Table 3.2. Statistical test for the results shown in Table 3.1.

Method	P-value
PFP	4.83e-88
PSI-BLAST	6.38e-19
Pfam	1.42e-226
SIFTER	0.0

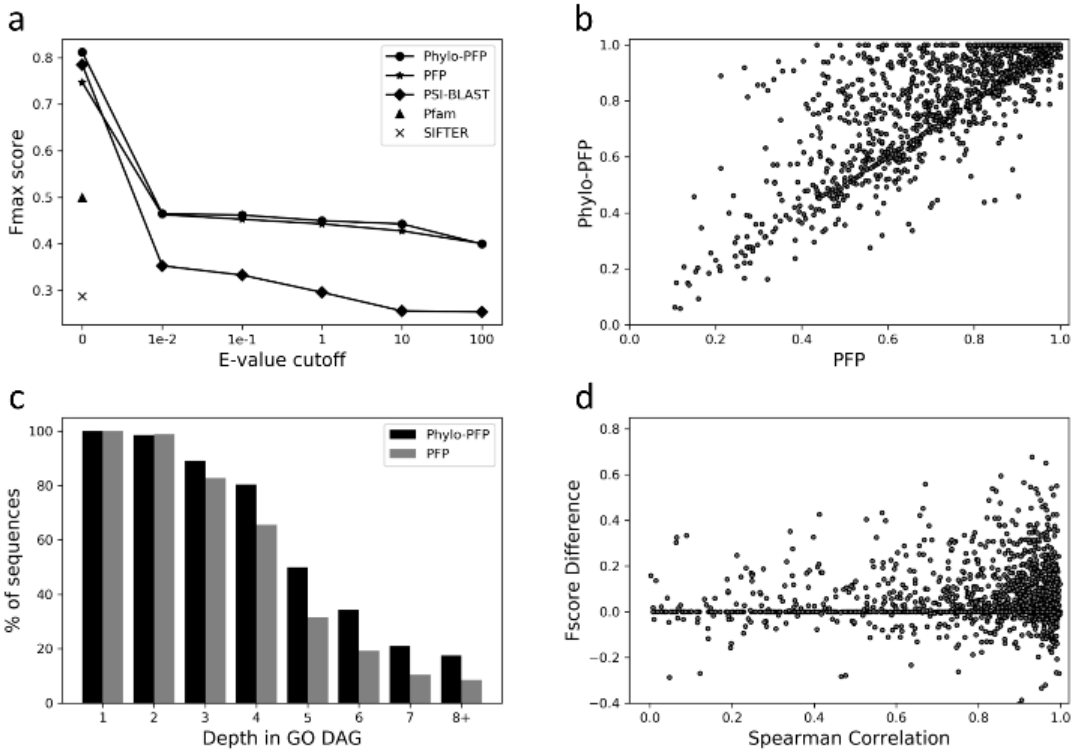


Figure 3.5. Prediction performance of Phylo-PFP on the benchmark dataset of 1702 non-redundant proteins. (**a**) Performance comparison with different E-value cutoffs applied to PSI-BLAST hits in terms of the Fmax score. Phylo-PFP (circles) was compared with PFP (stars), PSI-BLAST (diamonds), Pfam (triangle), and SIFTER (cross). Sequence hits that have an E-value smaller (i.e. more significant) than the E-value cutoff are removed and not used for extracting GO terms. (**b**) Comparison of predictions by Phylo-PFP and PFP for individual proteins. Fmax scores were compared. (**c**) The depth of correctly predicted GO terms with an E-value cutoff of 1e-2 by Phylo-PFP and PFP were compared. The x-axis represents the depth of the correctly predicted GO terms in the GO graph. If a GO term has multiple parental terms with different depths, the smallest depth for the term was considered. Predictions with a confidence score of 0.9 or higher were considered. If a sequence had multiple correctly predicted GO terms of different depths, the sequence was counted for all the depths. The right most bars, 8+, are for depths of 8 or larger. (**d**) Difference of Fmax scores of Phylo-PFP and PFP against the Spearman's correlation between the PSI-BLAST hits ranks of the two methods. Each data point corresponds to a protein sequence in the benchmark dataset.

As described in Methods, Phylo-PFP uses E-value of 125 as a sequence retrieval cutoff for PSI-BLAST if the number of sequences does not reach 500 before that E-value. We compared the performance of using E-value cutoff of 100 and 150 as well. In the benchmark dataset created from UniRef50, sequences retrieved from 78 of the 1702 query sequences had an E-value of 150 or larger within top 500 sequence hits of PSI-BLAST. Thus, we compared on these 78 sequences. The Table 3.3 shows the Fmax score of Phylo-PFP using three E-value cutoffs, 100,125 and 150,

for those sequences showing that 100 and 150 gave similar results but 125 had the highest Fmax score among them.

Additionally, we also used HHblits [97] and MMseqs2 [98] instead of PSI-BLAST in Phylo-PFP with the same parameters as we used PSI-BLAST, i.e. up to three iterations and an E-value cutoff of 125 with 500 maximum sequence hits. Interestingly, Phylo-PFP with MMseqs2 exceeded the Phlyo-PFP's performance with a Fmax of 0.842. Comparison of the two methods for each benchmark sequence are shown in Figure 3.6. Phylo-PFP-HHblits had an Fmax score of 0.633.

We further tested the methods Uniref50 dataset clustered with 30% identity cutoff, which included 1234 sequences. The results are shown in Table 3.4 and were consistent with results of 50% identify dataset.

In the subsequent panels in Figure 3.5, we analyzed the difference between Phylo-PFP and PFP from several different angles. Figure 3.5b shows a direct comparison of Fmax score of individual proteins in the benchmark dataset. Phylo-PFP showed larger or the same Fmax score than PFP for 83.72% of the sequences. Often the gain by Phylo-PFP over PFP was large; for 89 (5.23%) sequences the improvement of the score was more than 0.3 and the maximum Fmax score increase observed was 0.677 (from0.212 to 0.889). Phylo-PFP achieved the perfect score of 1.0 for 529 proteins while it was 338 for PFP. On the other hand, the deterioration of the score by Phylo-PFP was relatively small. For only 5 (0.29%) sequences the decrease in the score was more than 0.3.

Table 3.3.Phylo-PFP results with different E-value cutoff

E-value cutoff	Fmax
100	0.773
125	0.775
150	0.770

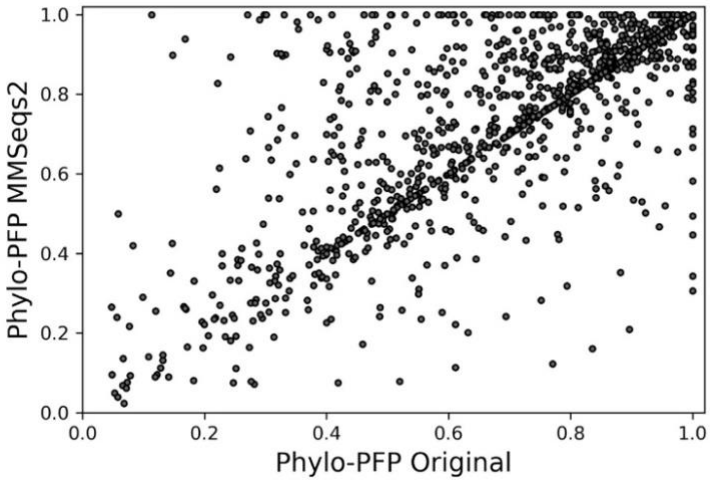


Figure 3.6. Comparison between original Phylo-PFP with Phylo-PFP-MMSeq2. The above plot compares the F-score for each sequence in the UniRef50 benchmark dataset between the two methods. Phylo-PFP with MMSeq2 showed a higher score than the original Phylo-PFP for 580 sequences, while original Phylo-PFP was better for other 303 sequences. The two methods showed the same score for the rest.

Table 3.4.Fmax score of Phylo-PFP and other methods on the benchmark dataset with 30% identity cutoff.

Method	50% similarity Fmax	30% similarity Fmax
Phylo-PFP	0.812	0.803
Phylo-PFP MMSeqs2	0.842	0.828
Phylo-PFP HHBlits	0.633	0.625
PFP	0.747	0.736
PSI-BLAST	0.785	0.773
Pfam	0.500	0.483
SIFTER	0.288	0.284

In Figure 3.5c, we examined the information content of predicted functions by Phylo-PFP and PFP quantified as the depth of correctly predicted GO terms. GO terms are organized in a directed acyclic graph ordered from general functional terms to more specific functions [99]. Thus, correct predictions of GO terms at larger depth (closer to leaves) are more valuable than prediction of shallower GO terms. In the plot, the results from the E-value cutoff of 1e-2 (Figure 3.5a) was used and only high confidence predictions with a confidence level over 0.9 were considered. It is shown in Figure 3.5c that Phylo-PFP predicted more terms at larger depths than PFP. Phylo-PFP predicted correct GO terms at a depth of five or deeper for 1.58 times more sequences than PFP. When only depths of eight or deeper were considered, Phylo-PFP predicted GO terms in 2.03 times as many cases as PFP.

In the last panel, Figure 3.5d, we examined the difference of the prediction performance (Fmax score) for each target protein between Phylo-PFP and PFP relative to the amount of the difference in the ranks of PSI-BLAST- retrieved sequences. Since Phylo-PFP and PFP use the same set of retrieved sequences from a PSI-BLAST search with only difference being ranking of the sequences due to the different scoring schemes used by the two methods, the performance difference may be correlated to the difference of the sequence ranks. The difference of the sequence rankings was evaluated by the Spearman's correlation (x-axis). We expected that a large improvement of prediction accuracy occurs when a large sequence ranking difference is observed, which should result in a small correlation coefficient. However, the trend seems to be rather opposite. Large Fmax score improvements were observed more frequently when the correlation values are close to 1.0, which indicates a small difference in the sequence rankings of the two methods. This may be implying that an improvement occurs when a small number of key sequences are adjusted in their ranks.

3.3.3 Permutation test of sequence ranking

We further tested statistical significance of rank change by Phylo-PFP. For each sequence in the benchmark dataset, we examined sequence hits from its PSI-BLAST run and marked "correct" sequences (that has a FunSim score of 0.6 or higher). Then, we checked the rank of all the correct sequences in the sequence rank based on ELE for Phylo-PFP and the E-value score for PFP. Ideally, correct sequences should go up in the ranking by ELE, as we showed in examples in Figure 3.7 and Figure 3.8. We performed Mann–Whitney U-test as the test statistic between the two sets

of ranks of E-value (in PSI-BLAST) and ELE (in Phylo-PFP). We then randomly shuffled the ranks 1000 times and performed the same test to obtain the distribution of the test statistic and calculated the p-value.

Out of 1702 benchmark sequences, 232 (13.6%) had a statistically significant rank change (p-value <0.05). This result is consistent with our initial observation that the changes in the ranking seemed to be not huge, and only key correct sequences move up in the rank. It should also be noted that the ranking change does not directly correlate with the improvement of function prediction accuracy, since Phylo-PFP uses a different scoring scheme from the original PFP.

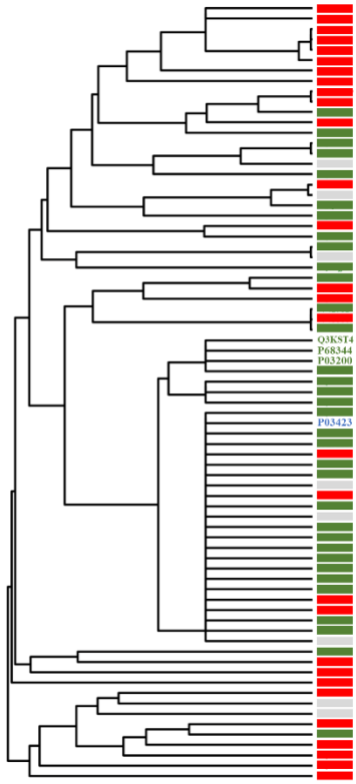


Figure 3.7. Visualization of sequence rank changes by ELE used in Phylo-PFP relative to the functional similarity to the query protein. The dendrogram shows functional similarity of each sequence to the query protein (shown in blue), which was quantified with the funSim score of GO terms annotations of two proteins. Top 75 sequences of highest functional similarity to a query protein are shown. In comparison with sequence hit ranks in PFP, sequences that went up or down in their ranking in Phylo-PFP are shown in green and red, respectively. UniProt IDs are shown for proteins that are mentioned in the text. The query is human major surface glycoprotein G (UniProt ID: P03423).

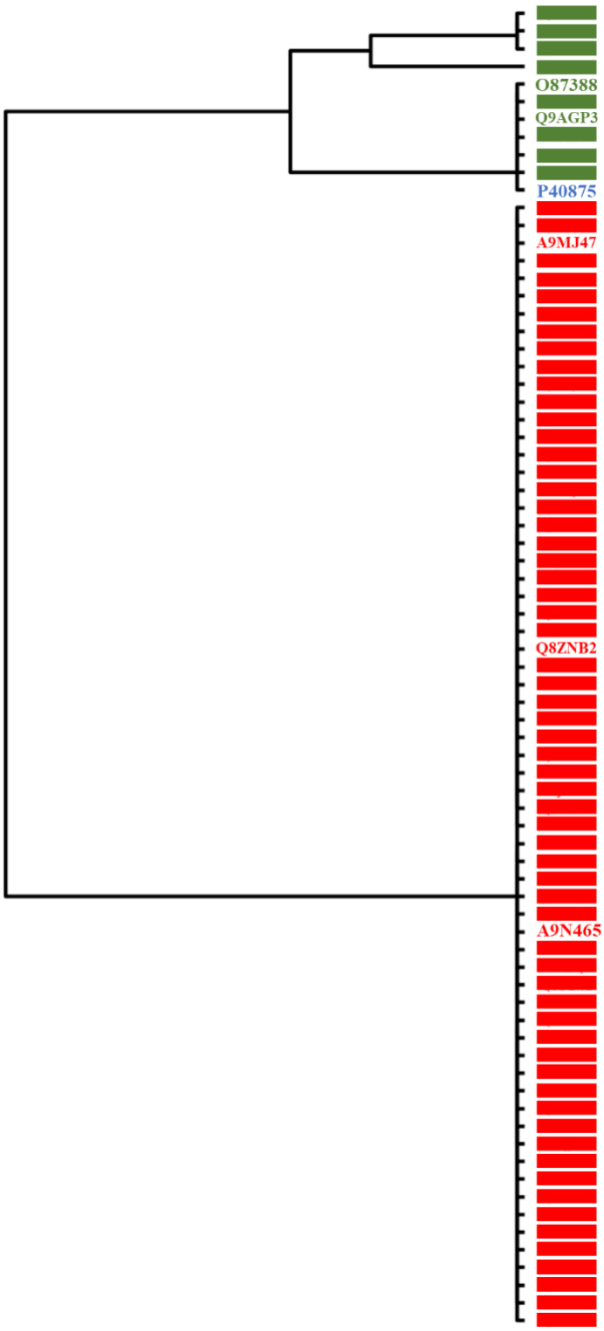


Figure 3.8. Visualization of sequence rank changes by ELE used in Phylo-PFP relative to the functional similarity to the query protein, sarcosine oxidase subunit beta from Corynebacterium sp. (UniProt: P40875). Sequence hits with their ranks moved up are shown in green, whereas sequences with lowered rank are shown in red. The query protein is show in blue.

