Protein Function Prediction for newly sequenced organisms

Mateo Torres (mateo.torres@fgv.br)[1,*],
Haixuan Yang (haixuan.yang@nuigalway.ie)[2,*],
Alfonso E. Romero (aeromero@cs.rhul.ac.uk)[3,*],
Alberto Paccanaro * (alberto.paccanaro@rhul.ac.uk)[1,3]

1: Escola de Matemática Aplicada, Fundação Getúlio Vargas, Rio de Janeiro, Brazil
2: School of Mathematics and Statistical Sciences, National University of Ireland Galway, Galway, Ireland
3: Department of Computer Science, Centre for Systems and Synthetic Biology, Royal Holloway, University of London, Egham Hill, Egham, UK
*: These authors contributed equally to this work.

Abstract

Recent successes in protein function prediction have shown the superiority of approaches that integrate multiple types of experimental evidence over methods that rely solely on homology. However, newly sequenced organisms continue to represent a difficult challenge, because only their protein sequences are available and they lack data derived from large scale experiments.

We introduce S2F (Sequence to Function), a network propagation approach for the functional annotation of newly sequenced organisms. Our main idea is to systematically transfer functionally relevant data from model organisms to newly sequenced ones, thus allowing us to use a label propagation approach. S2F introduces a novel label diffusion algorithm that can account for the presence of overlapping communities of proteins with related functions. Since most newly sequenced organisms are bacteria, we tested our approach in the context of bacterial genomes. Our extensive evaluation shows a great

improvement over existing sequence-based methods, as well as four state-of-the-art general-purpose protein function prediction methods.

Our work demonstrates that employing a diffusion process over networks of transferred functional data is an effective way to improve predictions over simple homology. S2F is applicable to any type of newly sequenced organism as well as to those for which experimental evidence is available. A free, easy to run version of S2F is available at https://www.paccanarolab.org/s2f.

Introduction

Less than 1% of the available protein sequences are currently annotated with reliable information and the gap between unannotated and annotated sequences is widening at an unprecedented rate[1] (Supplementary Note 1). Traditional experimental approaches to determine protein function are usually expensive, time consuming, and provide low throughput. While higher throughput approaches have recently been developed, they are also proving to be insufficient to cope with the sheer number of new sequences produced by next generation sequencing techniques[2]. In this context, the computational annotation of protein function has become a crucial step for a better understanding of the complex mechanisms of living cells.

Newly sequenced organisms represent a particularly difficult challenge for automated annotation methods because only their protein sequences are available and, in general, we lack any other data derived from large scale functional experiments. In fact, protein function prediction is somewhat easier for more studied organisms, including model organisms, where multiple types of functional experimental evidence (e.g., gene expression, proteomics data) are available that can be integrated with sequence information. The Critical Assessment of Functional Annotation Challenge (CAFA)[3] has indeed shown that advanced methods that integrate multiple types of information for the prediction of Gene Ontology (GO)[4] terms significantly outperform methods that use only sequence information.

Network propagation approaches have been shown to be among the most successful methods to predict protein function when some sort of experimental evidence is available[5].

These methods combine and amplify existing knowledge about the function of some of the proteins by propagating it through networks where nodes represent proteins, and edges represent pairwise functional relationships between them that are derived from experiments (e.g., physical interaction, co-occurrence in protein complexes, co-expression). In other words, these methods expand an initial set of functional labels available for some experimentally characterised proteins (seeds) to related neighbouring proteins, thus exploiting the guilt-by-association principle, according to which highly connected nodes should share similar functional properties. However, until now, these ideas could not be applied to newly sequenced organisms, since in this case both the seeds and the networks are unavailable.

This paper introduces S2F (sequence to function), a novel network propagation-based method for the functional annotation of newly sequenced organisms. Our main idea is to systematically transfer functionally relevant data that is available for model organisms to newly sequenced organisms, thus allowing us to use network propagation to predict protein function. S2F presents a novel network propagation algorithm that can account for the presence of overlapping communities of proteins with related functions.

Since most newly sequenced organisms are bacteria, we have developed and tested our solutions in the context of bacterial genomes. Bacteria is also the superkingdom with most available sequenced proteins in UniProtKB (Supplementary Note 1), and the functional characterisation of bacteria holds great potential in fields ranging from alternative energy sources to understanding and treating disease. However, the ideas presented here are more widely applicable to protein function prediction for any type of organism, and an earlier version of our algorithm has successfully been applied to organisms from other kingdoms[3,6].

