Graph Based Automatic Protein Function Annotation Improved by Semantic Similarity

Bishnu Sarker, Navya Khare, Marie-Dominique Devignes, Sabeur Aridhi

Graph Based Automatic Protein Function Annotation Improved By Semantic Similarity

Bishnu Sarker[1], Navya Khare[1,2], Marie-Dominique Devignes[1], and Sabeur Aridhi[1]

[1] University of Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
[2] IIIT, Hyderabad, India
{bishnu.sarker,navya.khare}@inria.fr
{marie-dominique.devignes,sabeur.aridhi}@loria.fr

Abstract. Functional annotation of protein is a very challenging task primarily because manual annotation requires a great amount of human efforts and still it's nearly impossible to keep pace with the exponentially growing number of protein sequences coming into the public databases, thanks to the high throughput sequencing technology. For example, the UniProt Knowledge-base (UniProtKB) is currently the largest and most comprehensive resource for protein sequence and annotation data. According to the November, 2019 release of UniProtKB, some 561,000 sequences are manually reviewed but over 150 million sequences lack reviewed functional annotations. Moreover, it is an expensive deal in terms of the cost it incurs and the time it takes. On the contrary, exploiting this huge quantity of data is important to understand life at the molecular level, and is central to understanding human disease processes and drug discovery. To be useful, protein sequences need to be annotated with functional properties such as Enzyme Commission (EC) numbers and Gene Ontology(GO) terms. The ability to automatically annotate protein sequences in UniProtKB/TrEMBL, the non-reviewed UniProt sequence repository, would represent a major step towards bridging the gap between annotated and un-annotated protein sequences. In this paper, we extend a neighborhood based network inference technique for automatic GO annotation using protein similarity graph built on protein domain and family information. The underlying philosophy of our approach assumes that proteins can be linked through the domains, families, and superfamilies that they share. We propose an efficient pruning and post-processing technique by integrating semantic similarity of GO terms. We show by empirical results that the proposed hierarchical post-processing potentially improves the performance of other GO annotation tools as well.

Keywords: Graph Mining · Bioinformatics · Knowledge Discovery · Protein Function Annotation · Network Inference · GrAPFI

1 Introduction

Thanks to advanced high throughput sequencing technologies, an ever growing number of protein sequences are accumulating into public databases [6].These

sequences should be annotated with functional properties to produce valuable information out of this huge quantity of data. Due to high costs and the time-consuming nature of manual function identification procedures, automated prediction of the functions of uncharacterized proteins is an important topic in the field of bioinformatics. This presents many challenges for biologists as well as computer scientists. A comprehensive knowledge about functional characteristics of proteins is central to understanding life at the molecular level, and is key to understanding human disease processes and drug discovery efforts [4]. The UniProt knowledgebase (UniProtKB), the most comprehensive publicly available protein database, consists of two parts: i) the UniProtKB/Swiss-Prot database contains the manually reviewed protein sequences [6, 12] and ii) the UniProtKB/TrEMBL database is used for storing un-reviewed protein sequences [12]. According to the November, 2019 release of UniProtKB, some 561,000 sequences have received manually reviewed functional annotations, whereas over 150 million protein sequences lack reviewed functional annotations. In UniProtKB, proteins information includes primary sequence as well as some other attributes such as the structural domains and family information. These attributes can be explored to compute pairwise similarity among proteins. However, in TrEMBL, proteins lack other important information such as their function and cellular localization in contrast to Swiss-Prot proteins that are often annotated with functional attributes. This huge quantity of TrEMBL proteins calls for efficient and rapid procedures to annotate them automatically.

Gene Ontology(GO)[3] terms are such functional attributes that indicate the functions that the proteins and genes are performing in our body. In gene ontology, GO terms are arranged hierarchically in three different directed acyclic graphs (DAG) namely 1) Biological Process (BP), 2) Molecular Function(MF), and 3) Cellular Component (CC). Every node in a DAG represents a GO term; and two connected GO terms are linked by different types of edges indicating different relationships. The most commonly used relationships are "is a", "part of", and "regulates".