3.3.4 Case Studies

In this section we discuss an illustrative case of Phylo-PFP's prediction. The focus is to examine how Phylo-PFP improved prediction over PFP by reranking PSI-BLAST sequence hits by the ELE weight.

The query protein used is human respiratory virus surface glycoprotein G (UniProt ID: P03423). This protein is present on the virus surface, for which GO terms such as virion membrane (GO:0055036), virion (GO:0019012), host cell surface (GO:0044228), extracellular region (GO:0005576), integral component of membrane (GO:0016021), and membrane (GO:0016020) are annotated in the CC category. The protein helps in attachment of the virus to the host cell membrane by interacting with heparan sulfate, initiating viral infection. This corresponds to GO annotations of virion attachment to host cell (GO:0019062), viral process (GO:0016032), evasion/tolerance by virus of host immune response (GO:0030683), and viral entry into host cell (GO:0046718) in the BP category. Phylo-PFP showed a high prediction accuracy, an Fmax score of 0.803 while it was 0.042 by PFP. If we calculate Fmax using the optimal score cutoff for this particular protein, then Phylo-PFP score was increased to 0.958, while PFP score increased to 0.741, still lower Phylo-PFP.

As shown in Table 3.5, Phylo-PFP predicted most of the correct GO terms with a high confidence score of 0.99 to 1.00, while PFP predicted them with a low score of 0.06 to 0.11. PFP instead predicted the incorrect terms, diaminopimelate metabolic (GO:0046451) and lysine biosynthetic process via diaminopimelate (GO:0009089) with the highest score of 1.0. These two incorrect GO terms came from sequence hits of diaminopimelate epimerase, which had a significant E-value (e.g. 1e-26 for bacterial diaminopimelate epimerase, UniProt ID: A6VQR8). In contrast to PFP, Phylo-PFP moved the ranks of Epstein-Barr virus envelope glycoproteins (e.g. Q3KST4, P03200, and P68344) higher, which are virus envelope proteins similar to the query protein. Figure 3.7 depicts how the sequence hits in PSI-BLAST search were reranked by ELE. The dendrogram shows the top 75 functionally similar sequences to the query, P03423 (shown in blue). Sequence hits are shown in green if their ranks went up by ELE in comparison with their original ranks in PSI-BLAST, which include the three proteins, Q3KST4, P03200, and P68344. Shown in red are sequences whose rank went lower by ELE. As illustrated, sequences that are less similar, i.e. far from the query in the dendrogram, went lower, while those more functional similar went up in the rank. Table 3.6 further illustrates the amended score contribution by ELE with a

few sequence examples. The three virus proteins in the table had insignificant E-values of 2.2, 0.23, and 0.55 respectively, and thus only contributed 2 to 10% of the scores relative to A6VQR8, a diaminopimelate epimerase sequence with a very small E-value. However, their relative contribution increased to 14 to 20% in ELE, which was sufficient, together with contributions of other functionally similar sequences to the query, P03423 (Figure 3.7), to rank correct GO terms with the highest confidence scores (Table 3.5).

Table 3.5. Confidence scores of correct GO terms for P03423 by Phylo-PFP and PFP. Two more GO terms discussed in the text are also listed.

Correct GO terms	Phylo-PFP Confidence Score	PFP Confidence Score
GO:0044228 (Host cell surface)	0.99	0.06
GO:0055036 (virion membrane)	0.99	0.06
GO:0046718 (viral entry into host)	1.00	0.11
GO:0005576 (extracellular region)	0.99	0.13
GO:0016021 (integral component of membrane)	1.00	0.09
GO:0030683 (evasion or tolerance by virus of host immune response)	1.00	0.11
GO:0019062 (virion attachment to host cell)	1.00	0.11
GO:0019012 (virion)	1.00	0.12
GO:0016032 (viral process)	1.00	0.11
GO:0016020 (membrane)	0.694	0.17
GO:0046462 (diaminopimerate metabolic process)	0.55	1.00
GO:0009089 (lysine biosynthetic process via diaminopimerate)	0.55	1.00

Table 3.6.Comparison of E-value, phylogenetic distance, and ELE of a few key PSI-BLAST hits for a query protein, P03423. *Diaminopimelate epimerase. # Epstein-barr virus envelope glycoprotein. a), the weight of the sequence used in PFP, i.e. –log(E-value) + b, relative to A6VQR8. b), the phylogenetic distance. c), ELE relative to A6VQR8.

Prot. ID	E-val.	-log(E)+b	Rel(PFP)[a)]	Phyl[b)]	ELE	Rel(ELE)[c)]	Func.
A6VQR8	1E-26	28.1	1.0	31.6	0.890	1.0	D.e*
Q5N013	2E-22	24.1	3	35.7	0.666	0.748	D.e
Q3KST4	2.2	0.758	0.027	13.9	0.126	0.142	E.-b.#
P03200	0.23	2.738	0.097	15.8	0.176	0.198	E.-b.
P68344	0.55	2.360	0.084	14.1	0.167	0.188	E.-b.

We discuss another case with sarcosine oxidase subunit β from *Corynebacterim sp. strain P-1* (UniProt ID: P40875). This protein catalyzes the oxidative demethylation of sarcosine into formaldehyde, glycine and hydrogen peroxide, which corresponds to GO annotations of oxidoreductase activity (GO:0016491) and sarcosine oxidase activity (GO:0008115) in the Molecular Function (MF) category as well as tetrahydrofolate metabolic process (GO:0046653) and oxidation-reduction process (GO:0055114) in the BP category. Phylo-PFP successfully predicted all the four GO terms yielding the Fmax score of 1.00 (1.00) while PFP's Fmax was 0.518 (0.623). Shown in the parentheses are the Fmax value obtained when optimized for this target protein. Table 3.7 shows confidence scores predicted for the correct GO terms by Phylo-PFP and PFP as well as two more terms that are over-predicted by PFP (the two terms at the bottom of the table). While analyzing PSI-BLAST hits to understand the different performance between

Table 3.7. Confidence scores of correct GO terms for a query protein P40875 by Phylo-PFP and PFP. Two more GO terms to be discussed in the text are also listed.

Correct GO terms	Phylo-PFP Confidence Score	PFP Confidence Score
GO:0016491 (oxidoreductase activity)	1.00	1.00
GO:0055114 (oxidation-reduction process)	1.00	1.00
GO:0008115 (sarcosine oxidase activity)	0.59	0.02
GO:0046653(tetrahydrofolate metabolic process)	0.47	0.02
GO:0005737 (cytoplasm)	1.00	0.61
GO:0008033 (tRNA processing)	0.07	0.38
GO:0008168 (methyltransferase activity)	0.13	0.38

the two methods, we found that sequence hits with significant E-values included both functionally related and non-related sequences. For example, Q9AGP3 and O87388, both of which are sarcosine oxidase were identified with a significant E-value (E-103 and 1E-76, respectively), while tRNA biosynthesis proteins were also among the top hits, Q8ZNB2 (E-value: 7E-73), A9MJ47 (E-value: 2E-72), and A9N465 (E-value: 2E-73). This caused PFP to predict GO terms related to these proteins, such as tRNA processing (GO:0008033) and methyltransferase activity (GO:0008168) with a medium level confidence score. In contrast, Phylo-PFP effectively reranked sequence hits with the ELE weight correctly, as shown in Table 3.8, by considering the phylogenetic distance of the sequence hits. Ranks of tRNA biosynthesis proteins, Q8ZNB2, A9MJ47, and QA9N467, were lowered due to their large phylogenetic distances (more than 85), while sarcosine oxidase sequences, Q9AGP3 and O87388, moved up in the rank because their phylogenetic distances were relatively small, 0.23 and 20.31 respectively. Consequently, Phylo-PFP managed to predict correct GO terms of sarcosine oxidase with a high score (Table 3.8). Oxidoreductase activity (GO:016491) was predicted with the highest confidence by both Phylo-PFP and PFP because it is the common annotation between sarcosine oxidase and tRNA biosynthesis proteins. Figure 3.8 illustrates this situation of sequence reranking by ELE for Phylo-PFP. Among those which were moved up in the rank (shown in green) are functionally similar to the query (blue) including Q9AGP3 and O87388, whereas the three tRNA biosynthesis proteins, Q8ZNB2, A9MJ47, and QA9N467 were among sequences which were moved to lower ranks.

Table 3.8. Comparison of E-value, phylogenetic distance, and ELE of a few key PSI-BLAST hits for a query protein, P40875.

Protein ID	E-value	Phylo distance	ELE	Description
Q8ZNB2	7E-73	85.58	0.868	tRNA biosynthesis bifunctional protein MnmC
A9MJ47	2E-72	87.10	0.825	tRNA biosynthesis bifunctional protein MnmC
A9N465	2E-73	85.30	0.877	tRNA biosynthesis bifunctional protein MnmC
Q9AGP3	E-103	0.23	457.059	Sarcosine oxidase
O87388	1E-76	20.31	3.845	Sarcosine oxidase

3.3.5 Prediction on the CAFA2 target protein dataset

We further tested Phylo-PFP on the target protein sequence dataset used in CAFA2 to compare the performance with top performing methods in the assessment. In total, 56 groups submitting 126 methods participated in CAFA2.

We compared the performance of Phylo-PFP with the best performing methods in CAFA2 as well as PFP and with a baseline method, BLAST (Table 3.9). The top performing methods from CAFA2 were taken from Figure 4 of the CAFA2 evaluation report [73].

Table 3.9. The Fmax score of predictions for the CAFA2 dataset by Phylo-PFP in comparison with top performing methods in CAFA2. Results of the three GO categories are separately shown. Fmax scores of the methods participated in CAFA2 were taken by matching the supplemental data and Figure 4 of the CAFA2 evaluation report. Dashes (-) indicate that method did not appear among top 10 methods in Figure 4. The largest Fmax value for each GO category is highlighted in bold.

Method	MF	BP	CC
MS-KNN	0.595	0.363	0.455
EVEX	0.593	-	0.468
Paccanaro Lab	-	0.372	-
Tian Lab	0.591	0.367	0.462
Orengo-FunFams	0.569	0.352	0.438
Go2Proto	0.563	-	-
SIFTER	0.561	-	-
INGA-Tosatto	0.555	0.347	-
Jones-UCL	0.554	0.352	0.450
Argot2	0.544	0.351	-
Gough Lab	-	0.352	0.458
PULP	-	0.350	0.441
Rost Lab	-	-	0.442
IASL	-	-	0.439
PFP	0.574	0.348	-
CONS	-	-	0.446
BLAST	0.473	0.251	0.347
Phylo-PFP	**0.606**	**0.380**	**0.506**

Remarkably, Phylo-PFP outperformed the other methods in all three categories with an Fmax score of 0.606, 0.380, and 0.506 for MF, BP, and CC category, respectively. Considering the methods from CAFA2, a different method excelled for each GO category and no method showed consistent high performance among all the categories. MS-KNN scored the highest in MF with an Fmax of 0.595, Paccanaro Lab was the top among the existing methods in BP with an Fmax of 0.380, while EVEX was best in CC with an Fmax of 0.372. This is a clear contrast with Phylo-PFP, which exhibited the best performance in all the three categories.

3.3.6 Computational time

The computational time for running Phylo-PFP are shown in comparison with PFP and PSI-BLAST in Table 3.10. To examine how the computational time grows as the number sequence hits by PSI-BLAST increases, the computational time was measured for three different numbers of hits, 10, 100, and 500. The time needed for Phylo-PFP increased substantially as the number of sequence hits grew, mainly due to the time needed to construct a phylogenetic tree from the sequence hits. With 10 sequence hits, Phylo-PFP took 3.1 times the computational time of PFP, which grew to 21.8 times when 500 sequence hits were retrieved.

Table 3.10. Computational time of the prediction methods. Computational times shown are the average values of ten query sequences in the unit of seconds. Hits (columns) indicate the number of sequence hits by PSI-BLAST. The number of hits were limited to 10, 100, and 500, for each method. The computations were performed on a computer operated by Linux with Intel Core i7-920 CPU 2.67GHz with 24.6 GB RAM.

Methods\Hits	10	100	500
PSI-BLAST	15.7	23.0	31.2
PFP	18.2	28.3	44.7
Phylo-PFP	56.6	103.5	975.2

3.3.7 Performance in CAFA3

We participated in world-wide function prediction competition CAFA3 with Phylo-PFP as wella as an ensemble method CONS [100] that included Phylo-PFP with the highest weightage. We updated the annotation and PSI-BLAST database in 2016, before the start of the competition. Our method came 2nd in CC category, 4th in MF category and 6th in BP category out of 68 teams and 144 total methods [101]. Results of the competition are shown in Figure 3.9. For human proteins, our method came 1st in CC category, 2nd in MF category and 1st in BP category, shown in Figure 3.10.