Results

The aim of S2F is to predict the function of each of the proteins in a newly sequenced organism. Functional categories are defined according to the Gene Ontology (GO)[4], where terms are organised in a hierarchical structure with several domains and levels of specificity. The prediction of protein function is a multi-class, multi-label classification problem: multi-class, as there are over 40,000 possible GO terms that can be annotated to a protein; multi-label, because each protein can be annotated with multiple GO terms. Importantly, the hierarchical structure of the Gene Ontology must be taken into account for the prediction, since whenever a protein is annotated with a GO term, it is also annotated with all its ancestor terms up to the root of the ontology (this is known as the "true path rule"[7,8]). Therefore, an important requirement for the output of any protein function prediction method is to be *consistent*: if a GO term is predicted with a certain probability, its parent terms must be predicted with an equal or greater probability[9].

S2F consists of four main components (see the pictorial representation in Fig. 1):

A. a method to *infer the initial seeds*, that combines the output of InterPro[10] and HMMER[11] to obtain a set of initial predictions that is consistent;
B. a method for *network transfer*, that relies on the concept of interolog[12,13] to infer several functional networks;
C. a method for *network combination,* that combines the different functional networks into a single one;
D. a *label propagation* algorithm, that diffuses the seed information to obtain a prediction.

In the following, we will describe each component in turn. We will assume that we wish to predict the function for a newly sequenced organism (target organism) with n proteins, and that the Gene Ontology contains t terms.

A. S2F Seed Inference InterPro[10] constitutes an excellent starting point for predicting protein function from sequence as it provides predictions from 14 different protein signature databases. We consolidate its output into an $n \times t$ matrix of predictions R (see Materials and Methods) which is consistent, and where each entry R_{ij} is the fraction of InterPro models in which the (i, j) association is present.

Although InterPro predictions are extremely accurate, they are often limited in number and involve only a few GO terms. In order to enrich the catalogue of GO terms that appear in our initial seed set, HMMER[11] is run for every protein in the target organism against the experimentally annotated sequences in UniProtKB/Swiss-Prot (Supplementary Note 2). This results in the HMMER seed set, a binary matrix H of size $(n \times t)$, which is then up-propagated according to the true path rule[7,8]. A convex combination of H and R gives us the consistent combined seed set $Y \in \mathbb{R}^{n \times t}$:

$$Y = \alpha R + (1 - \alpha) H$$

where $\alpha \in \mathbb{R}, 0 \leq \alpha \leq 1$ controls the relative contribution of InterPro and HMMER predictions, and each entry of $Y, 0 \leq Y_{ij} \leq 1$.

B. S2F Network Transfer We build networks where nodes represent target organism proteins, and edges represent pairwise functional relationships (interactions) between them. Since experimental evidence of functional relationships between proteins is not

available for newly sequenced organisms, in order to create these networks we exploit the fact that these relationships are often conserved across species[14,15]. This allows us to transfer existing evidence from well-studied organisms to newly sequenced ones.

Our starting point is the seminal work by Yu et al.[13] who transferred different types of functional networks with high precision using the concept of interolog-mapping first proposed by Walhout et al.[12]. The idea is that, given two proteins A and B in the target organism, if there exists a pair of proteins A' and B' that are known to interact in another organism (source organism), such that A is an orthologue of A', and B is an orthologue of B', then we can infer an interaction between A and B.

Our transfer algorithm derives from the one proposed by Yu et al.[13] (for details see Materials and Methods and Supplementary Note 3). S2F uses STRING[16] as the dataset of different types of experimental interactions in source organisms. For each type of interaction, S2F builds one transferred network, r, that can be represented as a matrix $W^{(r)} \in \mathbb{R}^{n \times n}$, where each entry $W_{ij}^{(r)}$represents the strength of the interaction between proteins i and j in r. For a given target organism, S2F transfers five types of interaction, namely "neighborhood", "experiments", "coexpression", "textmining", and "database" using the experimental interactions available for any organism in STRING.

C. S2F Network Combination Having obtained a set of transferred networks for the target organism, we now face the task of combining them into a single network for diffusing the seeds. Our approach is to linearly combine the different networks through *learned* coefficients. These coefficients provide us with interesting information about the relative importance and role of each network in the prediction. While other systems learn this

combination (e.g., GeneMANIA[17]), the solution we propose here is applicable to our problem, where no initial set of known labels is available.