At present, two complementary systems are in action for automatic annotation of UniProtKB/TrEMBL sequences: 1) UniRule [15] is a rule-based system that uses manually engineered rules to assign appropriate annotation. Although rules in UniRule are generally very reliable, designing rules is a laborious and time consuming process as well as it performs with low coverage. 2) SAAS(Statistical Automatic Annotation System) [23] reduces the manual labour in UniRule system by automatically generating rules using the annotations of the Swiss-Prot sequences and C4.5 decision tree algorithm [36]. Many other approaches exist for predicting protein functions leveraging various attributes. For example, structural similarity of proteins is used in [14, 33, 39, 46] for the purpose of function prediction. In [2, 26, 35, 38, 44, 47], authors exploit sequence similarities i.e. homology relationships to annotate un-reviewed sequences. Various machine learning based techniques are extensively studied in [7–9, 13, 19, 18, 27, 41, 28, 29, 33, 34, 43, 45, 52].

Recently, the notion of network science has attracted great attention across many scientific communities [5]. Network science has become a multidisciplinary area of research due to its ability to describe complex intertwined systems. It has found applications in many real-world scenarios from banking and the internet routing to modeling the human brain and understanding complex biological process. Several approaches for annotating protein function have used network science, particularly neighborhood based techniques for protein-to-protein propagation of functional information using protein-protein interaction (PPI) networks and Gene Ontology terms [42, 50, 17, 10, 32, 40]. The general belief is that interacting proteins share similar functional behaviours. A particular feature of biological networks is that they often require expert biological knowledge to fully understand and exploit the network.

We provide a brief overview of few state-of-the-art GO prediction tools that propose ensemble approaches and exploits different attributes such as sequence encoding, functional domain similarity and structural similarity, protein interaction network etc.

Blast2GO or B2G [11] is a sequence similarity-based functional annotation suite. By using BLAST [31] , B2G retrieves all GO annotations for the hit sequences, together with their evidence codes which is interpreted as an index of the trustworthiness of the GO annotation. To find specific annotations with certain level of reliability, an annotation score (AS) is computed for each candidate GO, which is composed of two additive terms. The first direct term (DT), represents the highest hit similarity of this GO weighted by a factor corresponding to its evidence code. The second term (AT) of the AS provides the possibility of abstraction. This term multiplies the number of total GOs unified at the node by a user defined GO weight factor that controls the possibility and strength of abstraction. Finally, lowest terms per branch that lies over a user defined threshold are selected. Once GO annotation is available through B2G, the application offers the possibility of direct statistical analysis on gene function information.

GoFDR [16] is a sequence alignment-based algorithm that runs BLAST[31] or PSI-BLAST[1], for a query protein, to obtain multiple sequence alignment (MSA) over the query sequence. It then identifies all GO terms associated with the sequences in MSA, and determines the functionally discriminating residues (FDRs) for each GO term. These FDRs are used to generate a position-specific scoring matrix (PSSM) which is then used to compute the score between the query protein and a GO term, followed by a raw score adjustment step to convert the raw score into a probability.

DeepGO [25] uses deep learning to learn features from protein sequences as well as from a cross-species protein–protein interaction (PPI) network. It utilizes the dependencies between GO classes as background information to construct a deep learning model. Input to the model is amino acid (AA) sequence of proteins in the form of trigrams, which is represented as one-hot encoding vectors followed by a dense embedding layer. An 1D convolution is applied over protein sequence data and redundant information from the resulting feature map is discarded through temporal max-pooling. In addition, DeepGO uses PPI networks

of multiple species, to generate knowledge graph embedding, which are with output of the max-pooling layer to form a combined feature vector. Finally, fully connected layers for each class in GO are used to create a hierarchical classification neural network model that encodes for transitivity of subclass relations. The main advantage of this approach is that it does not rely on manually crafted features, rather it is an entirely data-driven approach.