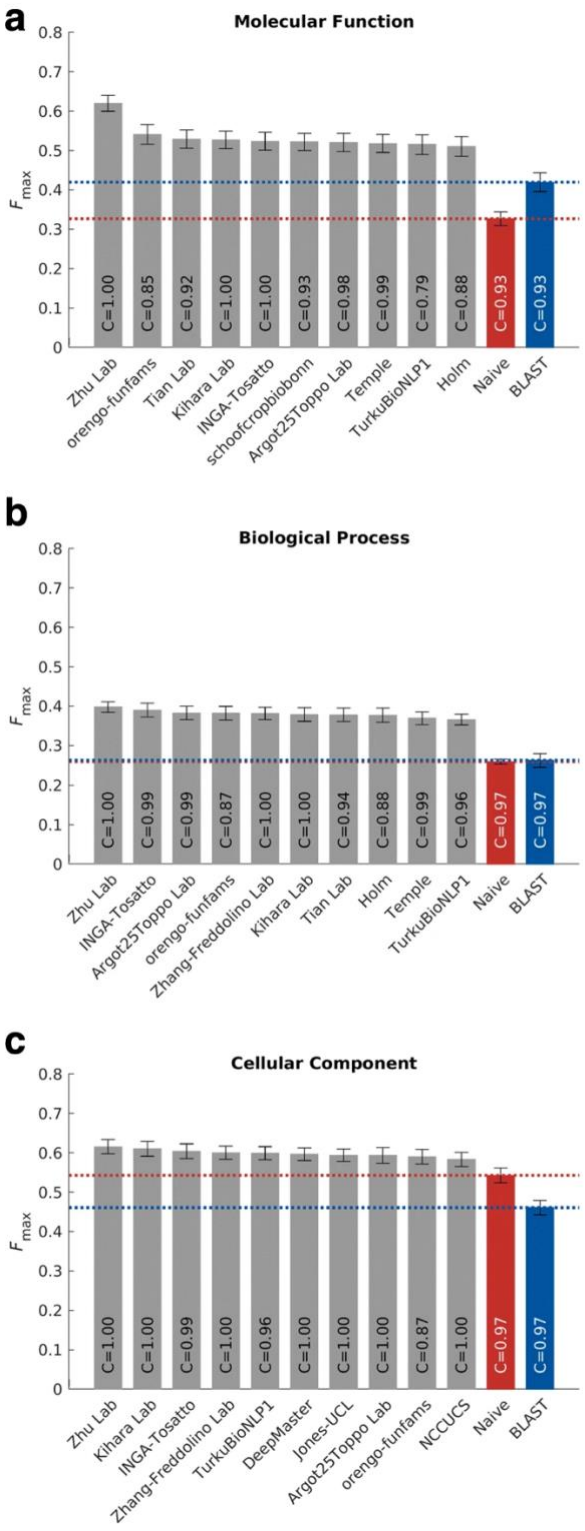
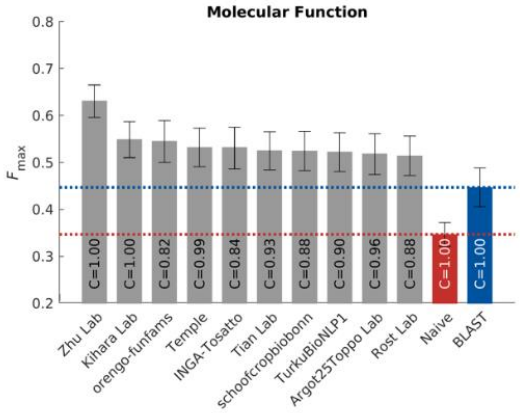
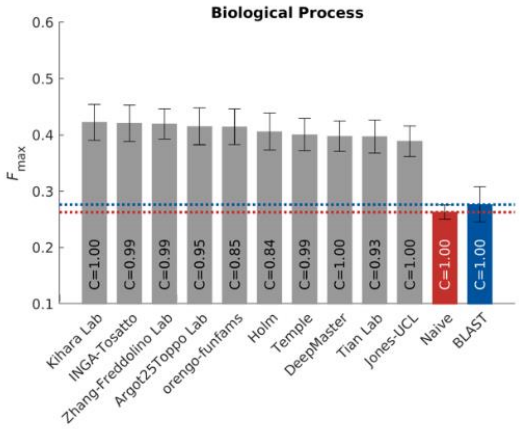


Figure 3.9. Performance evaluation based on the Fmax for the top-performing methods in CAFA3. Figure was obtained from Figure 3 of [101]



(a)



(b)

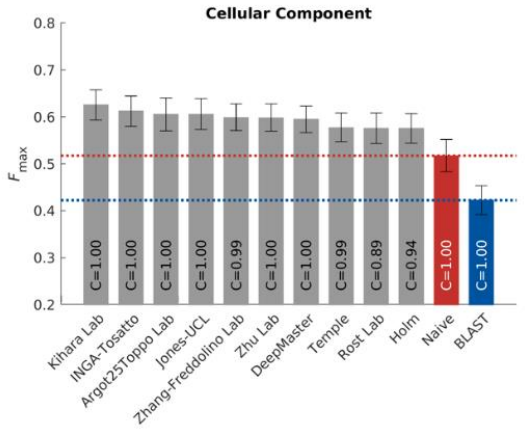


Figure 3.10. Fmax for the top-performing methods in CAFA3 for human targets. Figure was obtained from Figure S6 of [101]

3.4 Discussion

In this study, we developed a new sequence-based protein function prediction method, Phylo-PFP, which substantially improved the prediction accuracy from its predecessor, PFP, by using phylogenetics to determine the evolutionary distance of sequences retrieved from a database searches.

It has been discussed that the sequence similarity does not often accurately capture evolutionary relationship of sequences [94]. Here we showed that there was no strong correlation between the database search scores and the phylogenetic distances for most of the sequences in a large dataset on a realistic scenario of PSI-BLAST search. Subsequently, as a practical solution for improving PFP, we implemented a distance-based phylogenetic analysis, and achieved favorable prediction accuracy improvements. Phylo-PFP takes more computational time than PFP especially when the number of PSI-BLAST hits is large. A practical solution for performing prediction for many sequences would be to run the method in parallel on multi-core CPUs.

Further improvement of the accuracy is expected by considering several approaches. For example, instead of the distance-based phylogenetic analysis we used in this work, a more accurate tree construction technique such as maximum likelihood [102] or Bayesian inference [103, 104] may be used. Also, functional domain [105] or residue information [35, 106] can be explicitly considered, as currently functional transfer is performed in PFP and Phylo-PFP only by global sequence similarity.

CHAPTER 4. GENE ONTOLOGY-BASED PROTEIN TOXICITY PREDICTION

With advancements in synthetic biology, the cost and the time needed for designing and synthesizing customized gene products have been steadily decreasing. Many research laboratories in academia as well as industry routinely create genetically engineered proteins as a part of their research activities. However, manipulation of protein sequences could result in unintentional production of toxic proteins. Therefore, being able to identify the toxicity of a protein before the synthesis would reduce the risk of potential hazards. Existing methods are too specific, which limits their application. In this chapter I extended general function prediction methods for predicting the toxicity of proteins. Protein function prediction methods have been actively studied in the bioinformatics community and have shown significant improvement over the last decade. On top of our function prediction method Phylo-PFP described in chapter 3, we developed a neural network model, named NNTox, which uses predicted GO terms for a target protein to further predict the possibility of the protein being toxic. We have also developed a multi-label model, which can predict the specific toxicity type of the query sequence. Together, this work analyses the relationship between GO terms and protein toxicity and builds predictor models of protein toxicity.

4.1 Background

Proteins carry out various functions in a cell, forming functional networks and signaling pathways that are essential to sustain life. Understanding the function of component proteins in the networks is a fundamental step to obtain critical insights into complex cellular mechanisms. As a means to elucidate the function of a protein and the relationship between the function and the sequence or the structure of the protein, experimentally, it is common to construct mutants of the protein and test their function in vitro and in vivo. Advancements in synthetic biology [107, 108] as well as protein design [109] have made it now possible to construct artificial proteins that fold and assemble into desired structures and achieve specific tasks in a cell. Artificial protein synthesis has also revolutionized the biotechnology industry, where the technique has been used to program

microbes to produce drugs at reduced production cost, to create disease-resistant crops that improve the yield, or to design new vaccines and therapeutic antibodies to cure diseases [110-112].

While there are many applications of constructing desired artificial peptides and proteins, a potential problem is the production of harmful or toxic proteins. There are two scenarios where toxic proteins may be constructed: One situation would be that a newly designed protein happens to have an unexpected harmful function. There are many aspects of cell function that are still unclear, thus, foreseeing such side effects when designing a new protein may be very difficult. The second possible case would be an intentional design or release of toxic proteins for biological attack [113]. To prevent release of toxic proteins, there are ongoing efforts to build systems and devices that collect unknown proteins or organisms together that identify proteins with potential harm[114-117]. There is a strong demand for such systems for lab facilities of gene synthesis, places where many people gather, e.g. airports, and war zones where biological attack might occur.

A computational algorithm for detecting toxic proteins should take a protein or DNA sequence as input and alerts if the protein can be harmful. ThreatSEQ developed by Battelle Memorial Institute identifies sequences of concern by comparing them with a curated database of known toxic proteins [118]. ToxinPred [119] and other series of methods developed by the Raghava group target detection of toxic bacterial peptides using machine learning methods based on sequence information [120, 121]. ClanTox uses a machine learning method that was trained on known peptide ion-channel inhibitors [122]. These methods are similar in approach in that they use sequence information. Moreover, the methods except for ThreatSEQ have a limited application to peptide toxins.

In this paper, we present a new method, NNTox (Neural Network-based protein Toxicity prediction), which can predict the toxicity of a query protein sequence based on the protein's Gene Ontology (GO) annotation [123]. GO is a controlled vocabulary of function of proteins and has been widely used for function annotation and prediction. Previously, our lab has developed a series of function prediction methods [124, 125] including PFP [39, 126, 127] and Phylo-PFP [125], which have been shown to be among the top-performing function prediction methods in the community-wide automatic function prediction experiment, Critical Assessment of protein Function Annotation (CAFA) [72, 73]. Here, we show that the toxicity of proteins can be well predicted from GO terms that are predicted by PFP. First, we examined the distribution of GO terms in annotations of toxic proteins and showed that GO terms are promising features for

predicting toxicity. Next, we developed a neural network for predicting protein's toxicity from their GO term annotations. Finally, we have further extended the method to the mode of action of toxicity of a protein.

4.2 Methods

First, we will describe the datasets used in this study. Then, we explain the neural network model of NNTox.

4.2.1 Toxic protein dataset

Toxin proteins were collected from the UniProtKB-SwissProt database [128] using the keyword "Toxin" (UniProtKB KW-0800). A total of 6,497 toxin proteins were obtained. From the 6,497 toxin proteins, we collected a set of 1,506 unique GO terms that were included in their GO annotations. The GO term of "toxin activity" (GO:0090729) was removed from the collection because this term obviously related to toxicity and can bias prediction if it is included in the annotation of proteins in the training and testing set for the toxicity prediction. From this toxin protein set, we removed proteins that were redundant to other proteins in terms of their GO term annotations. We did not use sequence similarity for the redundancy criterion because the input to our model is GO terms. The non-redundant dataset contained 488 toxin proteins.

Non-toxin proteins were also collected from UniProtKB SwissProt using the following two conditions: 1), they are not tagged with the keyword "Toxin". 2), 95% of GO terms annotating the protein belong to the toxin GO term set. The second criterion makes most of the GO term annotation of toxin and non-toxin proteins very similar. Using this approach 82,583 non-toxin proteins were obtained. Then, as was done for the toxin protein dataset, proteins with redundant GO annotations were removed, which resulted in 6,594 non-toxin GO proteins.

The Toxin keyword had 11 sub-classes, which were cardiotoxin (134/8), enterotoxin (94/12), neurotoxin (2744/100), ion channel impairing toxin (2429/74), myotoxin (121/22), dermonecrotic toxin (148/4), hemostasis impairing toxin (865/95), G-protein coupled receptor impairing toxin (186/33), complement system impairing toxin (160/6), cell adhesion impairing toxin (207/18), and viral exotoxin (9/4). The first number in the parentheses is the total number of proteins in the sub-class downloaded from UniProtKB-SwissProt while the second number is those in the non-redundant toxin proteins. Using this information, we compiled a dataset of the mode of

action of the toxin proteins. Out of the 488 non-redundant toxin proteins, 270 proteins had information of the mode of action. A protein is assigned to multiple classes if it belongs to more than one sub-class keywords. Out of the 270 proteins, 173 proteins belong only to one sub-class, 88 proteins have two assigned sub-classes, and 9 have three sub-classes.

4.2.2 Feature vector representing a protein

A protein in the dataset is represented by a vector of 2,596 binary (1 or 0) values (except for the last position), which indicates existence of the particular GO term in its GO annotation. 2595 GO terms represents all the GO terms found in toxin proteins as well as general GO terms that frequently appear in UniProtKB database (concretely, all GO terms that annotate more than 1000 proteins). The last position of the vector represents the number of GO terms that are associated with the protein but are not present among the above 2,595 GO terms. Using only toxin GO terms in the feature vector limits the scope of GO terms that the network can see and using all (>35,000) will lead to spare features. As a middle ground, we added top background GO term in the feature vector as well.

4.2.3 Neural network models

We used a five-layer fully connected feedforward neural network for the toxin/non-toxin prediction (Figure 4.1). The input layer has 2,596 neurons representing the GO term feature vector. The input layer is connected to three hidden layers, each of which has 200 neurons. The last layer uses the softmax nonlinearity to convert the output into class probability, toxin and non-toxin. Neurons are connected with a sigmoidal activation function. The code is available at http://www.github.com/kiharalab/NNTox.

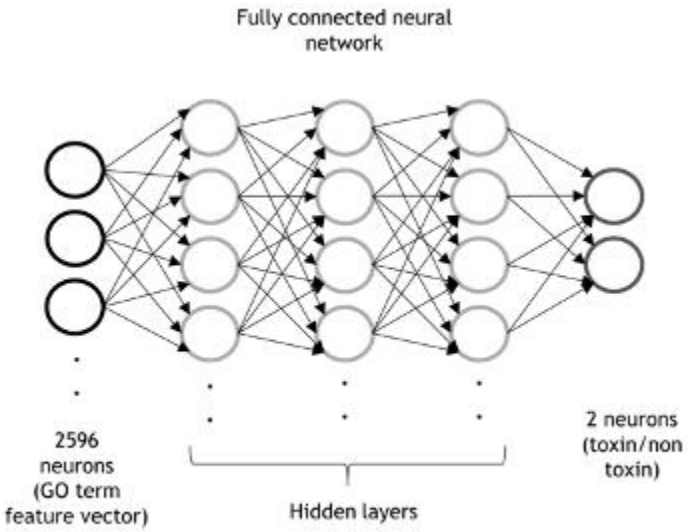


Figure 4.1. The network architecture of NNTox for toxin/non-toxin binary prediction.

Predicting the mode of action of toxin proteins is a multi-label classification problem, where one toxin could have more than one mode of action. For example, conotoxin, a snail toxin, is both a neuro-toxin and an ion channel inhibitor toxin. Thus, classes are not mutually exclusive. We modified the neural network described above to perform multi-label prediction, by replacing softmax in the last layer with computing the sigmoid cross-entropy loss. In the sigmoid cross-entropy loss, the loss calculated for every label is independent of the loss in other labels, and thus allows for multiple labels to be predicted.

The sub-classes of toxins are imbalanced, e.g. neurotoxin and ion channel inhibiting toxin have more proteins than other sub-classes. This can cause bias in the network while training towards highly represented classes. To overcome this problem, we added a weight to each correct class prediction in the multi-label neural network, where the weight is inversely proportional to the number of the times that class is present in the training set. For a protein, $v = [v_1, v_2, ,,, v_{11}]$ is the label vector, where $v_i = 1$ represents that the protein has the mode of action i. For each mode of action i, we calculated the positive count (i), i.e., the number of times $v_i=1$ and the negative count (i), i.e., the number of times $v_i=0$ in the training dataset. The weight w_i given to a mode of action/class i is w_i = (negative count (i))/(positive count (i)). Thus, the weight is 1 if the number of positive and negative counts is equal while giving more weight as the positive count decreases.