We begin by using the InterPro predictions to build a network of functional similarities $T \in \mathbb{R}^{n \times n}$, where the similarity between proteins i and j, T_{ij}, is defined as:

$$T_{ij} = \frac{|N_i \cap N_j|}{|N_i \cup N_j|}$$

where N_i and N_j are the sets of all GO terms above a threshold τ that are associated to proteins i and j respectively in R, that is, $N_i = \{k | R_{ik} > \tau\}$, and $N_j = \{k | R_{jk} > \tau\}$. Therefore, T_{ij} is the Jaccard similarity between sets of GO terms that are assigned by InterPro to proteins i and j.

Given p networks $W^{(r)}$ with $r \in \{1, \ldots, p\}$, we combine them into a single network $W \in \mathbb{R}^{n \times n}$ using a weighted linear combination, where the vector of weights $\hat{\boldsymbol{c}} \in \mathbb{R}^p$ is learnt by minimising the square of the difference between T and the linear combination (see Materials and Methods).

D. S2F label propagation Proteins rarely perform their functions in isolation, but rather they act as part of functional groups. As mentioned earlier, network propagation methods for protein function prediction exploit exactly this fact – groups of proteins that are highly connected in functional networks form communities that share a similar function. Importantly, when a protein has more than one function, it will belong to more than one of such functional groups. We notice that such proteins, lying at the intersection of

communities are, in general, more functionally similar compared to their neighbours, since they share more functional roles. Therefore, when a set of proteins has more than one function, the propagation of information (or diffusion) between proteins within this set should be higher than the diffusion between proteins in this set and proteins outside this set. However, this does not happen with existing diffusion methods (for details see Supplementary Note 6). Here we propose a novel label propagation method that explicitly models overlapping communities and, in this way, corrects this problem.

We begin by defining the matrix $W^{S2F} \in \mathbb{R}^{n \times n}$, a transformation of the combined network W whose entry W_{ij}^{S2F} is defined as:

$$W_{ij}^{S2F} = \frac{1}{2}\left(\frac{1}{d_i} + \frac{1}{d_j}\right) J_{ij} W_{ij}$$

where $d_i = \sum_{j=1}^{n} J_{ij} W_{ij}$ and J is a weighted Jaccard similarity matrix that models the overlapping community effect (see Materials and Methods). We also define a diagonal matrix D^{S2F} where the i-th diagonal element $D_{ii}^{S2F} = \sum_j W_{ij}^{S2F}$. Our algorithm produces a prediction matrix $F \in \mathbb{R}^{n \times t}$ for all the n proteins of the organism and all the t GO terms by computing the following:

$$F = (I + \lambda L)^{-1} Y$$

where Y is the matrix containing the initial labelling, I is the identity matrix, $L = D^{S2F} - W^{S2F}$ is the Laplacian of W^{S2F}, and $\lambda > 0$ is the regularisation parameter (see Materials and Methods).

We show that this label propagation algorithm does not suffer from the problem described above for overlapping communities (see Supplementary Note 6). Moreover, we prove that it satisfies the necessary conditions to ensure that, for each pair of terms j and k such that j is an ancestor of k (in these cases $Y_{ij} \geq Y_{ik}$ for every i), we have that $F_{ij} \geq F_{ik}$ for every i (the proof can be found in Supplementary Note 7). As a consequence, since Y is consistent with the Gene Ontology structure, F will also be consistent.

Experimental Setup

We present the evaluation of S2F on bacteria from UniProtKB. Following the evaluation procedure used by most authors[3,18] the performance of S2F in predicting protein function was assessed both in a *per-gene* and in a *per-term* setting. In per-gene predictions, given a gene, we assess the performance of S2F at predicting a set of functions associated to that gene. Conversely, in per-term predictions, given a function, we assess the performance of S2F at predicting a set of genes that perform that function.

The performance was assessed against a set of known experimental annotations. Therefore, the bacteria used for testing were chosen so that they had at least a few experimentally annotated genes (to be able to assess the performance in a per-gene setting) while maintaining a reasonable diversity of annotated GO terms (to be able to assess the performance in a per-term setting) in the GOA database[19]. The ten bacteria in Table 1 satisfied our set of criteria (the criteria are detailed in Materials and Methods).