PANNZER [30, 22] uses weighted k-nearest neighbours approach with statistical testing to predict protein functional annotation. It starts with a sequence search against sequence database, to obtain a Sequence Similarity Result List (SSRL). To avoid biases towards large sequence families due to locally similar but globally dissimilar sequences, there is a limitation on number of sequences taken for analysis. Focus is only on the sequences that obtained strongest results from sequence scoring and hence apply pre-set filtering thresholds on alignment coverage, identity percentage, sequence length and informative descriptions. Non-linear weighting of taxonomic distances is another source of information used in PANNZER, corrected with a non-linear similarity function between the descriptions of compared query and target sequence. The second step of the PANNZER pipeline is to re-score the sequence hits using a sparse regression model that combines various signals from sequence alignment and non-linear taxonomic distance score and the weighted sum of score functions obtained is optimized against weighted similarity. In the final regression model all terms that had negative correlation with predicted variable from the model are excluded and final score is obtained.

A structure-based protein function annotation is proposed in COFACTOR [39, 48]. COFACTOR proposes a hybrid model combining information from structure and sequence homology, as well as protein–protein interaction (PPI) network for optimal protein function predictions focusing on three most widely-used and computationally amenable categories of function: Gene Ontology(GO) term, Enzyme Commission(EC) number and ligand-binding sites.

In this paper, we apply GrAPFI [40], a Graph-based Automatic Protein Function Inference approach for Gene Ontology(GO) annotation. We propose a pruning and hierarchical post-processing to eliminate the outlier annotations based on functional similarity discussed in GOGO[51]. More specifically, our contributions are the followings:

- We extend GrAPFI to perform Gene Ontology Prediction. GrAPFI is a neighborhood-based label propagation approach that works on a network of proteins connected using domains and family information. Originally proposed for Enzymatic protein function prediction using Enzyme Commission(EC) Number.
- We integrate semantic similarity to take into account the hierarchical nature of the Gene Ontology data and to prune outlier annotations based on their distance in semantic space. To find functional similarity, we used GOGO[51] which is claimed to be a fast and efficient way of computing GO term similarity.

– We experimentally evaluate the performance of the proposed approach by annotating protein sequences with Gene Ontology (GO) terms and report a comparative understanding of the efficacy of the proposed pruning technique for GO term prediction.

The remainder of the paper is organized as follows. Section 2 presents a brief overview of GrAPFI and describes the proposed pruning and hierarchical post-processing technique. In Section 3, we describe our experimental evaluation and we discuss our findings.

2 Method

A graph is a collection of objects denoted as $G = (V, E)$, where V is a set of vertices/nodes and $E \subseteq V \times V$ is a set of edges.

2.1 Function Annotation Using GrAPFI

GrAPFI [40] is a neighborhood-based label propagation approach that works on a network of proteins connected using domains and family information. GrAPFI follows the following steps to perform function annotation:

1. First, we construct a graph using the protein information. Each node u of the graph represents a protein. An edge (u, v) between two nodes/proteins u and v means that the linked proteins share some attributes like, domains and functional sites. A node u may have a set of labels $L(u)$ (one or more annotations to propagate), has a set of neighbors $N(u)$, and for every neighbor $v \in N(u)$, it has an associated weight $W_{u,v}$. Jaccard similarity is used to compute the link weight and computed as $W_{P1,P2} = \frac{|D1 \cap D2|}{|D1 \cup D2|}$ for two protein P1 and P2 having sets of domains $D1$ and $D2$, respectively.
2. Then,a label propagation approach is applied to the protein graph in order to infer functional properties of the unlabeled nodes. Given a query protein, based on the domains and family information of it contains, all the neighboring proteins and their annotations are retrieved from the weighted graph. After having the neighbors, each of the labels of the neighbors are weighted with edge-weights that these neighbors exhibit with this query protein. When retrieving neighbors, it is possible to select only those neighbors which meet a certain similarity threshold. That means that the links can be filtered based on a predefined cut-off weight. For each candidate annotation, GrAPFI provides a confidence score, namely model score (MS) that is computed as:

$$MS(u, i) = \frac{\sum_{v \in N(u)} W_{u,v} \delta(v^i, i)}{\sum_{v \in N(u)} W_{u,v}} \tag{1}$$

where $MS(u, i)$ is the weighted score of the candidate function i for the query protein u. And $\delta(v^i, i)$ is 1 if the function v^i of the protein v is the same as function i, otherwise, 0.