4.2.4 Training and validation with nested cross-validation

Training was performed with backpropagation using the ADAM optimizer, implemented in TensorFlow [129] We performed a five-fold nested cross validation to tune four hyper-parameters: the number of neurons in hidden layer [10, 50, 100, 200, 500], the regularization strength [10, 1, 0.1, 0.01, 0.001], the learning rate [10, 1, 0.1, 0.01, 0.001] and the number of epochs [100, 500, 1000, 2000, 5000]. Shown in the parentheses are the values tested for each hyper-parameter.

Nested cross-validation provides robust and unbiased training and testing using the full data available from the dataset. In the nested cross-validation there were two cross validation loops. In the outer loop, the dataset was divided into k (=5) subsets, where one subset was considered as the test set and the rest are used for training & validation set, and the test set was changed for k times. Furthermore, the inner loop was to perform a cross-validation on the training & validation set, i.e. the set was divided into k (=5) pieces again and one of them was considered as the validation set. Each different combinations of hyper-parameters were trained on the training set and tested on the validation set. This was performed for k times by changing the validation set. Then, the best hyper-parameter was chosen based on the average error on the k validation set, and the model trained using the hyper-parameter set on all training and validation set was applied to the testing set. This is repeated for k times, and the final result was the average performance on the k test sets.

4.2.5 Protein function prediction with PFP

We examined the performance of NNTox using two sets of GO terms for proteins. First, we tested NNTox using the GO annotations of proteins obtained from UniProtKB-SwissProt. This is to test the performance of the architecture of NNTox in the best possible cases when all the correct GO terms are known. Second, we used a GO-term prediction method, PFP, to predict GO terms of each protein and trained NNTox on the predicted GO terms. This is to simulate the situation when true GO terms for a query protein are not present.

PFP was developed in our group and has been successful in the Critical Assessment of protein Function Annotation algorithms (CAFA). PFP uses PSI-BLAST [130] to retrieve similar sequences from a database to a query sequence and obtains GO-term annotations from the sequences with an E-value of up to 125. Then, each GO term will be assigned with a score that

reflects the E-value of sequences that have the GO term in their annotation as well as the conditional probability that the GO term occurs given other GO terms are observed. For the sequence database, we used UniProtKB Swiss-Prot downloaded in March 2018. To avoid retrieving GO terms from the query protein itself, sequences retrieved with an E-value of 0 were discarded.

PFP provides a confidence score to each GO term predicted that ranges from 0.0 to 1.0 with 1.0 for the highest confidence (Appendix Table A1). Using PFP, we devised a simple baseline strategy to predict if a protein is toxin or not directly from assigned GO terms. If PFP predictions include the "toxin activity" GO term (GO:0090729) with high confidence (>= 0.9) then we label the protein as a toxin. We also trained NNTox network with PFP-predicted GO terms. Only predicted terms were used for this training, i.e. known GO term annotations were not considered to simulate the situation that query proteins do not have any known annotations. We removed the "toxin activity" GO term from the PFP predictions as having this GO term would bias the model and make the toxin prediction easy.

4.2.6 Additional baseline method

To evaluate the performance of NNTox, we developed a naïve GO term based baseline approach. In this approach, a protein is classified as toxin if all the GO terms associated with it are present in the Toxin GO term set. This approach reflects the idea that if a set of GO terms are already known to be associated with a toxin, we classify a new protein associated with those GO terms as toxin as well. For baseline method, the non-redundant toxin protein dataset was split into a 70:30 train:test ratio, where 70% of the dataset was used to create the Toxin GO term set. The method was tested with 30% of the toxin test dataset and all the non-redundant non-toxin proteins.

4.2.7 Prediction evaluation

Prediction accuracy was evaluated with the F1 score. The precision P, recall R and F1 score was calculated as

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 * P * R}{(P + R)}$$

where TP is the total number of proteins that are toxin and were predicted correctly as toxin, FP is the total number of proteins that are non-toxin but predicted as toxin, and FN is the total number of proteins which are toxin but predicted as non-toxin.

4.3 Results

4.3.1 GO term specificity for toxin proteins

To begin with, we examined if any GO terms have a specific association with the toxicity of proteins. We computed the specificity of GO terms for toxin proteins, which was defined as the fraction of the toxin proteins that are annotated with the specific GO term among all proteins in UniProtKB-SwissProt with the GO term annotation. Table 4.1 lists top 20 GO terms with the highest toxin specificity. Appendix Table A2 provides a complete list of GO terms associated with toxin keywords. Besides GO terms that are apparently related to toxins, e.g. those with the word "inhibitor" in their description, there are highly toxin-specific terms that do not directly indicate toxicity.

The first GO term in the table, "Other organism postsynaptic membrane" (GO: 0035792) has 100% of the toxin specificity. Proteins with this GO term are indeed toxins, e.g. alpha-conotoxin in a sea snail (Uni-Prot ID: CDKA_CONVX) and cobrotoxin in Chinese cobra (UniProt ID: 3S1CB_NAJAT). These toxins bind to nicotinic acetylcholine receptors, inhibiting them, and impairing neuromuscular transmission. Thus, it is involved in neurotoxicity and ion channel impairing toxicity. "N-acylphosphatidylethanolamine-specific phospholipase D (NAPE-PLD) activity" (GO: 0070290, example proteins: UniProt ID: A1lB1_LOXIN) has a high toxin specificity of 75.35%. Phospholipid D catalyzes the hydrolysis of sphingomyelin and induces complement-dependent hemolysis, dermonecrosis, blood vessel permeability, and platelet aggregation. Thus, it is involved in dermonecrotic and complement system toxicity. It is possessed by recluse spiders and causes necrotic damage. "Phospholipase A2 activity" (GO:0004623), the

last one in the table, has a toxin specificity of 58.41% with neurotoxin specificity of 22%, myotoxin specificity of 14%, and hemostasis impairing toxin specificity of 23%. Phospholipase A2 catalyzes the calcium-dependent hydrolysis of the 2-acyl groups in 3-sn-phosphoglycerides. It affects neuromuscular transmission by blocking acetylcholine release from the nerve termini. It also has anticoagulant activity and weakly inhibits ADP-induced platelet aggregation. The protein with this activity exists in venomous snakes, e.g. Chinese krait (UniProt ID: PA2B1_BUNMU) and Nikolsky's Viper (UniProt ID: PA2B2_VIPBN). Overall the results show GO terms are promising features for predicting protein toxicity.

Table 4.1. Toxin specific GO terms.

GO ID	Function	Toxin Spec. (%) [a)]
0035792	other organism postsynaptic membrane	100.00 (554)
0072556	other organism presynaptic membrane	98.14 (317)
0042151	nematocyst	91.64 (252)
0030550	acetylcholine receptor inhibitor activity	91.11 (123)
0019871	sodium channel inhibitor activity	89.89 (169)
0008200	ion channel inhibitor activity	87.89 (1415)
0016248	channel inhibitor activity	87.56 (1415)
0099602	neurotransmitter receptor regulator activity	75.46 (123)
0034548	acetylcholine receptor regulator activity	75.46 (123)
0070290	N-APE-PLD D activity [b)]	75.35 (214)
0004630	phospholipase D activity	75.09 (214)
0016247	channel regulator activity	71.72 (1415)
0030547	receptor inhibitor activity	69.44 (125)
0009405	pathogenesis	66.26 (6497)
0102568	phospholipase A2 activity (12-DOPE) [c)]	59.51 (319)
0102567	phospholipase A2 activity (12- DPPtdCho) [d)]	59.51 (319)
1903963	arachidonate transport	59.48 (342)
0050482	arachidonic acid secretion	59.47 (342)
0017080	sodium channel regulator activity	59.31 (172)
0004623	phospholipase A2 activity	58.41 (375)

a) the number of toxin proteins with the GO term is shown in the parenthesis. b), N-acylphosphatidylethanolamine-specific phospholipase D activity. c), phospho-lipase A2 activity consuming 12-dioleoylphosphatidylethanolamine. d), phospho-lipase A2 activity (consuming 12-dipalmitoylphosphatidylcholine).

4.3.2 Performance of toxin prediction

In this section we discuss the performance of our NNTox on distinguishing toxin and non-toxin proteins. We compare the performance with the baseline methods. Table 4.2 summarizes the results. The table shows precision, recall, and the F1 score, which was defined as the harmonic mean of precision and recall of toxin protein prediction.

Table 4.2. Summary of the toxin prediction.

Method	Precision	Recall	F1 score
With GO annotation			
Baseline exact	0.029	0.626	0.055
Baseline 1 mismatch	0.023	0.714	0.044
Baseline 2 mismatches	0.021	0.769	0.041
NNTox (GO Annotation)	0.903	0.898	0.900
With PFP prediction			
Baseline exact	0.110	0.156	0.129
Baseline 1 mismatch	0.102	0.184	0.131
Baseline 2 mismatches	0.115	0.259	0.159
PFP	0.873	0.535	0.663
NNTox (PFP)	0.801	0.750	0.775
PFP + NNTox(PFP)	0.807	0.781	0.794

The baseline method is explained in Methods. NNTox (GO Annotation) used the GO annotations of proteins from UniProtKB-SwissProt. "PFP" checked if the "toxin activity" GO term was predicted with 0.9 or a higher confidence score. NNTox (PFP) uses predicted GO terms by PFP using 0.1 as the prediction confidence cutoff value (Figure 4.2). PFP + NNTox(PFP) is a two-step prediction using first PFP and then to apply NNTox(PFP) for proteins that are not identified as toxin by PFP.

In the first three rows of Table 4.2, we showed the prediction performance one can obtain by simply comparing GO annotation of a target protein with known proteins in the reference database (the baseline method). When the exact match of GO terms was counted, recall for toxin proteins was 0.626. When the condition was relaxed, allowing 1 or 2 miss matches of GO terms, the recall for toxin proteins naturally increased to 0.714 by sacrificing the precision. This is intuitive because with 1 mismatch allowed, proteins which had only one GO term not present in the toxin GO set were now predicted as toxins as well but with the cost of false positives. F1 scores

of the baseline method were as low as 0.055 due to low precision values that were caused by a large number of false positives (i.e. non-toxin proteins predicted as toxins).

In contrast, prediction by NNTox performed substantially better than the baseline method. The precision and recall for detecting toxin proteins was 0.903 and 0.898, respectively, indicating that the predictions made for toxin and non-toxin proteins were well balanced. The NNTox F1 score was 0.900, which is a clear contrast compared to baseline method that showed substantially lower F1 score.

The second half of Table 4.2 shows results using PFP predicted GO terms. Using predicted GO terms, the baseline method showed lower recall as compared with results using GO annotations. This is because predicted GO terms for a protein have a low random chance to perfectly agree with toxin GO terms. As another baseline, PFP prediction was also directly used to determine if a protein is toxin by checking if the prediction included "toxin activity" GO term with a high confidence (>= 0.9). This approach performed better than the baseline method showing an F1 score of 0.663 and a recall of 0.535. Thus, about half of the toxin proteins were identified correctly by the PFP baseline. NNTox performed better than the baseline methods and the PFP baseline with an F1 score of 0.775, although the performance was worse than the cases with correct GO annotation. For NNTox, we used predicted GO terms with PFP's confidence score of over 0.1, since that gave the best performance (Figure 4.2). We also tested a two-step prediction process

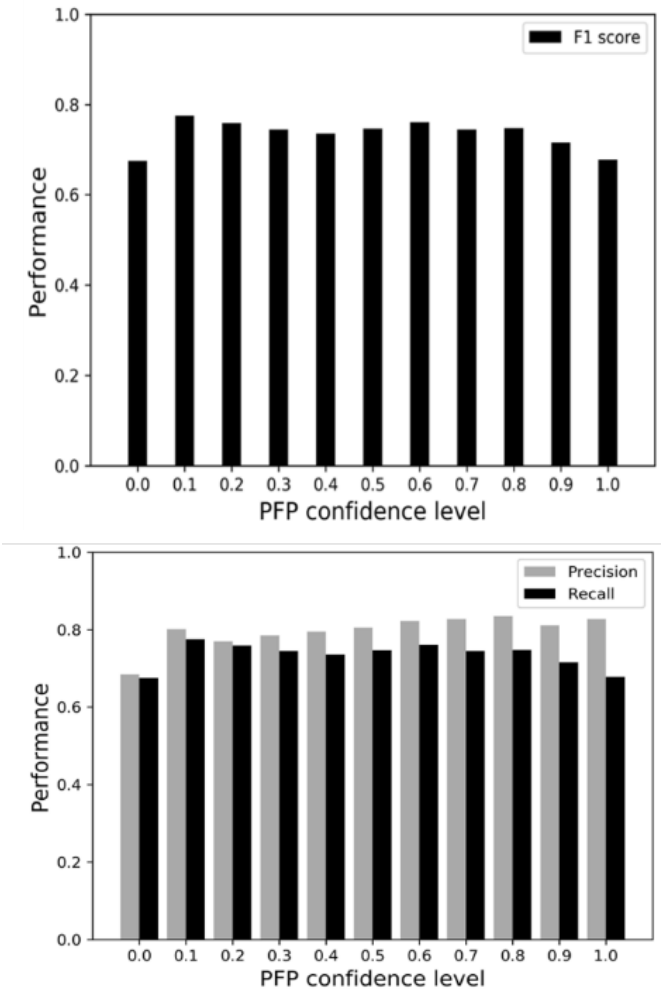


Figure 4.2. F1 score, precision, and recall of toxin prediction for different PFP's GO prediction confidence levels.

where PFP and NNTox with PFP predicted GO terms were combined (the last row in Table 4.2). First, the protein was determined to be toxin based on direct PFP predictions. Then, if the protein is not predicted to be toxin, then NNTox was applied. This procedure further improved NNTox in all the evaluation metrics. The F1 score increased from 0.775 to 0.794. Looking closely, the first step of the PFP application filtered 261 toxin proteins correctly (i.e. true positives), then additional 120 toxin proteins were selected by the NNTox.

In Figure 4.3, we analyzed the importance of each GO term in distinguishing toxin and non-toxin proteins. For each GO in the feature vector, we computed the mutual information to the toxin classification relative to the toxin specificity (Table 4.1). As shown, a large specificity of a GO term does not necessarily mean a large mutual information for the classification. Such cases

happen for GO terms that are highly specific for toxins but only appear in annotation of a small number of proteins, thus not much helpful for the classification for many proteins in the dataset. The top three GO terms were pathogenesis (GO:0009405), interspecies interaction between organisms (GO:0044419) and multi-organism process (GO:0051704), which is not surprising as these terms highly indicative of a protein being toxin.