This set of bacteria provides a good testbed for our experiments. The amount of experimental annotations in these bacteria covers a wide spectrum, ranging from well-

studied bacteria (e.g., *E. coli*) to more obscure ones that are not even included in STRING (e.g., *Brucella abortus*).

In our experiments, we tested the performance at predicting the functional annotation for the whole genome for each of the ten bacteria, in turn. To avoid circular reasonings, when testing each bacterium, we carefully removed any functional information for that bacterium as well as for any phylogenetically close species. To do this, for each bacterium, we created a list of excluded species in two steps. First, starting from that bacterium, we navigated the NCBI taxonomy moving up two levels (i.e., to the parent of the parent node) and we included in our list that node and all its descendants. Second, we added to the list all the nodes in the NCBI taxonomy that had a similar name. Having created a list of excluded species, we removed any information about these species from STRING, as well as about their proteins from the GOA database. The detailed list of all organisms excluded when testing each specific bacterium is provided in the Supplementary Data.

Predicted annotations were evaluated against the existing functional annotations (GOA files in Supplementary Data) using the well-established metrics that have been used in the CAFA challenge[3]: $F_{\max}$, $S_{\min}$, AUC-ROC, and AUC-PR metrics (for details, see Supplementary Note 12).

Evaluation

We compared the performance of S2F against InterPro, HMMER, Argot 2.5[20], DeepGOPlus[21], GOLabeler[22] and NetGO[23]. InterPro and HMMER are among the best and most widely used sequence-based methods for predicting protein function for newly sequenced organisms. The other four methods, although they were not explicitly conceived for this problem, could

nevertheless be employed here as they are able to predict protein function using sequence information alone. Argot 2.5[20], and GOLabeler[22] were among the top performer in the last edition of the CAFA competition[6]; NetGO[23] and DeepGOPlus[21] were introduced after the last CAFA competition, and they were shown to perform very well against top CAFA algorithms. (For details of the implementation, parameter settings and a description of these algorithms see Materials and Methods and Supplementary Notes 14, 16 and 17).

Figures 2-5 show the AUC-ROC, AUC-PR, $F_{\max}$ and $S_{\min}$ evaluated *per-gene* and the *per-term* for S2F and each competitor algorithm. (An interactive version of these results is also available in the result explorer on our website: https://www.paccanarolab.org/s2f). We can see that S2F outperformed the other methods according to the vast majority of the performance measures for the ten bacteria – it is surpassed only in 4 out of the 80 bacteria-measure combinations, most often on the AUC-ROC measure. In order to better appreciate the increase in performance offered by S2F, we also explicitly report the percentage of improvement of S2F vs each competitor for each of the 10 organisms (see Supplementary Figures 53-59 in Supplementary Note 15).

Analysing these results, we see that, as expected, the accuracy of the S2F predictions does depend on the accuracy of InterPro and HMMER, that provide the initial seeds for the diffusion process of S2F. An interesting question is whether the improved performance of S2F is merely due to the fact that it combines the labels of InterPro and HMMER, or whether the diffusion of these labels through the transferred networks has a role in its performance. For this reason, we also report in the figures the performance of the linear combination of InterPro and HMMER labels that we used as seeds for the diffusion process in S2F (matrix Y). We can see that, with the only exception of the AUC-ROC for *Brucella Abortus,* S2F

shows an improvement when compared with the simple linear combination of the InterPro and HMMER outputs. This means that S2F is able to effectively combine the information of these labels together with the evolutionary information contained in the interolog graphs.

As we mentioned earlier, by integrating InterPro and HMMER we aimed at obtaining seeds that combined the high accuracy and specificity offered by InterPro with the high coverage provided by HMMER. To check whether our linear combination, controlled by the parameter α achieved this, we analysed how the different setting of α affected the S2F results (details of the experiments are described in Supplementary Note 13). Supplementary Figures 48-51 show that, in general, a combination of InterPro and HMMER seeds ($0 < \alpha < 1$) gives much better results in terms of S2F performance than when using only seeds from either of them ($\alpha = 0$ or $\alpha = 1$). However, just looking at S2F performance, it is unclear how to set the value of α, as there is disagreement among different performance measures and organisms. At the same time, an important objective in real-world scenarios is to predict, for a given gene, a small set of terms that are highly accurate while being as specific as possible. Therefore, we analysed the information content of the top genes predicted by S2F for different values of α (see Supplementary Figure 52). Our results show that, in this scenario, high values of α (e.g., $\alpha = 0.9$) should be preferred.