2.2 Pruning Prediction Set Using Functional Similarity Score

We observed that the state-of-the-art tools in the field of GO annotation [20, 37] yield a large number of predictions. Due to the large number of predicted annotations for each protein, precision of the model declines and leads to an increase in recall. But results from these approaches raise a big concern on false positives in the predictions. Therefore we need a method that increases the precision of the model, and hence decreases false positives in the predicted set.

To reduce the number of false positive annotations, we adopted an naive pruning technique by identifying and eliminating the outlier annotations using semantic similarity. Measuring semantic similarity between GO terms has always been an essential step in functional bioinformatics research. In a set of predicted GO annotations for a protein, pairwise semantic similarity between GO terms can show how closely these terms are related to each other and not just to the protein. We used an open-source tool called GOGO [51] for calculating the functional similarity score between GO terms and thus used it to compute the membership score of each predicted GO terms.

GOGO is relatively fast method which does not need to calculate the information content (IC) from a large gene annotation corpus and it considers the number of children nodes in the GO DAGs when calculating the semantic contribution of an ancestor node toward its descendent nodes. GOGO is based on GO DAG topology instead of IC which means that it is comparatively stable.

Given, $DAG_g = (g, T_g, E_g)$ be the Directed Acyclic GO Graph of a term g and it's ancestors T_g, the weight of semantic contribution is calculated as,

$$w_e(t) = 1/(c + nc(t)) + d \tag{2}$$

Where, c and d are constants, $nc(t)$ is the total number of children of the term $t \in T_g$. And E_g is the set of edges of the links among the terms in T_g. The semantic contribution of each term in $DAG_g = (g, T_g, E_g)$ is defined as,

$$S_g(t) = \begin{cases} 1 & if \;\; t = g \\ max\{w_c(t) * S_g(t') | t' \in children(t)\} & if \;\; t \neq g \end{cases} \tag{3}$$

Therefore, the aggregated semantic value for the term g is computed as,

$$SV(g) = \sum_{t \in T_g} S_g(t) \tag{4}$$

In the case of two terms where $DAG_{g1} = (g1, T_{g1}, E_{g1})$ of term g1 and $DAG_{g2} = (g2, T_{g2}, E_{g2})$ of term g2, the semantic similarity between them is as follows:

$$SS(g1, g2) = \frac{\sum_{t \in T_{g1} \cap T_{g2}} (S_{g1}(t) + S_{g2}(t))}{SV(g1) + SV(g2)} \tag{5}$$

Finally, the functional similarity between a set of GO terms, A= {g1, g1, g1,...,gm} and a query GO term $g \notin A$ and is as follows:

$$SS(g, A) = max_{1 \leq i \leq m}(SS(g, g_i \in A)) \tag{6}$$

Once the semantic similarity between each pair of GO terms in the predicted set is calculated, we measure the membership of each annotation in the set. $SS(g, A)$ can be used to find the membership score of a particular GO term in a set of predicted GO annotations. Equation 6 is reused to compute the membership score as follows:

$$SS(g_i, A) = max_{1 \leq j \leq m}(SS(g_i \in A, g_j \in A \setminus \{g_i\})) \tag{7}$$

Where, $SS(g_i, A)$ denotes the membership score of term g_i in a set of terms A.

Instead of maximum, membership score can also be calculated as the average and Root Mean Square(RMS) score of each annotation in the set. For this study, we used RMS score as it gave the best results. We name this measure of membership as semantic similarity (SS) score.

2.3 Aggregation of Scores

In all state-of-the-art GO annotation models, used for experiments in this study, there is a prediction score associated to each predicted annotation for every protein. We refer to this as model score(MS).