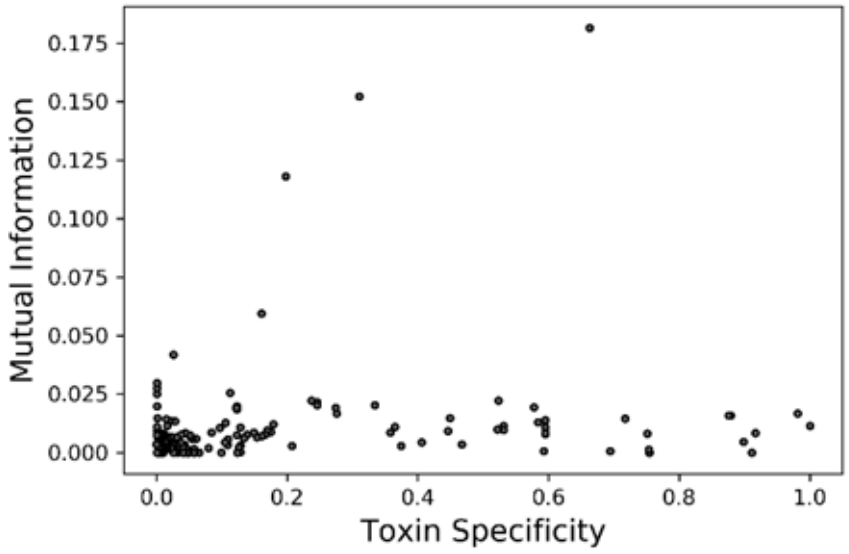


Figure 4.3. Mutual information and toxin specificity of GO terms for toxin/non-toxin classification.

4.3.3 Neural network visualization

In Figure 4.4 we visualized the network to illustrate how the neural network model separated the toxin and non-toxin proteins using the principal component analysis (PCA). For each protein in the non-redundant tox-in/non-toxin set, we ran the trained network and calculated the output of each of the three hidden layers and passed it through the sigmoid activation function. The top figure shows that toxin proteins (red) mostly overlapped with non-toxin proteins in the PCA space. The distinction between the two classes became substantially clearer in the second layer (the middle panel), and further improved in the third layer. Thus, as the network went deeper and the model complexity increased, the model was able to separate the two classes better.

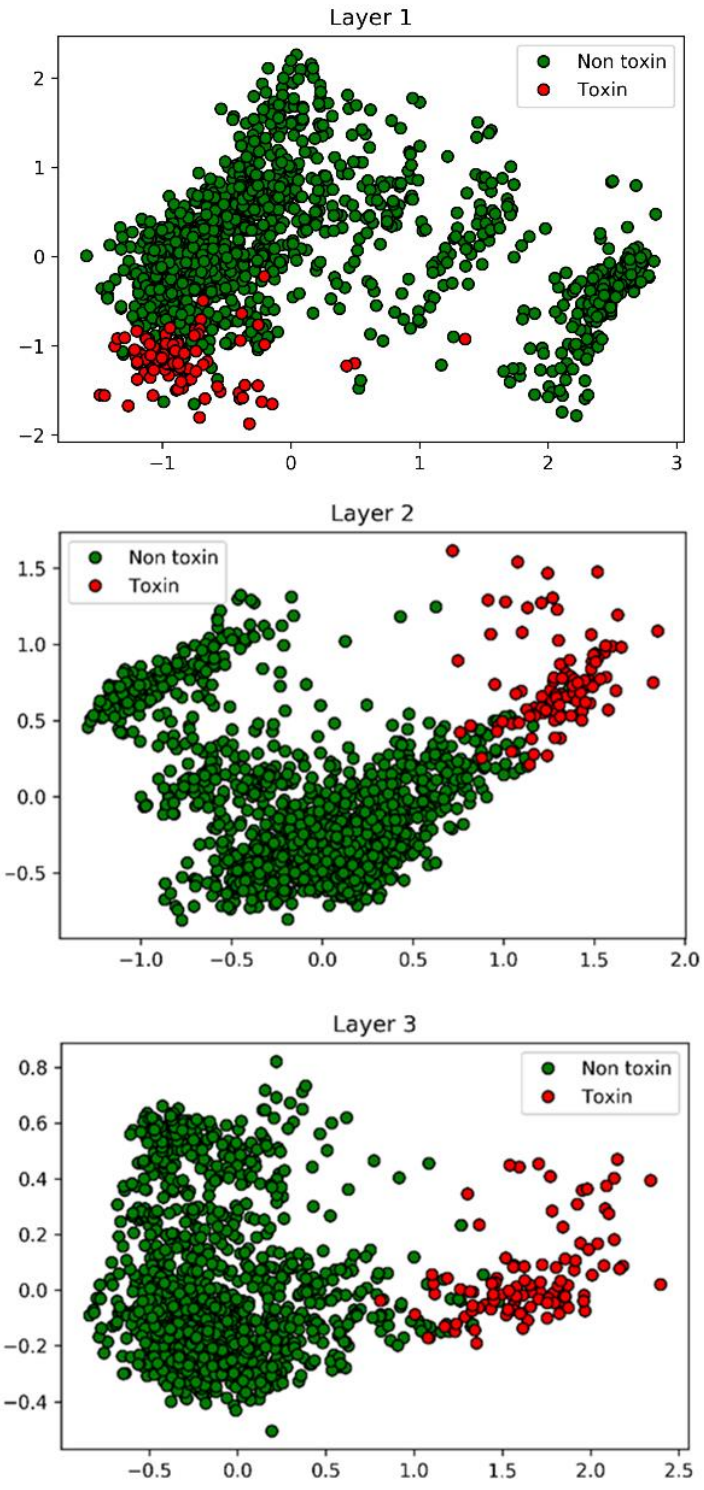


Figure 4.4. Separations of toxin and non-toxin proteins in the neural network layers. Outputs from each of the three hidden layers of the neural network for toxin (red) and non-toxin (green) proteins are visualized by PCA. The x- and the y-axis are the first and the second principal components of the output values of the layer through the sigmoid activation function.

4.3.4 Prediction of toxin mode action

Next, we developed a multi-label neural network model, which predicts the mode of action of a toxin protein. The input to the model is the same feature vector of GO terms and the output is a binary vector for the 11 modes of action. Multiple action predictions are also allowed for a protein, which makes the prediction task more complex. To evaluate the prediction performance of the model, we computed the elementwise accuracy of the predicted vector (Table 4.3) as usually used for multi-label classification [131], where the number of correctly predicted modes for each of the target proteins was counted. NNTox (Mode of action) showed good performance with an accuracy of over 0.8, even when predicted GO terms were used. The high accuracy indicates that the method was overall successful in not only for pointing out the correct mode of the toxin proteins but also in avoiding over predicting incorrect modes.

Table 4.3. Summary of the mode of action prediction accuracy.

Input GO terms	Accuracy
UniProtKB	0.879
Prediction by PFP	0.825

The values are the average for test sets in the five-fold nested cross-validation. In this multi-class prediction, a prediction output for a protein is a binary vector of 11 values, where 1 indicates the class is predicted and 0 for a negative prediction for the class. The accuracy was computed by counting the agreement of the predicted binary class for each toxin mode of action in all the proteins.

Figure 4.5 shows the F1 score of each mode of action separately for toxin proteins with a single action mode. Precision and recall values are provided in Appendix Table A3 and A4, respectively. Naturally, F1 scores correlated strongly with the number of data available for modes, which is shown in the parentheses of the mode labels on the x-axis. A relatively high F1 score was observed for modes that have more data, but low scores were resulted in for modes with small data size. Thus, the data availability of the current database limits the prediction performance for several toxin modes, nevertheless, the results indicate that in principle the model is reasonable and will only improve by the increase of toxin data to be available in the future.

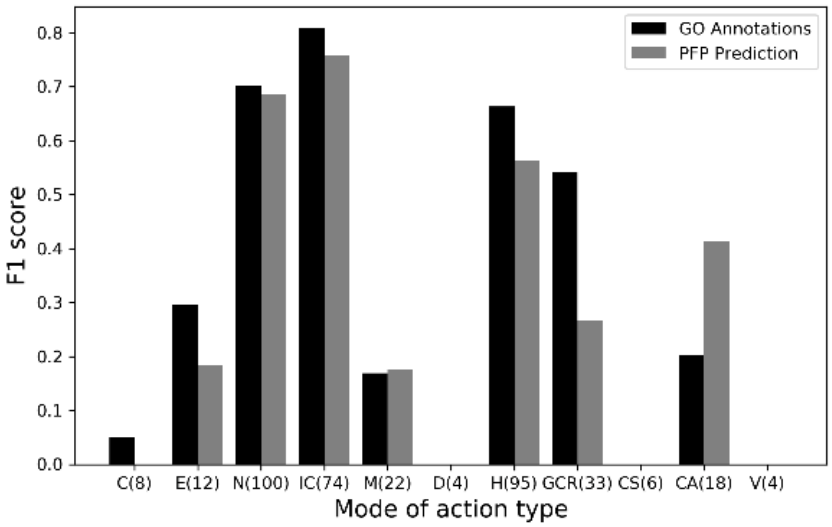


Figure 4.5. F1 scores of single-mode toxin proteins of 11different modes of action. 11 modes shown on the x-axis are: C, cardiotoxin; EN, enterotoxin; N, neurotoxin; IC, ion channel impairing toxin; M, myotoxin; D, dermonecrotic toxin; H, hemostasis impairing toxin; GCR, G-protein coupled receptor impairing toxin; CS, complement system impairing toxin; CA, cell adhesion impairing toxin; V, viral toxin. In the parentheses, the number of proteins of the mode is shown. 173 toxin proteins that have only one mode of action were analyzed. Black bars, predictions using GO annotations from UniProtKB; gray bars, predictions using PFP's GO term predictions.

Among the toxin protein dataset with the mode of action, there are 88 proteins that have two mode labels. Here we examine predictions made to the two largest toxin groups with two labels. 54 out of 88 proteins are labeled as neurotoxin (N) and Ion channel impairing toxin (IC). Out of them, 30 (55.6%) had the exactly correct predictions, i.e. correct positive predictions for the two labels and correct negative predictions to the other modes. For 9 other cases, the two labels, N and IC, were correctly predicted but with other false positive predictions. Finally, 48 of them (88.9%) had at least 1 mode, either N or IC, correctly predicted. The second-largest group with two modes were with hemostasis impairing toxin (H) and cell adhesion impairing toxin (CA), with 16 proteins. For this group, five of them have the exact correct prediction, and another protein was counted if we include the prediction with the two correct modes and one more over-predicted mode (37.5%). The number of proteins with at least one correctly predicted mode, H or CA, was 12 (75.0%). Thus, overall, NNTox (Mode of action) was able to capture the dual labels of the proteins reasonably well.

4.4 Discussion

Here, we developed NNTox, which predicts the toxicity of proteins via GO term annotation. In contrast to existing methods that compare a query protein sequence to known toxin proteins, NNTox's approach is less dependent on the known similar toxin proteins because prediction is made via GO terms. This approach exploits the success of general function predictors that have constantly been improving in the past years. We used PFP for the current development because it was developed by our lab and is one of the top-performing methods in the field. As the function prediction method improves, the toxin prediction by NNTox will also improve. Performance is also expected to improve by using additional input features, such as protein local structure information, e.g. protein main-chain conformation [132], which can be predicted with a stable accuracy.

The multi-label classification performed for toxin action mode prediction showed high elementwise accuracy (Table 4.3). Naturally, the accuracy for each mode was correlated to the data size of the category, which indicates that the architecture of the model is appropriate for this task and will further improve as more data become available.

In this work, we trained the network model so that the overall F1 score was maximized. The method can also be trained differently, for example, in a way to increase the sensitivity of toxin detection (allowing more false positives), considering that missing life-threatening toxins can cause a catastrophic outcome.

4.5 Availability of data and materials

The code and the dataset used in this study are made available at http://www.github.com/kiharalab/NNTox and http://kiharalab.org/nntox_dataset/

CHAPTER 5. CONCLUSION

5.1 Structure Prediction

In chapter 2 we presented AttentiveDist, a new method for prediction protein inter-residue distances/contact, orientation angles and sidechain center distances from sequence using deep learning. The predicted distances and angles were then used as constraints to model the protein structure. The predictive performance of deep learning-based models is known to depend on the size of input MSA. One way to incorporate more sequences in MSA is to relax the similarity cutoff by increasing the e-value. We showed that using multiple MSA of increasing e-values improves the contact prediction precision. To let the model focus on relevant information from different MSA's we added an attention mechanism. Using attention improved the performance compared to combination of individual MSA trained models. AttentiveDist outperformed the top CASP13 server models in free modelling targets, showing its better predictive power. Rigorous efforts have been made recently towards accurately predicting inter-residue distance/contact and the work in chapter 2 is another strong example that demonstrates the utility of modern deep learning technologies for improving protein structure prediction.

Homologous proteins with known 3D structure called templates can provide useful structural information when available. As a recent ongoing improvement to AttentiveDist, we added template information as features to the model. We used the first 5 templates found using HHsearch [133] regardless of their similarity to the query protein. From each template we extracted the protein structure information which includes Cβ-Cβ and side chain center distances, and template similarity measures which includes template coverage and sequence identity. An attention layer was added to extract useful information from different templates, similar to the attention mechanism for MSA's. We also updated the training dataset to include sequences up to 90% sequence similarity which increases the dataset size as well as provide small sequence-structural variations data to the model. AttentiveDist-Template improves the performance significantly, achieving long range L/1 precision of 0.564 on CASP13 FM and FM/TBM targets compared to 0.493 of AttentiveDist.

Further improvement is still needed especially for cases when the available sequences are sparse for input MSAs. Currently, similar sequences for MSA are found using sequence alignment-

based algorithms. To expand the similar sequences set, one can use protein sequence embeddings that are trained on millions of protein sequences using deep learning techniques like transformer model. Such embedding can capture semantic information and using sequences which are similar in embedding space can augment the current sequences in MSA. Another avenue to explore is how to convert MSA into features. PSSM and HMM profile summarize the MSA information, however, allowing the deep learning model to itself summarize the MSA could capture different importance patterns. Such approach might even reduce the need for a large sequence set in MSA.