We also evaluated the predictions obtained by diffusing the outputs of InterPro and HMMER, separately, on the interolog network W. Supplementary Figures 11-14 (Supplementary Note 8, also available in the interactive data explorer on our website https://www.paccanarolab.org/s2f) show how our diffusion process is able to improve the labels obtained by InterPro (or HMMER). This means that our diffusion on combined

interolog networks is an effective way to improve protein function prediction over simpler homology methods.

Our diffusion method was motivated by our desire to model the presence of overlapping communities in functional networks. It is unclear how to quantify exactly the number of proteins being shared across communities, as this is obscured by the relationships among functional labels as well as the noise and incompleteness of available annotations. However, the semantic similarity of proteins with known function can provide some insight, as we can quantify the correlation between the graph onto which we diffuse, W^{S2F}, and a graph of semantic similarities among functionally annotated proteins, G^{SS}. Supplementary Figure 17 shows the values of these correlations for each of the ten bacteria and compares them with correlations between G^{SS} and W^{GM}, the graph used by GeneMANIA[17], a diffusion-based method for protein function prediction in model organisms that does not explicitly model overlapping communities (for details of these experiments see Supplementary Note 6). We can see that W^{S2F} shows higher correlation with the semantic similarity graph G^{SS} in the great majority of the cases, for different organisms and across different GO ontologies.

Finally, to further demonstrate how S2F can facilitate biological research by generating feasible hypothesis, we performed a prospective evaluation. We deployed S2F to make predictions using only data available up to December 2014 and we assessed its accuracy on proteins that were experimentally annotated between 2015 and 2021. The experiments are detailed in Supplementary Note 11. Supplementary Figures 44-47 show that while the performance of InterPro is relatively stable, for some bacteria the overall performance of HMMER (and, as a consequence, of the InterPro + HMMER combination) seems to worsen greatly. As expected, the performance of S2F decreases in these cases, but overall the

327 diffusion process is able to alleviate the effect and compensate for the lower quality of the

328 seeds.

329

Discussion

The difficulty of protein function prediction, one of the most important problems in computational biology, varies greatly, depending on how much experimental information is available for the organism under investigation. Predictions for well-studied organisms can rely on multiple types of functional experimental evidence (e.g., gene expression, proteomics data) that can be represented in the form of graphs. For these organisms, network propagation approaches that amplify existing knowledge about the function of some of the proteins have been shown to be very effective[5,17,24,25].

This paper introduces S2F, a method that applies a network propagation algorithm to organisms for which only sequence information is available. The main idea is to create networks of interologs by systematically transferring functional data that is available for model organisms, and to use these networks to combine and amplify a few preliminary GO labels (seeds) obtained through homology or identifiable protein features.

Our work shows that employing a diffusion process over networks of interologs is an effective way to improve predictions over simple homology. The improvement comes from combining information: S2F effectively integrates homology information and identifiable protein features (preliminary GO labels from HMMER and InterPro) together with evolutionary information contained in the interolog graphs, through a diffusion process. S2F includes a novel network propagation algorithm that can account for the presence of overlapping communities of nodes with related functions.

Ultimately, the accuracy of S2F when predicting function for a specific organism will depend on several factors, including the specificity and diversity of the preliminary GO labels, and the density of the interolog networks, which in turn depends on the evolutionary distance from organisms with existing functional experimental evidence. When predicting a GO term for a specific gene, these factors affect how many neighbours that gene has, how many of these genes have preliminary GO labels, and how accurate these labels are. These factors are highly interleaved, and it is difficult to quantify the effect of each one individually. For example, it would seem reasonable to expect that S2F would generate better predictions for more highly connected nodes. We tested this hypothesis by measuring the correlation between node degrees and the performance measures for the bacteria in this study. However, our results show that the correlation was either weak and negative, or not statistically significant (see Supplementary Note 10, as well as Supplementary Figures 26-36).