For a protein, u with a set of predicted annotations A, each annotation $g \in A$ has two scores associated to it: 1) first, the Model Score(MS) , defined as $MS(u, g)$, which shows the credibility with which the annotation was predicted by a particular annotation tool and 2) second, the Semantic Similarity(SS) score, defined as $SS_u(g, A)$, which shows the semantic similarity of each member annotation g to the predicted set A. Now, we need to combine these scores to find a combined prediction (CP) score, defined as $CP_u(g, A)$, for each annotation $g \in A$ of protein u. Joining the scores into a single score provides an overall assessment. A score should be able to distinguish between annotations that score average in both MS score and SS score, from those that score high in one scoring scheme and low in the other scheme. Therefore, instead of averaging the scores, we follow the following scheme:

$$CP_u(g, A) = \sqrt{\frac{(\frac{MS(u,g)}{max\_MS})^2 + (\frac{SS_u(g,A)}{max\_SS})^2}{2}} \tag{8}$$

Here, max_MS and max_SS denotes the maximal model score and semantic similarity score, respectively. Range for both the scores is from 0 to 1 and hence are bounded. Since this is a technique to prune an already predicted set, we take square root in the equation to increase the overall value of combined scores so as to increase threshold cutoff. Once we have the combined score, we can take a certain score as cutoff to filter the predicted set. Annotations with scores above the cutoff forms a new predicted set.

The final step of the process is hierarchical post-processing of predictions in the new predicted set. In the Gene Ontology DAGs, the GO terms holds different parent-child relations putting biologically closer GO terms hierarchically nearer in the graph. We implemented a methodology to include more reliable predictions by including the ancestors of target GO term in the new set of prediction. The ancestors of a GO term in the DAG that the term belongs to, have a very high semantic similarity with the term. Therefore, we first topologically sorted the DAG for each GO category and determined all possible paths from each GO term to the root of the corresponding category. Finally, we follow these paths from terms to the root, one by one and add corresponding ancestors to the set of predictions to obtain final prediction set.

3 Experiments and Result Analysis

3.1 Datasets

To experimentally validate the performance of the proposed technique, we have used a benchmark test set published in MetaGo[49]. For GrAPFI, we build the network using the training data from CAFA3[3]. CAFA3 is a well known competition that seek to annotate a list of protein sequences waiting for proper annotation. Along with target sequences, CAFA3 also published sequences that can be used as training data to develop model. In this study, to build the network, we have used CAFA3 training sequences and we have collected domain and family information for those proteins from UniprotKB. After that, we have built the graph of CAFA3 training proteins. This graph contains more than 65,000 nodes as proteins and an average 16 ground truth GO terms per protein. To prepare the test set, we have used MetaGO benchmark sequences and run InterProScan [21] to identify the domains and family information from sequence. Using the domains and family information of test proteins, we run GrAPFI and other annotation tools over all of the test proteins and annotated them with appropriate GO terms post-processed using proposed approach.

3.2 Result Analysis

In the Table 1, we show the annotation performance using standard evaluation measure namely precision, recall and F1 score. Among the top performing methods, only a few had their source code available to run experiments. Therefore, we focus on three easily available tools namely GrAPFI, PANNZER and DeepGOPlus[24], an improved version of DeepGO. DeepGoPlus learns models with less parameters than DeepGO. These tools are recently published and claimed to be high performing. We show that the semantic similarity improves the precision by many folds. However, the approach suffer from low recall as the number of predictions is much lower than the original predictions. This reduced number of predictions per protein essentially reduce the recall score and this ending up having lower f1-max score.

[3] https://www.biofunctionprediction.org/cafa/

Table 1. The experimental results for cases when 1) No-post-processing: without post-processing and pruning 2) SS-max: post-processed using highest semantic similarity score as cut-off, 3) SS-5: post-processed using 5th highest semantic similarity(SS) score as cut-off and 4) SS-5-MS-max/2: post-processed using 5th highest semantic similarity and (maximum model score)/2 as cut-off