In the recent CASP competition, Alphafold2 achieved a surprisingly high accuracy structure prediction. So far only general information about their methodology available. They mentioned that significant increase in performance is attributed to SE(3) transformer [26] allowing the deep learning model to output the 3D structure directly, instead of distances. This allows the model to optimize directly on the main goal. Another important contribution of their model is the use of attention layers where the information flow is dynamically controlled by the network instead of fixed local grids flow as seen in convolution layers. AlphaFold2 is trained on 128 TPUv3 GPU's for couple of weeks. This is a big resource limitation for many research labs. Future work should involve reducing the resource requirement by distilling the model using teacher student network as well as analyzing the performance of a shallow network.

The current structure prediction models are focused heavily on single chain prediction, however, several multi-chain proteins exist. Next steps for deep learning models would be to prediction interaction between different chains of a protein. Finally, the SE(3) transformer layers because of its ability to directly predict 3D structure can be applied to other structure related problems like protein docking and protein design.

5.2 Function Prediction

In chapter 3 we presented a new sequence-based protein function prediction method, Phylo-PFP, which substantially improved the prediction accuracy from its predecessor, PFP, by using phylogenetics to determine the evolutionary distance of sequences retrieved from database search. In this work we showed that database search scores and the phylogenetic distances are not strongly correlated, indicating that similar sequences retrieved through database searches may not be phylogenetically similar. It is known that phylogenetically close sequences that have evolved from same ancestor share functional similarity. In Phylo-PFP, we re-ranked sequences retrieved from

database search based on a distance-based phylogenetic analysis leading to substantial improvement in prediction accuracy. Contrasting to general function prediction, we also developed a specific function predictor NNTox, which uses GO terms to predict the toxicity of a protein sequence. Compared to previous approaches while rely on sequence similarity towards known toxin sequences, our proposed approach uses general function predictor that mines data from all possible sequences. For a deeper understand of toxicity, we predict toxin action mode using multi-label classification.

A future direction to improve the function prediction would be to use a more accurate tree construction technique such as maximum likelihood [102] or Bayesian inference [103, 104], instead of the distance-based phylogenetic analysis. Another important area to explore would be to explicitly considered functional domain [105] or residue information [35, 106] instead of the global sequence for functional transfer. Currently the same score is given to all GO terms of a similar sequence. Identifying domain-based functions would allow a more finetuned score given to different GO terms, leading to better precision. For NNTox, a future direction would be to incorporate structural information in the model. For instance protein local structure information, e.g. protein main-chain conformation [132], which can be predicted with a stable accuracy, can be added as input to the model.

APPENDIX A. SUPPLEMENTARY INFORMATION

Table A1. F1 score of GO term prediction by PFP on the non-redundant toxin dataset (488 toxin proteins).

PFP confidence cut-off	Precision	Recall	F1 score
0.0	0.282	0.562	0.376
0.1	0.602	0.522	0.560
0.2	0.699	0.497	0.580
0.3	0.741	0.476	0.580
0.4	0.765	0.459	0.575
0.5	0.786	0.446	0.570
0.6	0.806	0.436	0.566
0.7	0.821	0.410	0.547
0.8	0.831	0.393	0.533
0.9	0.849	0.361	0.507
1.0	0.860	0.293	0.436

Table A2. Association of GO terms with Toxin Keywords in UniProt. The file shows the toxin specificity, i.e. how much GO terms associate with toxin keywords of UniProt. The first and the second columns are the ID and the text description of GO terms, the toxin specificity (the third column) shows the fraction of proteins in UniProtKB-SwissProt that are toxins (i.e. with a keyword 'Toxin' UniProtKB KW-0800) among all the proteins in UniProtKB-SwissProt. The rest of the columns, Toxin Mode 1 to 3 show the dominant action mode(s) of the toxin if any that share above 10% of the toxin proteins.

GO ID	Function description	Toxin Specificity	Toxin Mode 1	Toxin Mode 2	Toxin Mode 3
GO:0035792	other organism postsynaptic membrane	554 (100.00%)	IC 443(80.0%)	N 554(100.0%)	
GO:0072556	other organism presynaptic membrane	317 (98.14%)	IC 172(53.0%)	N 317(98.0%)	
GO:0042151	nematocyst	252 (91.64%)	IC 164(60.0%)	N 143(52.0%)	
GO:0030550	acetylcholine receptor inhibitor activity	123 (91.11%)	IC 107(79.0%)	N 116(86.0%)	
GO:0019871	sodium channel inhibitor activity	169 (89.89%)	IC 165(88.0%)	N 89(47.0%)	
GO:0008200	ion channel inhibitor activity	1415 (87.89%)	IC 1123(70.0%)	N 1026(64.0%)	
GO:0016248	channel inhibitor activity	1415 (87.56%)	IC 1123(69.0%)	N 1026(63.0%)	
GO:0099602	neurotransmitter receptor regulator activity	123 (75.46%)	IC 107(66.0%)	N 116(71.0%)	
GO:0030548	acetylcholine receptor regulator activity	123 (75.46%)	IC 107(66.0%)	N 116(71.0%)	
GO:0070290	N-acylphosphatidylethanolamine-specific phospholipase D activity	214 (75.35%)	CS 141(50.0%)	D 147(52.0%)	
GO:0004630	phospholipase D activity	214 (75.09%)	CS 141(49.0%)	D 147(52.0%)	
GO:0016247	channel regulator activity	1415 (71.72%)	IC 1123(57.0%)	N 1026(52.0%)	
GO:0030547	receptor inhibitor activity	125 (69.44%)	IC 108(60.0%)	N 118(66.0%)	
GO:0009405	pathogenesis	6497 (66.26%)	IC 2427(25.0%)	N 2741(28.0%)	
GO:0102568	phospholipase A2 activity consuming 12-dioleoylphosphatidylethanolamine)	319 (59.51%)	N 126(24.0%)	H 134(25.0%)	
GO:0102567	phospholipase A2 activity (consuming 12-dipalmitoylphosphatidylcholine)	319 (59.51%)	N 126(24.0%)	H 134(25.0%)	
GO:1903963	arachidonate transport	342 (59.48%)	M 84(15.0%)	N 123(21.0%)	H 136(24.0%)
GO:0050482	arachidonic acid secretion	342 (59.48%)	M 84(15.0%)	N 123(21.0%)	H 136(24.0%)
GO:0017080	sodium channel regulator activity	172 (59.31%)	IC 168(58.0%)	N 92(32.0%)	
GO:0004623	phospholipase A2 activity	375 (58.41%)	M 88(14.0%)	N 140(22.0%)	H 146(23.0%)
GO:0044179	hemolysis in other organism	499 (57.82%)	CS 142(16.0%)	D 147(17.0%)	
GO:0032309	icosanoid secretion	342 (53.11%)	M 84(13.0%)	N 123(19.0%)	H 136(21.0%)
GO:0015909	long-chain fatty acid transport	342 (53.11%)	M 84(13.0%)	N 123(19.0%)	H 136(21.0%)
GO:0051715	cytolysis in other organism	508 (52.32%)	CS 142(15.0%)	D 147(15.0%)	
GO:0071715	icosanoid transport	342 (52.21%)	M 84(13.0%)	N 123(19.0%)	H 136(21.0%)
GO:0008191	metalloendopeptidase inhibitor activity	107 (46.72%)			
GO:0015908	fatty acid transport	342 (44.94%)	M 84(11.0%)	N 123(16.0%)	H 136(18.0%)
GO:0004620	phospholipase activity	605 (44.65%)	H 148(11.0%)	D 147(11.0%)	
GO:0042311	vasodilation	117 (40.62%)	GCR 62(22.0%)		
GO:0030545	receptor regulator activity	125 (37.43%)	IC 108(32.0%)	N 118(35.0%)	
GO:0016298	lipase activity	606 (36.53%)			
GO:0015718	monocarboxylic acid transport	342 (35.81%)	N 123(13.0%)	H 136(14.0%)	
GO:0019835	cytolysis	613 (33.46%)			

Table A2. Continued

GO ID	Function description	Toxin Specificity	Toxin Mode 1	Toxin Mode 2	Toxin Mode 3
GO:0044419	interspecies interaction between organisms	6497 (31.03%)	IC 2427(12.0%)	N 2741(13.0%)	
GO:0031640	killing of cells of other organism	540 (27.57%)			
GO:0044364	disruption of cells of other organism	545 (27.40%)			
GO:0044279	other organism membrane	1189 (24.62%)	IC 618(13.0%)	N 872(18.0%)	
GO:0044218	other organism cell membrane	1189 (24.62%)	IC 618(13.0%)	N 872(18.0%)	
GO:0001906	cell killing	541 (23.71%)			
GO:0008081	phosphoric diester hydrolase activity	230 (20.76%)	CS 142(13.0%)	D 148(13.0%)	
GO:0051704	multi-organism process	6497 (19.76%)			
GO:0008217	regulation of blood pressure	156 (17.91%)			
GO:0098772	molecular function regulator	1855 (17.50%)	IC 1341(13.0%)	N 1186(11.0%)	
GO:0050880	regulation of blood vessel size	133 (16.94%)			
GO:0044448	cell cortex part	252 (16.67%)	IC 164(11.0%)		
GO:0035150	regulation of tube size	133 (16.12%)			
GO:0005576	extracellular region	6233 (16.05%)			
GO:0003018	vascular process in circulatory system	133 (15.36%)			
GO:0006869	lipid transport	342 (14.87%)			
GO:0046942	carboxylic acid transport	342 (13.91%)			
GO:0010876	lipid localization	342 (13.44%)			
GO:0015711	organic anion transport	342 (12.87%)			
GO:0004866	endopeptidase inhibitor activity	265 (12.86%)			
GO:0061134	peptidase regulator activity	301 (12.85%)			
GO:0061135	endopeptidase regulator activity	265 (12.59%)			
GO:0030414	peptidase inhibitor activity	266 (12.35%)			
GO:0044217	other organism part	1216 (12.34%)			
GO:0044215	other organism	1216 (12.34%)			
GO:0016042	lipid catabolic process	381 (12.28%)			
GO:0044216	other organism cell	1202 (12.22%)			
GO:0035821	modification of morphology or physiology of other organism	579 (11.25%)			
GO:0005938	cell cortex	252 (10.99%)			
GO:0004867	serine-type endopeptidase inhibitor activity	153 (10.91%)			
GO:0008015	blood circulation	222 (10.58%)			
GO:0003013	circulatory system process	222 (10.47%)			
GO:0099568	cytoplasmic region	252 (9.91%)			
GO:0006820	anion transport	343 (9.71%)			
GO:0005509	calcium ion binding	383 (8.44%)			
GO:0004857	enzyme inhibitor activity	282 (7.94%)			
GO:0030435	sporulation resulting in formation of a cellular spore	104 (6.55%)			
GO:0052689	carboxylic ester hydrolase activity	380 (6.05%)			
GO:0004222	metalloendopeptidase activity	188 (5.84%)			

Table A2. Continued

GO ID	Function description	Toxin Specificity	Toxin Mode 1	Toxin Mode 2	Toxin Mode 3
GO:0042742	defense response to bacterium	214 (5.76%)			
GO:0043934	sporulation	104 (5.70%)			
GO:0046903	secretion	373 (5.43%)			
GO:0006952	defense response	722 (5.29%)			
GO:0090066	regulation of anatomical structure size	133 (5.12%)			
GO:0004252	serine-type endopeptidase activity	167 (4.95%)			
GO:0005179	hormone activity	111 (4.85%)			
GO:0008236	serine-type peptidase activity	205 (4.73%)			
GO:0017171	serine hydrolase activity	205 (4.71%)			
GO:0044764	multi-organism cellular process	567 (4.39%)			
GO:0009617	response to bacterium	215 (4.30%)			
GO:0030234	enzyme regulator activity	318 (4.15%)			
GO:0033644	host cell membrane	150 (4.05%)			
GO:0042578	phosphoric ester hydrolase activity	231 (4.01%)			
GO:0008237	metallopeptidase activity	235 (3.78%)			
GO:0004175	endopeptidase activity	356 (3.75%)			
GO:0098542	defense response to other organism	237 (3.25%)			
GO:0003008	system process	231 (3.12%)			
GO:0070011	peptidase activity acting on L-amino acid peptides	450 (2.98%)			
GO:0005102	signaling receptor binding	260 (2.96%)			
GO:0008233	peptidase activity	452 (2.91%)			
GO:0006644	phospholipid metabolic process	344 (2.85%)			
GO:0065008	regulation of biological quality	843 (2.63%)			
GO:0051707	response to other organism	238 (2.57%)			
GO:0043207	response to external biotic stimulus	238 (2.57%)			
GO:0009607	response to biotic stimulus	238 (2.47%)			
GO:0016788	hydrolase activity acting on ester bonds	632 (2.34%)			
GO:0006629	lipid metabolic process	588 (2.24%)			
GO:0006811	ion transport	418 (2.19%)			
GO:0018995	host	181 (2.08%)			
GO:0033643	host cell part	167 (1.94%)			
GO:0043657	host cell	167 (1.92%)			
GO:0006950	response to stress	768 (1.76%)			
GO:0044712	catabolic process	386 (1.68%)			
GO:0048646	anatomical structure formation involved in morphogenesis	112 (1.66%)			
GO:0033036	macromolecule localization	344 (1.53%)			
GO:0005575	cellular_component	6268 (1.53%)			
GO:0044255	cellular lipid metabolic process	345 (1.47%)			
GO:0008150	biological_process	6497 (1.47%)			

Table A2. Continued

GO ID	Function description	Toxin Specificity	Toxin Mode 1	Toxin Mode 2	Toxin Mode 3
GO:0071702	organic substance transport	344 (1.44%)			
GO:0044765	transport	387 (1.44%)			
GO:1902578	localization	387 (1.36%)			
GO:0009605	response to external stimulus	283 (1.31%)			
GO:0016787	hydrolase activity	1179 (1.28%)			
GO:0050896	response to stimulus	872 (1.19%)			
GO:0065007	biological regulation	1023 (1.15%)			
GO:1901575	organic substance catabolic process	390 (1.04%)			
GO:0044403	symbiont process	122 (1.04%)			
GO:0009056	catabolic process	395 (1.00%)			
GO:0006810	transport	472 (0.86%)			
GO:0051234	establishment of localization	472 (0.85%)			
GO:0032501	multicellular organismal process	292 (0.84%)			
GO:0046872	metal ion binding	989 (0.83%)			
GO:0043169	cation binding	989 (0.83%)			
GO:0048583	regulation of response to stimulus	134 (0.80%)			
GO:0043167	ion binding	989 (0.79%)			
GO:0051179	localization	475 (0.77%)			
GO:0003674	molecular_function	3520 (0.76%)			
GO:0005515	protein binding	284 (0.66%)			
GO:0030154	cell differentiation	113 (0.60%)			
GO:0019637	organophosphate metabolic process	348 (0.58%)			
GO:0009653	anatomical structure morphogenesis	112 (0.53%)			
GO:0048519	negative regulation of biological process	131 (0.52%)			
GO:0006796	phosphate-containing compound metabolic process	353 (0.51%)			
GO:0006793	phosphorus metabolic process	353 (0.50%)			
GO:0005488	binding	1362 (0.43%)			
GO:0003824	catalytic activity	1266 (0.43%)			
GO:0048869	cellular developmental process	113 (0.40%)			
GO:0044699	biological_process	1000 (0.40%)			
GO:0071944	cell periphery	295 (0.39%)			
GO:0044710	metabolic process	631 (0.38%)			
GO:0050789	regulation of biological process	287 (0.35%)			
GO:0009987	cellular process	1343 (0.34%)			
GO:0043232	intracellular non-membrane-bounded organelle	252 (0.33%)			
GO:0043228	non-membrane-bounded organelle	252 (0.33%)			
GO:0016020	membrane	394 (0.32%)			
GO:0048856	anatomical structure development	117 (0.31%)			
GO:0044767	developmental process	117 (0.30%)			