The different interolog networks that we combine are extremely sparse with virtually no overlap among them (see Supplementary Figures 3 and 4). In this scenario, in terms of prediction performance, different combination methods would give results that are as good as the simple average of the networks (Supplementary Figures 5-8 compare our combination strategy, the network combination used by STRING[16], and the simple average). However, our approach allows the linear combination of the different networks through learned coefficients, which provides us with information about the relative importance and role of each network in the prediction (see Supplementary Note 4). Our combination method is similar to the one used in GeneMANIA, but it allows us to learn these linear weights without relying on an initial set of known functional labels.

We note that the removal of functional information regarding each bacterium and its phylogenetically close species, makes this problem much harder than the one tested in the regular CAFA competition settings. For this reason, the performances for Argot 2.5[20], DeepGOPlus[21], GOLabeler[22], and NetGO[23] seem generally lower than those reported earlier. Also, methods that are able to integrate global and local information seem to perform better than local methods in our setting. This can be seen by comparing the results obtained by S2F and the Consistency Method[26] – another method that integrates global information – with the results obtained by NetGO, where the use of network information is limited locally to nodes that are just one link away from the query node. A performance comparison between our label propagation method and the Consistency Method is available in Supplementary Note 9.

In this paper we have focused and presented results for bacteria, but S2F can be applied to any organism, independently on how well functionally characterised it is. An earlier version of S2F which is optimised to use existing functional evidence for target organisms was submitted to the CAFA2 challenge[3], where it ranked as one of the top performing methods.

The code for S2F is freely available at https://www.paccanarolab.org/s2f. The S2F software is fast, robust, and easy to setup and run. The software is fully documented, including a wiki with instructions for common use cases, instructions on how to use S2F to predict function for newly sequenced bacteria and details on how to replicate all our results, together with the necessary input data (see Supplementary Data).

Materials and Methods

S2F Seed Inference

InterPro produces m binary matrices of predictions $R^{(k)}$, each of size $(n \times t)$ (here $k \in \{1, \ldots, m\}$ and $m \leq 14$ is the number of models for which InterPro gives at least one prediction for the target organism). To combine these matrices while ensuring that the combination is consistent with the hierarchical structure of GO, we first up-propagate these associations according to the true path rule[7,8] considering both the "is_a", and "part_of" relations. Each matrix $R^{(k)}$ is up-propagated separately, and therefore any convex combination of the up-propagated matrices will be consistent. We combine them to obtain a consistent InterPro seed set $R \in \mathbb{R}^{n \times t}$ where each entry of R, R_{ij}, is defined as:

$$R_{ij} = \frac{\sum_{k=1}^{m} R_{ij}^{k}}{m}$$

S2F Network Transfer

STRING[16] is a database that compiles several 3,123,056,667 interactions between proteins in 5,090 organisms. Interactions are divided into 7 types: "neighborhood", "fusion", "co-occurrence", "experiments", "co-expression", "textmining", and "database". Each interaction is annotated with a score that ranges from 0 to 1 representing the confidence that STRING assigns for the two proteins to be functionally related.

In our transfer procedure, two proteins A and A' are considered to be orthologues if three conditions are met:

- they are BLAST mutual best hits, with both e-values smaller than 1e-6;
- percent identity is greater than 80% – this is to avoid transference between multi-domain proteins with different domain architecture;
- their "joint identity" (geometric mean of the two percent identities) is above 60% – Yu et al.[13] showed that this condition achieves almost perfect accuracy at identifying interacting orthologues.

When the same interaction can be transferred from multiple organisms, only the one with the highest "joint identity" is kept. The pseudocode of the algorithm for building a collection of transferred networks for the target organism is provided in Supplementary Algorithm 1 (Supplementary Note 3). S2F only considers networks with at least 3 edges, that is, for every interaction type in STRING, we consider the transferred network r only if $W^{(r)}$ contains at least 3 values.

Finally, a homology network is added to the collection of interolog networks to increase the combined network connectivity and facilitate the diffusion process. The homology network $W^{(h)}$ is defined as the negative log of the BLAST e-value for every pair of proteins.

S2F Network combination

Given p networks $W^{(r)}$ with $r \in \{1, \ldots, p\}$, we combine them into a single network W using a weighted linear combination. The vector of weights $\hat{\boldsymbol{c}} \in \mathbb{R}^p$, and bias $\hat{b}$ are learnt by minimising:

$$(\hat{\boldsymbol{c}}, \hat{b}) = \underset{\boldsymbol{c},b}{argmin} \sum_{i,j} \left(b + \sum_{r=1}^{p} c_r W_{ij}^{(r)} - T_{ij} \right)^2$$

This linear regression can be solved efficiently, and we can interpret each learnt coefficient c_r as representing how much each network r contributes to the combination. An analysis on these coefficients is provided in the Supplementary Note 4.