Method	Post-processing cut-off	Precision	Recall	F1-max
GrAPFI	No-post-processing	0.165	0.108	0.107
	SS-max	**0.573**	0.115	0.175
	SS-5	0.445	0.380	0.376
	SS-5-MS-max/2	0.440	**0.391**	**0.379**
Pannzer	No-post-processing	0.547	**0.942**	0.668
	SS-max	**0.637**	0.225	0.301
	SS-5	0.634	0.515	0.536
	SS-5-MS-max/2	0.603	**0.689**	**0.609**
DeepGOPlus	No-post-processing	0.053	**0.653**	0.095
	SS-max	**0.249**	0.120	0.138
	SS-5	0.186	0.182	0.160
	SS-5-MS-max/2	0.167	0.233	**0.1725**

We run the above mentioned annotation tools on MetaGo benchmark data and obtain results of annotation prediction. These predicted sets are further pruned using semantic similarity and hierarchical post-processing and results are mentioned below (Table-1). Semantic similarity score and hierarchical post-processing score is obtained for each prediction for each protein. Different cut-offs of these two scores along with the score obtained from the model is used for analysis. For each annotation tool, Table:1 shows the annotation outcome in four cases: 1) without any kind of pruning and post processing, 2) when highest semantic similarity score is the cutoff, 3) when 5th highest semantic similarity score is the cutoff and 4) when 5th highest semantic similarity score and half of the maximum model score are the cutoff. From the table-1, it is evident that the proposed post-processing and pruning techniques that uses the semantic similarity of predicted GO terms improves the overal performance in most cases. In particular, it improves the precision by many folds. For example, the precision of GrAPFI is improved from 16.5% to 57.3% using maximum semantic similarity score as cut-off during post-processing. Similarly, the proposed combined scoring improves the precision of Pannzer and DeepGOPlus by many folds.

4 Conclusion

Automatic protein function annotation is an important topic in the field of bioinformatics because of the lack of annotation of proteins due to high costs and time-consuming nature of manual functional identification procedures. There are a number of tools exist to perform automatic protein function annotation

using GO terms, EC numbers, ligand binding sites etc. These tools use various attributes and different methods to accomplish the task. Although they show higher performance based on F1 score, the high F1 score is coming from a higher recall as they predict a large number of candidate annotations. This, in turn, increases the number of false positive annotations. In this paper, 1) we present a graph based protein function inference method extended for GO term prediction, and 2) we propose an efficient pruning and hierarchical post-processing technique by integrating semantic similarity of candidate annotations. We experimentally validate that the proposed method can significantly improve the annotation outcome. In fact, in most cases, recall is significantly low as the number of annotations is fewer compared to the number of annotations from predicted by other tools. Nevertheless, the precision is improved by many folds as we select the highly coherent semantically close annotations.

Acknowledgement

We would like to thank INRIA for providing the doctoral funding for Bishnu SARKER with CORDI-S grant and a internship funding to Navya Khare.

References

1. Altschul, S.F., et al.: Gapped blast and psi-blast: a new generation of protein database search programs. Nucleic Acids Research **25**(17), 3389–3402 (1997)
2. Arakaki, A.K., Huang, Y., Skolnick, J.: Eficaz 2: enzyme function inference by a combined approach enhanced by machine learning. BMC bioinformatics **10**(1), 107 (2009)
3. Ashburner, M., et al.: Gene ontology: tool for the unification of biology. Nature genetics **25**(1), 25 (2000)
4. Bakheet, T.M., Doig, A.J.: Properties and identification of human protein drug targets. Bioinformatics **25**(4), 451–457 (2009)
5. Barabási, A.L.: Linked: The new science of networks (2003)
6. Berger, B., Daniels, N.M., Yu, Y.W.: Computational biology in the 21st century: Scaling with compressive algorithms. Commun. ACM **59**(8), 72–80 (Jul 2016)
7. Cai, C., Han, L., Ji, Z.L., Chen, X., Chen, Y.Z.: Svm-prot: web-based support vector machine software for functional classification of a protein from its primary sequence. Nucleic acids research **31**(13), 3692–3697 (2003)
8. Cai, C., Han, L., Ji, Z., Chen, Y.: Enzyme family classification by support vector machines. Proteins: Structure, Function, and Bioinformatics **55**(1), 66–76 (2004)
9. Cai, Y.D., Chou, K.C.: Predicting enzyme subclass by functional domain composition and pseudo amino acid composition. Journal of Proteome Research **4**(3), 967–971 (2005)
10. Chua, H.N., Sung, W.K., Wong, L.: Exploiting indirect neighbours and topological weight to predict protein function from protein–protein interactions. Bioinformatics **22**(13), 1623–1630 (2006)
11. Conesa, A., Götz, S., García-Gómez, J.M., Terol, J., Talón, M., Robles, M.: Blast2go: a universal tool for annotation, visualization and analysis in functional genomics research. Bioinformatics **21**(18), 3674–3676 (2005)