Table A2. Continued

GO ID	Function description	Toxin Specificity	Toxin Mode 1	Toxin Mode 2	Toxin Mode 3
GO:0044763	cellular process	671 (0.29%)			
GO:0032502	developmental process	117 (0.29%)			
GO:0031224	intrinsic component of membrane	233 (0.29%)			
GO:0016021	integral component of membrane	233 (0.29%)			
GO:0050794	regulation of cellular process	203 (0.26%)			
GO:0044425	membrane part	234 (0.25%)			
GO:0044238	primary metabolic process	740 (0.25%)			
GO:0071704	organic substance metabolic process	748 (0.22%)			
GO:0044444	cytoplasmic part	284 (0.21%)			
GO:0008152	metabolic process	752 (0.21%)			
GO:0043229	intracellular organelle	278 (0.18%)			
GO:0043226	organelle	279 (0.18%)			
GO:0044237	cellular metabolic process	455 (0.14%)			
GO:0019538	protein metabolic process	117 (0.11%)			
GO:0005737	cytoplasm	297 (0.11%)			
GO:0044424	intracellular part	298 (0.10%)			
GO:0005622	intracellular	298 (0.10%)			
GO:0044464	cell part	338 (0.09%)			
GO:0005623	cell	339 (0.09%)			
GO:0043170	macromolecule metabolic process	131 (0.07%)			
GO:0044260	cellular macromolecule metabolic process	102 (0.05%)			
GO:0006807	nitrogen compound metabolic process	102 (0.04%)			

The labels of the modes of toxin are:

C: Cardiotoxin

EN: Enterotoxin

N: Neurotoxin

IC: Ion channel impairing toxin

M: Myotoxin

D: Dermonecrotic toxin

H: Hemostasis impairing toxin

GCR: G-protein coupled receptor impairing toxin

CS: Complement system impairing toxin

CA: Cell adhesion impairing toxin

V: Viral exotoxin

Table A3. Results of the mode of action prediction for individual categories using UniProtKB GO annotations.

Mode of Action	Precision	Recall	F1 score	Total Number of Proteins
Cardiotoxin	0.031	0.125	0.050	8
Enterotoxin	0.267	0.334	0.296	12
Neurotoxin	0.736	0.670	0.702	100
Ion channel impairing toxin	0.819	0.797	0.808	74
Myotoxin	0.135	0.227	0.169	22
Dermonecrotic toxin	0	0	0	4
Hemostasis impairing toxin	0.774	0.579	0.663	95
G-protein coupled receptor impairing toxin	0.413	0.788	0.542	33
Complement system impairing toxin	0	0	0	6
Cell adhesion impairing toxin	0.146	0.334	0.203	18
Viral exotoxin	0	0	0	4

Table A4. Results of the mode of action prediction for individual categories. using PFP predictions.

Mode of Action	Precision	Recall	F1 score	Total Number of Proteins
Cardiotoxin	0	0	0	8
Enterotoxin	0.109	0.583	0.184	12
Neurotoxin	0.630	0.750	0.685	100
Ion channel impairing toxin	0.674	0.865	0.757	74
Myotoxin	0.109	0.455	0.175	22
Dermonecrotic toxin	0	0	0	4
Hemostasis impairing toxin	0.721	0.463	0.564	95
G-protein coupled receptor impairing toxin	0.184	0.485	0.267	33
Complement system impairing toxin	0	0	0	6
Cell adhesion impairing toxin	0.300	0.667	0.414	18
Viral exotoxin	0	0	0	4

REFERENCES

1. Benson, D.A., et al., *GenBank.* Nucleic acids research, 2012. **41**(D1): p. D36-D42.

2. Burley, S.K., et al., *RCSB Protein Data Bank: powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences.* Nucleic acids research, 2021. **49**(D1): p. D437-D451.

3. Bateman, A., et al., *UniProt: the universal protein knowledgebase in 2021.* Nucleic Acids Research, 2020.

4. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* Nucleic acids research, 1997. **25**(17): p. 3389-3402.

5. Steinegger, M., et al., *HH-suite3 for fast remote homology detection and deep protein annotation.* BMC bioinformatics, 2019. **20**(1): p. 1-15.

6. Morcos, F., et al., *Direct-coupling analysis of residue coevolution captures native contacts across many protein families.* Proc Natl Acad Sci U S A, 2011. **108**(49): p. E1293-301.

7. Seemayer, S., M. Gruber, and J. Soding, *CCMpred--fast and precise prediction of protein residue-residue contacts from correlated mutations.* Bioinformatics, 2014. **30**(21): p. 3128-30.

8. Jones, D.T., et al., *PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments.* Bioinformatics, 2012. **28**(2): p. 184-90.

9. Ovchinnikov, S., H. Kamisetty, and D. Baker, *Robust and accurate prediction of residue-residue interactions across protein interfaces using evolutionary information.* Elife, 2014. **3**: p. e02030.

10. Marks, D.S., et al., *Protein 3D structure computed from evolutionary sequence variation.* PLoS One, 2011. **6**(12): p. e28766.

11. Ekeberg, M., et al., *Improved contact prediction in proteins: using pseudolikelihoods to infer Potts models.* Phys Rev E Stat Nonlin Soft Matter Phys, 2013. **87**(1): p. 012707.

12. Abriata, L.A., G.E. Tamò, and M. Dal Peraro, *A further leap of improvement in tertiary structure prediction in CASP13 prompts new routes for future assessments.* Proteins: Structure, Function, and Bioinformatics, 2019. **87**(12): p. 1100-1112.

13. Senior, A.W., et al., *Improved protein structure prediction using potentials from deep learning.* Nature, 2020. **577**(7792): p. 706-710.

14. Xu, J. and S. Wang, *Analysis of distance-based protein structure prediction by deep learning in CASP13.* Proteins, 2019. **87**(12): p. 1069-1081.

15. Li, Y., et al., *Ensembling multiple raw coevolutionary features with deep residual neural networks for contact-map prediction in CASP13.* Proteins, 2019. **87**(12): p. 1082-1091.

16. Hanson, J., et al., *Accurate prediction of protein contact maps by coupling residual two-dimensional bidirectional long short-term memory with convolutional neural networks.* Bioinformatics, 2018. **34**(23): p. 4039-4045.

17. Zhang, C., et al., *DeepMSA: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins.* Bioinformatics, 2020. **36**(7): p. 2105-2112.

18. Xu, J., *Distance-based protein folding powered by deep learning.* Proc Natl Acad Sci U S A, 2019. **116**(34): p. 16856-16865.

19. Addou, S., et al., *Domain-Based and Family-Specific Sequence Identity Thresholds Increase the Levels of Reliable Protein Function Transfer.* Journal of Molecular Biology, 2009. **387**(2): p. 416-430.

20. Tian, W. and J. Skolnick, *How Well is Enzyme Function Conserved as a Function of Pairwise Sequence Identity?* Journal of Molecular Biology, 2003. **333**(4): p. 863-882.

21. Luong, M.-T., H. Pham, and C.D. Manning, *Effective approaches to attention-based neural machine translation.* arXiv preprint arXiv:1508.04025, 2015.

22. Vaswani, A., et al., *Attention is all you need.* Advances in neural information processing systems, 2017: p. 5998-6008.

23. Xu, K., et al., *Show, attend and tell: Neural image caption generation with visual attention.* International conference on machine learning, 2015: p. 2048-2057.

24. Ramachandran, P., et al., *Stand-alone self-attention in vision models.* arXiv preprint arXiv:1906.05909, 2019.

25. https://deepmind.com/blog/article/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology.

26. Fuchs, F.B., et al., *SE (3)-transformers: 3D roto-translation equivariant attention networks.* arXiv preprint arXiv:2006.10503, 2020.

27. Consortium, G.O., *Gene ontology consortium: going forward.* Nucleic acids research, 2015. **43**(D1): p. D1049-D1056.

28. Ding, Z., Q. Wei, and D. Kihara, *Computing and Visualizing Gene Function Similarity and Coherence with NaviGO,* in *Data Mining for Systems Biology*. 2018, Springer. p. 113-130.

29. Vazquez, A., et al., *Global protein function prediction from protein-protein interaction networks.* Nature biotechnology, 2003. **21**(6): p. 697-700.

30. Letovsky, S. and S. Kasif, *Predicting protein function from protein/protein interaction data: a probabilistic approach.* Bioinformatics, 2003. **19**(suppl_1): p. i197-i204.

31. Pal, D. and D. Eisenberg, *Inference of protein function from protein structure.* Structure, 2005. **13**(1): p. 121-130.

32. Pazos, F. and M.J. Sternberg, *Automated prediction of protein function and detection of functional sites from structure.* Proceedings of the National Academy of Sciences, 2004. **101**(41): p. 14754-14759.

33. Huttenhower, C., et al., *A scalable method for integration and functional analysis of multiple microarray datasets.* Bioinformatics, 2006. **22**(23): p. 2890-2897.

34. Wass, M.N. and M.J. Sternberg, *ConFunc—functional annotation in the twilight zone.* Bioinformatics, 2008. **24**(6): p. 798-806.

35. Gong, Q., W. Ning, and W. Tian, *GoFDR: a sequence alignment based method for predicting protein functions.* Methods, 2016. **93**: p. 3-14.

36. Chitale, M., et al., *ESG: extended similarity group method for automated protein function prediction.* Bioinformatics, 2009. **25**(14): p. 1739-1745.

37. Sahraeian, S.M., K.R. Luo, and S.E. Brenner, *SIFTER search: a web server for accurate phylogeny-based protein function prediction.* Nucleic acids research, 2015. **43**(W1): p. W141-W147.

38. Hawkins, T., S. Luban, and D. Kihara, *Enhanced automated function prediction using distantly related sequences and contextual association by PFP.* Protein Sci, 2006. **15**(6): p. 1550-1556.

39. Hawkins, T., et al., *PFP: Automated prediction of gene ontology functional annotations with confidence scores using protein sequence data.* Proteins, 2009. **74**(3): p. 566-82.

40. Jiang, Y., et al., *An expanded evaluation of protein function prediction methods shows an improvement in accuracy.* Genome biology, 2016. **17**(1): p. 1-19.

41. Lopez, G., et al., *Assessment of predictions submitted for the CASP7 function prediction category.* Proteins, 2007. **69**(S8): p. 165-174.

42. Kuhlman, B. and P. Bradley, *Advances in protein structure prediction and design.* Nat Rev Mol Cell Biol, 2019. **20**(11): p. 681-697.

43. Shin, W.-H., et al., *PL-PatchSurfer2: Improved Local Surface Matching-Based Virtual Screening Method That Is Tolerant to Target and Ligand Structure Variation.* Journal of Chemical Information and Modeling, 2016. **56**(9): p. 1676-1691.

44. Adhikari, B. and J. Cheng, *CONFOLD2: improved contact-driven ab initio protein structure modeling.* BMC Bioinformatics, 2018. **19**(1): p. 22.

45. Yang, J., et al., *Improved protein structure prediction using predicted interresidue orientations.* Proc Natl Acad Sci U S A, 2020. **117**(3): p. 1496-1503.

46. Chaudhury, S., S. Lyskov, and J.J. Gray, *PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta.* Bioinformatics, 2010. **26**(5): p. 689-691.

47. Wang, G. and R.L. Dunbrack, Jr., *PISCES: recent improvements to a PDB sequence culling server.* Nucleic Acids Res, 2005. **33**(Web Server issue): p. W94-8.

48. Steinegger, M., et al., *HH-suite3 for fast remote homology detection and deep protein annotation.* BMC Bioinformatics, 2019. **20**(1): p. 473.

49. Potter, S.C., et al., *HMMER web server: 2018 update.* Nucleic acids research, 2018. **46**(W1): p. W200-W204.

50. Mirdita, M., et al., *Uniclust databases of clustered and deeply annotated protein sequences and alignments.* Nucleic acids research, 2017. **45**(D1): p. D170-D176.

51. Suzek, B.E., et al., *UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches.* Bioinformatics, 2015. **31**(6): p. 926-932.

52. Steinegger, M. and J. Söding, *Clustering huge protein sequence sets in linear time.* Nature communications, 2018. **9**(1): p. 1-8.

53. Loshchilov, I. and F. Hutter, *Decoupled weight decay regularization.* arXiv preprint arXiv:1711.05101, 2017.

54. Christoffer, C., et al., *Performance and enhancement of the LZerD protein assembly pipeline in CAPRI 38-46.* Proteins, 2020. **88**(8): p. 948-961.

55. Zhou, H. and J. Skolnick, *GOAP: A Generalized Orientation-Dependent, All-Atom Statistical Potential for Protein Structure Prediction.* Biophysical Journal, 2011. **101**(8): p. 2043-2052.

56. Zhou, H. and Y. Zhou, *Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction.* Protein science, 2002. **11**(11): p. 2714-2726.

57. Huang, S.Y. and X. Zou, *An iterative knowledge-based scoring function to predict protein–ligand interactions: II. Validation of the scoring function.* Journal of computational chemistry, 2006. **27**(15): p. 1876-1882.

58. He, K., et al., *Deep residual learning for image recognition.* Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: p. 770-778.

59. Ulyanov, D., A. Vedaldi, and V. Lempitsky, *Instance normalization: The missing ingredient for fast stylization.* arXiv preprint arXiv:1607.08022, 2016.

60. Shah, A., et al., *Deep residual networks with exponential linear unit.* Proceedings of the Third International Symposium on Computer Vision and the Internet, 2016: p. 59-65.

61. Remmert, M., et al., *HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment.* Nature methods, 2012. **9**(2): p. 173-175.

62. Hanson, J., et al., *Improving prediction of protein secondary structure, backbone angles, solvent accessibility and contact numbers by using predicted contact maps and an ensemble of recurrent and residual convolutional neural networks.* Bioinformatics, 2019. **35**(14): p. 2403-2410.