S2F label propagation

The weighted Jaccard coefficient matrix J is defined elementwise as:

$$J_{ij} = \frac{\sum_k W_{ik} W_{jk}}{\sum_k W_{ik} + \sum_k W_{jk} - \sum_k W_{ik} W_{jk}}$$

.

Thus, the element J_{ij} relates to how much elements i and j belong to the same community in network W. For a given term k, we learn the k-th column of matrix F, that we denote by F_k, by minimising the cost function $\mathcal{Q}(F_k)$:

$$\mathcal{Q}(F_k) = \sum_{i=1}^{n} (F_{ik} - Y_{ik})^2 + \frac{\lambda}{2} \sum_{i=1}^{n} \frac{1}{d_i} \sum_{j=1}^{n} J_{ij} W_{ij} \left(F_{ik} - F_{jk}\right)^2$$

Similarly to the cost function used by the Consistency Method (CM)[26], ours is the sum of two terms. The role of the first term is to conserve the initial labels Y_{ik} – this term is minimised when the node labels F_{ik} are the same as the initial labels. The second term accounts for the consistency of the labels of adjacent nodes (reflecting the guilt-by-association principle) – this term is minimised when adjacent nodes have similar labels (i.e. the difference between F_{ik} and F_{jk} becomes small). Note that the importance of the difference between F_{ik} and F_{jk} is proportional to $J_{ij}W_{ij}$ which models the community effect — the more i and j are connected through their neighbours, the greater their contribution to the cost function. Furthermore, notice that:

$$\frac{1}{d_i} = \frac{1}{\sum_j J_{ij}W_{ij}}$$

is a normalisation factor that gives to each protein in the network similar ability to influence its neighbours, independently of its degree.

The closed-form solution that minimises $Q(F_k)$ is:

$$F_k^* = (I + \lambda L)^{-1}\mathrm{Y_k}$$

where Y_k is the initial labelling, $L = D^{S2F} - W^{S2F}$ is the Laplacian of W^{S2F}, whose entry W_{ij}^{S2F} is defined as:

$$W_{ij}^{S2F} = \frac{1}{2}\left(\frac{1}{d_i} + \frac{1}{d_j}\right)J_{ij}W_{ij},$$

and D^{S2F} is a diagonal matrix where the i-th diagonal element is $D_{ii}^{S2F} = \sum_j W_{ij}^{S2F}$.

Bacteria selection criteria and Datasets

The criteria we used for selecting bacteria were:

- The bacteria must have at least 10 functional annotations with an experimental or curated GO evidence code (EXP, IDA, IPI, IMP, IGI, IEP, TAS, or IC) in the GOA database[19].
- The bacteria must have at least 8 terms annotated with at least 3 genes after up-propagation, for each GO subdomain – biological process (BP), molecular function (MF), and cellular component (CC).

In our experiments, we used STRING version 11.0. All sequences in FASTA format were downloaded from UniProtKB/Swiss-Prot using the taxonomy identifiers listed in Table 1. The GO annotations were downloaded from the GOA database[19]. All datasets were downloaded in April 2020. We used HMMER version 3.1b2, InterProScan version 5.42-78.0, and blastp from BLAST 2.6.0+.

Competitor algorithms

In all our experiments, in order to simulate a real case scenario for the problem of predicting function in newly sequenced organisms, for each bacterium, we removed any functional information regarding that bacterium as well as any functional information about species that are phylogenetically close (the list of all organisms excluded is provided in the Supplementary Data).

GOLabeler[22] and its successor, NetGO[23], are only offered as web services which use all the data available from their sources (namely GOA, STRING, UniProtKB, InterPro) for their prediction. Therefore, the results for NetGO and GOLabeler presented here were obtained running our own implementation of these systems that had been trained using datasets from which all the aforementioned functional information had been removed. All the parameters of the component models as well as the learning to rank ensemble were set using the default values suggested by the authors[22,23]. A detailed description on how to prepare the input data and how to use our implementation of these methods is available in Supplementary Note 17.