12. Consortium, T.U.: Uniprot: a hub for protein information. Nucleic Acids Research **43**(D204-D212) (Jan 2015)
13. De Ferrari, L., Aitken, S., van Hemert, J., Goryanin, I.: Enzml: multi-label prediction of enzyme classes using interpro signatures. BMC Bioinformatics **13**(1), 61 (Apr 2012)
14. Dobson, P.D., Doig, A.J.: Predicting enzyme class from protein structure without alignments. Journal of molecular biology **345**(1), 187–199 (2005)
15. Gattiker, A., et al.: Automated annotation of microbial proteomes in SWISS-PROT. Computational Biology and Chemistry **27**(1), 49–58 (feb 2003)
16. Gong, Q., Ning, W., Tian, W.: Gofdr: a sequence alignment based method for predicting protein functions. Methods **93**, 3–14 (2016)
17. Hishigaki, H., et al.: Assessment of prediction accuracy of protein function from protein–protein interaction data. Yeast **18**(6), 523–531 (2001)
18. Huang, W.L., Chen, H.M., Hwang, S.F., Ho, S.Y.: Accurate prediction of enzyme subfamily class using an adaptive fuzzy k-nearest neighbor method. Biosystems **90**(2), 405–413 (2007)
19. des Jardins, M., Karp, P.D., Krummenacker, M., Lee, T.J., Ouzounis, C.A.: Prediction of enzyme classification from protein sequence without the use of sequence similarity. In: Proc Int Conf Intell Syst Mol Biol. vol. 5, pp. 92–99 (1997)
20. Jiang, Y., et al.: An expanded evaluation of protein function prediction methods shows an improvement in accuracy. Genome biology **17**(1), 184 (2016)
21. Jones, P., et al.: Interproscan 5: genome-scale protein function classification. Bioinformatics **30**(9), 1236–1240 (2014)
22. Koskinen, P., Törönen, P., Nokso-Koivisto, J., Holm, L.: Pannzer: high-throughput functional annotation of uncharacterized proteins in an error-prone environment. Bioinformatics **31**(10), 1544–1552 (2015)
23. Kretschmann, E., Fleischmann, W., Apweiler, R.: Automatic rule generation for protein annotation with the c4.5 data mining algorithm applied on swiss-prot. Bioinformatics **17 10**, 920–6 (2001)
24. Kulmanov, M., Hoehndorf, R.: Deepgoplus: improved protein function prediction from sequence. Bioinformatics **36**(2), 422–429 (2020)
25. Kulmanov, M., Khan, M.A., Hoehndorf, R.: Deepgo: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. Bioinformatics **34**(4), 660–668 (2017)
26. Kumar, N., Skolnick, J.: Eficaz2. 5: application of a high-precision enzyme function predictor to 396 proteomes. Bioinformatics **28**(20), 2687–2688 (2012)
27. Li, Yu, e.a.: Deepre: sequence-based enzyme ec number prediction by deep learning. Bioinformatics **34**(5), 760–769 (2018)
28. Li, Y.H., et al.: Svm-prot 2016: a web-server for machine learning prediction of protein functional families from sequence irrespective of similarity. PloS one **11**(8), e0155290 (2016)
29. Lu, L., Qian, Z., Cai, Y.D., Li, Y.: Ecs: an automatic enzyme classifier based on functional domain composition. Computational biology and chemistry **31**(3), 226–232 (2007)
30. Medlar, A.J., Törönen, P., Zosa, E., Holm, L.: Pannzer 2: Annotate a complete proteome in minutes! Nucl. Acids Res **43**, W24–W29 (2018)
31. Mount, D.W.: Using the basic local alignment search tool (blast). Cold Spring Harbor Protocols **2007**(7), pdb–top17 (2007)
32. Nabieva, E., et al.: Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. Bioinformatics **21**(suppl_1), i302–i310 (2005)