63. Betancourt, M.R. and D. Thirumalai, *Pair potentials for protein folding: choice of reference states and sensitivity of predicted native states to variations in the interaction schemes.* Protein science, 1999. **8**(2): p. 361-369.

64. Xu, J., M. Mcpartlon, and J. Li, *Improved protein structure prediction by deep learning irrespective of co-evolution information.* bioRxiv, 2020.

65. Olechnovič, K., E. Kulberkytė, and Č. Venclovas, *CAD-score: A new contact area difference-based function for evaluation of protein structural models.* Proteins: Structure, Function, and Bioinformatics, 2013. **81**(1): p. 149-162.

66. https://predictioncenter.org/casp14/zscores_rrc.cgi.

67. Steinegger, M., M. Mirdita, and J. Söding, *Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold.* Nature methods, 2019. **16**(7): p. 603-606.

68. Nordberg, H., et al., *The genome portal of the Department of Energy Joint Genome Institute: 2014 updates.* Nucleic acids research, 2014. **42**(D1): p. D26-D31.

69. Mitchell, A.L., et al., *MGnify: the microbiome analysis resource in 2020.* Nucleic acids research, 2020. **48**(D1): p. D570-D578.

70. Mardis, E.R., *Next-generation sequencing platforms.* Annu Rev Anal Chem (Palo Alto Calif), 2013. **6**: p. 287-303.

71. Hawkins, T. and D. Kihara, *Function prediction of uncharacterized proteins.* J. Bioinform. Comput. Biol., 2007. **5**(1): p. 1-30.

72. Radivojac, P., et al., *A large-scale evaluation of computational protein function prediction.* Nat Methods, 2013. **10**(3): p. 221-7.

73. Jiang, Y., et al., *An expanded evaluation of protein function prediction methods shows an improvement in accuracy.* Genome Biol, 2016. **17**(1): p. 184.

74. Edgar, R.C., *MUSCLE: multiple sequence alignment with high accuracy and high throughput.* Nucleic Acids Res, 2004. **32**(5): p. 1792-1797.

75. Felsenstein, J., *Evolutionary trees from DNA sequences: a maximum likelihood approach.* J Mol Evol, 1981. **17**(6): p. 368-76.

76. Boutet, E., et al., *UniProtKB/Swiss-Prot, the Manually Annotated Section of the UniProt KnowledgeBase: How to Use the Entry View.* Methods Mol Biol, 2016. **1374**: p. 23-54.

77. Morgat, A., et al., *UniPathway: a resource for the exploration and annotation of metabolic pathways.* Nucleic Acids Res, 2012. **40**(Database issue): p. D761-9.

78. Haft, D.H., et al., *TIGRFAMs and Genome Properties in 2013.* Nucleic Acids Res, 2013. **41**(Database issue): p. D387-95.

79. Letunic, I. and P. Bork, *20 years of the SMART protein domain annotation resource.* Nucleic Acids Res, 2018. **46**(D1): p. D493-D496.

80. Fabregat, A., et al., *The Reactome Pathway Knowledgebase.* Nucleic Acids Res, 2018. **46**(D1): p. D649-D655.

81. Sigrist, C.J., et al., *New and continuing developments at PROSITE.* Nucleic Acids Res, 2013. **41**(Database issue): p. D344-7.

82. Bru, C., et al., *The ProDom database of protein domain families: more emphasis on 3D.* Nucleic Acids Res., 2005. **33**(Database issue): p. D212-D215.

83. Attwood, T.K., et al., *The PRINTS database: a fine-grained protein sequence annotation and analysis resource--its status in 2012.* Database (Oxford), 2012. **2012**: p. bas019.

84. Nikolskaya, A.N., et al., *PIRSF family classification system for protein functional and evolutionary analysis.* Evol Bioinform Online, 2007. **2**: p. 197-209.

85. Finn, R.D., et al., *The Pfam protein families database: towards a more sustainable future.* Nucleic Acids Res, 2016. **44**(D1): p. D279-85.

86. Finn, R.D., et al., *InterPro in 2017-beyond protein family and domain annotations.* Nucleic Acids Res, 2017. **45**(D1): p. D190-D199.

87. Pedruzzi, I., et al., *HAMAP in 2015: updates to the protein family classification and annotation system.* Nucleic Acids Res, 2015. **43**(Database issue): p. D1064-70.

88. Suzek, B.E., et al., *UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches.* Bioinformatics, 2015. **31**(6): p. 926-32.

89. Finn, R.D., et al., *The Pfam protein families database: towards a more sustainable future.* Nucleic acids research, 2016. **44**(D1): p. D279-D285.

90. Sahraeian, S.M., K.R. Luo, and S.E. Brenner, *SIFTER search: a web server for accurate phylogeny-based protein function prediction.* Nucleic Acids Res, 2015. **43**(W1): p. W141-7.

91. Schlicker, A., et al., *A new measure for functional similarity of gene products based on Gene Ontology.* BMC bioinformatics, 2006. **7**(1): p. 1-16.

92. Hawkins, T., M. Chitale, and D. Kihara, *Functional enrichment analyses and construction of functional similarity networks with high confidence function prediction by PFP.* BMC bioinformatics, 2010. **11**(1): p. 1-22.

93. Cantarel, B.L., H.G. Morrison, and W. Pearson, *Exploring the relationship between sequence similarity and accurate phylogenetic trees.* Mol Biol Evol, 2006. **23**(11): p. 2090-100.

94. Smith, S.A. and J.B. Pease, *Heterogeneous molecular processes among the causes of how sequence similarity scores can fail to recapitulate phylogeny.* Brief Bioinform, 2017. **18**(3): p. 451-457.

95. Eisen, J.A., *Phylogenomics: improving functional predictions for uncharacterized genes by evolutionary analysis.* Genome Res, 1998. **8**(3): p. 163-7.

96. Chitale, M., I.K. Khan, and D. Kihara, *In-depth performance evaluation of PFP and ESG sequence-based function prediction methods in CAFA 2011 experiment.* BMC Bioinformatics, 2013. **14 Suppl 3**: p. S2.

97. Remmert, M., et al., *HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment.* Nat Methods, 2012. **9**(2): p. 173-5.

98. Steinegger, M. and J. Soding, *MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets.* Nat Biotechnol, 2017. **35**(11): p. 1026-1028.

99. Consortium, G.O., *Gene Ontology Consortium: going forward.* Nucleic Acids Res, 2015. **43**(Database issue): p. D1049-56.

100. Khan, I.K., et al., *The PFP and ESG protein function prediction methods in 2014: effect of database updates and ensemble approaches.* GigaScience, 2015. **4**(1): p. s13742-015-0083-4.

101. Zhou, N., et al., *The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens.* Genome biology, 2019. **20**(1): p. 1-23.

102. Yang, Z., *PAML 4: phylogenetic analysis by maximum likelihood.* Mol Biol Evol, 2007. **24**(8): p. 1586-91.

103. Ronquist, F., et al., *MrBayes 3.2: efficient Bayesian phylogenetic inference and model choice across a large model space.* Syst Biol, 2012. **61**(3): p. 539-42.

104. Bouckaert, R., et al., *BEAST 2: a software platform for Bayesian evolutionary analysis.* PLoS Comput Biol, 2014. **10**(4): p. e1003537.

105. Messih, M.A., et al., *Protein domain recurrence and order can enhance prediction of protein functions.* Bioinformatics, 2012. **28**(18): p. i444-i450.

106. Wass, M.N. and M.J. Sternberg, *ConFunc - Functional Annotation in the Twilight Zone.* Bioinformatics, 2008. **24**(6): p. 798-806.

107. Ma, S., N. Tang, and J. Tian, *DNA synthesis, assembly and applications in synthetic biology.* Curr Opin Chem Biol, 2012. **16**(3-4): p. 260-7.

108. Hughes, R.A. and A.D. Ellington, *Synthetic DNA Synthesis and Assembly: Putting the Synthetic in Synthetic Biology.* Cold Spring Harb Perspect Biol, 2017. **9**(1).

109. Huang, P.S., S.E. Boyken, and D. Baker, *The coming of age of de novo protein design.* Nature, 2016. **537**(7620): p. 320-7.

110. Gupta, S.K. and P. Shukla, *Microbial platform technology for recombinant antibody fragment production: A review.* Crit Rev Microbiol, 2017. **43**(1): p. 31-42.

111. Borobova, E.A., et al., *Design of Artificial Immunogens Containing Melanoma-associated T-cell Epitopes.* Curr Gene Ther, 2018. **18**(6): p. 375-385.

112. Imran, M., et al., *Genetically transformed tobacco plants expressing synthetic EPSPS gene confer tolerance against glyphosate herbicide.* Physiol Mol Biol Plants, 2017. **23**(2): p. 453-460.

113. Berger, T., et al., *Toxins as biological weapons for terror-characteristics, challenges and medical countermeasures: a mini-review.* Disaster Mil Med, 2016. **2**: p. 7.

114. Taitt, C.R., et al., *Discrimination between biothreat agents and 'near neighbor' species using a resequencing array.* FEMS Immunol Med Microbiol, 2008. **54**(3): p. 356-64.

115. Duracova, M., et al., *Proteomic Methods of Detection and Quantification of Protein Toxins.* Toxins (Basel), 2018. **10**(3).

116. Walper, S.A., et al., *Detecting Biothreat Agents: From Current Diagnostics to Developing Sensor Technologies.* ACS Sens, 2018. **3**(10): p. 1894-2024.

117. Dunbar, J., et al., *Perspective on Improving Environmental Monitoring of Biothreats.* Front Bioeng Biotechnol, 2018. **6**: p. 147.

118. Rudraraju, S., T. Petrel, and O.P. Tabbaa, *ThreatSEQ Web Service, a Flexible Web-Deployed DNA Screening Platform for Wide-Spread and Cost-Effective Threat Detection and Interpretation.* ASM Biothreats, 2019.

119. Gupta, S., et al., *In silico approach for predicting toxicity of peptides and proteins.* PLoS One, 2013. **8**(9): p. e73957.

120. Agrawal, P., et al., *In Silico Approach for Prediction of Antifungal Peptides.* Front Microbiol, 2018. **9**: p. 323.

121. Saha, S. and G.P. Raghava, *BTXpred: prediction of bacterial toxins.* In Silico Biol, 2007. **7**(4-5): p. 405-12.

122. Naamati, G., M. Askenazi, and M. Linial, *ClanTox: a classifier of short animal toxins.* Nucleic Acids Res, 2009. **37**(Web Server issue): p. W363-8.

123. Gene Ontology Consortium, *Gene Ontology Consortium: going forward.* Nucleic Acids Res, 2015. **43**(Database issue): p. D1049-56.

124. Khan, I.K., et al., *PFP/ESG: automated protein function prediction servers enhanced with Gene Ontology visualization tool.* Bioinformatics, 2015. **31**(2): p. 271-2.

125. Jain, A. and D. Kihara, *Phylo-PFP: improved automated protein function prediction using phylogenetic distance of distantly related sequences.* Bioinformatics, 2019. **35**(5): p. 753-759.

126. Wei, Q., et al., *Using PFP and ESG Protein Function Prediction Web Servers.* Methods Mol Biol, 2017. **1611**: p. 1-14.

127. Khan, I.K., et al., *The PFP and ESG protein function prediction methods in 2014: effect of database updates and ensemble approaches.* Gigascience, 2015. **4**: p. 43.

128. UniProt Consortium, T., *UniProt: the universal protein knowledgebase.* Nucleic Acids Res, 2018. **46**(5): p. 2699.

129. Google Research. *Tensorflow*. 2019 [cited 2019; Available from: https://www.tensorflow.org/.

130. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* Nucleic Acids Res, 1997. **25**(17): p. 3389-3402.

131. Zhang, M.-L. and K. Zhang. *Multi-label learning by exploiting label dependency*. in *The 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2010. New York, NY.

132. Hanson, J., et al., *Improving Prediction of Protein Secondary Structure, Backbone Angles, Solvent Accessibility, and Contact Numbers by Using Predicted Contact Maps and an Ensemble of Recurrent and Residual Convolutional Neural Networks.* Bioinformatics, 2018: p. Epub.

133. Söding, J., *Protein homology detection by HMM–HMM comparison.* Bioinformatics, 2005. **21**(7): p. 951-960.

VITA

Aashish Jain

Education

Ph.D., Computer Science, 2021, Purdue University, West Lafayette, Indiana, USA

M.Sc., Computer Science, 2021, Purdue University, West Lafayette, Indiana, USA

M.Sc., Biotechnology, 2015, Indian Institute of Technology Roorkee, India

B.Sc., Microbiology, 2013, Delhi University, India

Awards

Best Poster Award, Sigma Xi 2019

PUBLICATIONS

1. **Jain, A.**, Terashi, G., Kagaya, Y., Subramaniya, S.R.M.V., Christoffer, C., and Kihara, D., Analyzing effect of quadruple multiple sequence alignments on deep learning based protein inter-residue distance prediction. Scientific Reports, 2021. 11(1): p. 1-13.

2. Maddhuri Venkata Subramaniya, S.R., Terashi, G., **Jain, A.**, Kagaya, Y. and Kihara, D., Protein contact map refinement for improving structure prediction using generative adversarial networks. Bioinformatics, 2021.

3. **Jain, A.** and Kihara, D., NNTox: gene ontology-based protein toxicity prediction using neural network. Scientific reports, 2019. 9(1): p. 1-10.

4. **Jain, A.** and Kihara, D., Phylo-PFP: improved automated protein function prediction using phylogenetic distance of distantly related sequences. Bioinformatics, 2019. 35(5): p. 753-759.

5. Khan, I.K., **Jain, A.**, Rawi, R., Bensmail, H. and Kihara, D., Prediction of protein group function by iterative classification on functional relevance network. Bioinformatics, 2019. 35(8): p. 1388-1394.

6. Macossay-Castillo, M., Marvelli, G., Guharoy, M., **Jain, A.**, Kihara, D., Tompa, P., and Wodak, S.J., The balancing act of intrinsically disordered proteins: enabling functional diversity while minimizing promiscuity. Journal of molecular biology, 2019. 431(8): p. 1650-1670.

7. Zhou, N., Jiang, Y., Bergquist, T.R., Lee, A.J., Kacsoh, B.Z., Crocker, A.W., . . ., **Jain, A.**, Kihara, D., . . ., Hamid, M.N., The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. Genome biology, 2019. 20(1): p. 1-23.

8. **Jain, A.**, Gali, H. and Kihara, D., Identification of moonlighting proteins in genomes using text mining techniques. Proteomics, 2018. 18(21-22): p. 1800083.