33. Nagao, C., Nagano, N., Mizuguchi, K.: Prediction of detailed enzyme functions and identification of specificity determining residues by random forests. PloS one **9**(1), e84623 (2014)
34. Nasibov, E., Kandemir-Cavas, C.: Efficiency analysis of knn and minimum distance-based classifiers in enzyme family prediction. Computational biology and chemistry **33**(6), 461–464 (2009)
35. Quester, S., Schomburg, D.: Enzymedetector: an integrated enzyme function prediction tool and database. BMC bioinformatics **12**(1), 376 (2011)
36. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (Mar 1986)
37. Radivojac, P., et al.: A large-scale evaluation of computational protein function prediction. Nature methods **10**(3), 221 (2013)
38. Rahman, S.A., et al.: Ec-blast: a tool to automatically search and compare enzyme reactions. Nature methods **11**(2), 171 (2014)
39. Roy, A., Yang, J., Zhang, Y.: Cofactor: an accurate comparative algorithm for structure-based protein function annotation. Nucleic acids research **40**(W1), W471–W477 (2012)
40. Sarker, B., Ritchie, D.W., Aridhi, S.: Exploiting complex protein domain networks for protein function annotation. pp. 598–610. In: Proc. of 7th International Conference on Complex Networks and Their Applications, Cambridge, UK (Dec 2018)
41. Sarker., B., Ritchie., D.W., Aridhi., S.: Functional annotation of proteins using domain embedding based sequence classification. pp. 163–170 (2019)
42. Schwikowski, B., Uetz, P., Fields, S.: A network of protein–protein interactions in yeast. Nature biotechnology **18**(12), 1257 (2000)
43. Shen, H.B., Chou, K.C.: Ezypred: a top–down approach for predicting enzyme functional classes and subclasses. Biochemical and biophysical research communications **364**(1), 53–59 (2007)
44. Tian, W., Arakaki, A.K., Skolnick, J.: Eficaz: a comprehensive approach for accurate genome-scale enzyme function inference. Nucleic acids research **32**(21), 6226–6239 (2004)
45. Volpato, V., Adelfio, A., Pollastri, G.: Accurate prediction of protein enzymatic class by n-to-1 neural networks. BMC bioinformatics **14**(1), S11 (2013)
46. Yang, J., et al.: The i-tasser suite: protein structure and function prediction. Nature methods **12**(1), 7 (2015)
47. Yu, C., Zavaljevski, N., Desai, V., Reifman, J.: Genome-wide enzyme annotation with precision control: Catalytic families (catfam) databases. Proteins: Structure, Function, and Bioinformatics **74**(2), 449–460 (2009)
48. Zhang, C., Freddolino, P.L., Zhang, Y.: Cofactor: improved protein function prediction by combining structure, sequence and protein–protein interaction information. Nucleic Acids Research **45**(W1), W291–W299 (2017)
49. Zhang, C., Zheng, W., Freddolino, P.L., Zhang, Y.: Metago: Predicting gene ontology of non-homologous proteins through low-resolution protein structure prediction and protein–protein network mapping. Journal of molecular biology **430**(15), 2256–2265 (2018)
50. Zhao, B., et al.: An efficient method for protein function annotation based on multilayer protein networks. Human genomics **10**(1), 33 (2016)
51. Zhao, C., Wang, Z.: Gogo: An improved algorithm to measure the semantic similarity between gene ontology terms. Scientific reports **8**(1), 15107 (2018)
52. Zhou, N., et al.: The cafa challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. bioRxiv p. 653105 (2019)