Argot 2.5[20] was run on its web server (http://www.medcomp.medicina.unipd.it/Argot2-5/). For each bacterium, we first used BLAST and HMMER to obtain alignments between its proteins and a version of UniProtKB from which the sequences of excluded organisms (for that bacterium) were omitted. These alignments were then submitted to the Argot 2.5 web server.

DeepGOPlus[21] was run using the code from the latest stable version available (1.0.1). In order to remove the information from phylogenetically close organisms, we added some pre-processing steps to the input files and small corrections were made to the prediction script. A detailed guide on how to setup and run the pre-processing and prediction is described in Supplementary Note 16.

InterPro was run using InterProScan version 5.42-78.0, the output file was then processed to extract the predictions that included GO terms.

HMMER version 3.1b2 was run against a GO annotation file that was pre-processed to keep only the experimental or curated evidence codes (EXP, IDA, IPI, IMP, IGI, IEP, TAS, or IC). The output file was post-processed to remove any alignment that came from an organism that had been excluded in the prediction.

Acknowledgements

The first idea for this project was conceived in discussions with Tara Gianoulis. We remember Tara dearly for her intelligence, kindness, enthusiasm and passion for research. The authors also thank Prajwal Bhat, Tamás Nepusz, Juan Caceres, Marco Frasca, Giorgio Valentini, Alessandra Devoto, Laszlo Bögre, Rajkumar Sasidharan, and Mark Gerstein for many important and stimulating discussions.

A.P. was supported by Biotechnology and Biological Sciences Research Council (https://bbsrc.ukri.org/) grants BB/K004131/1, BB/F00964X/1 and BB/M025047/1; Medical Research Council (https://mrc.ukri.uk) grant MR/T001070/1; Consejo Nacional de Ciencia y Tecnología Paraguay (https://www.conacyt.gov.py/) grants 14-INV-088 and PINV15-315; National Science Foundation Advances in Bio Informatics (https://www.nsf.gov/) grant 1660648; Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro grant E-26/201.079/2021 (260380); and Fundação Getulio Vargas.

Author Contributions

A.P. conceived the study. A.P. and H.Y. devised the algorithms, developed the prototype and performed preliminary evaluations. M.T. and A.E.R. implemented and extended the algorithms and evaluation metrics, performed large scale experiments and analysed the results. A.P., M.T., A.E.R. wrote the manuscript and evaluated the biological relevance of the results. All the authors discussed the results and implications. A.P. supervised the project.

Competing Interests

The authors declare no competing interests.

Data Availability

The input sequence files[27] in FASTA format for all the organisms used in this paper, are available at: https://doi.org/10.5281/zenodo.5514323. The same URL also contains the detailed list of all organisms excluded when testing each specific bacterium.

Code Availability

The code for S2F is freely available and maintained at https://www.paccanarolab.org/s2f. The exact version[28] used for this publication is available at:

https://doi.org/10.5281/zenodo.5513071

Tables

Table 1 List of bacteria that satisfies our selection criteria with number of genes and annotations. The number of terms with more than 3 annotations in each of the GO domains (BP = biological process, MF = molecular function, CC = cellular component) is calculated after up-propagation, and therefore may be larger than the number of experimentally annotated genes.

NCBI ID	Name	Genes	Experimentally annotated genes	BP terms with > 3 annotations	MF terms with > 3 annotations	CC terms with > 3 annotations
272624	*Legionella pneumophila subsp. Pneumophila Philadelphia 1*	2,076	18	30	8	8
223283	*Pseudomonas syringae pv. tomato*	5,055	25	48	32	15
359391	*Brucella abortus*	2,229	26	17	8	14
99287	*Salmonella typhimurium*	3,764	116	183	46	24
198628	*Dickeya dadantii*	3,411	102	214	21	13
1111708	*Synechocystis* sp.	2,442	137	101	21	30
224308	*Bacillus subtilis*	3,410	375	301	120	24
208964	*Pseudomonas aeruginosa*	4,487	947	695	222	42
83332	*Mycobacterium tuberculosis*	3,284	1,027	797	280	45
83333	*Escherichia coli*	3,906	3,350	1,546	706	134

578

Figure Legends/Captions

Figure 1 Overview of the S2F approach The set of n protein sequences of the target organism (shown in red) constitutes the input to the system; t is the total number of GO terms to be predicted. External datasets (STRING [16], GOA [19], and UniProtKB [29]) are shown in orange. **Seed Inference.** Running HMMER on the input sequences against experimentally annotated sequences from UniProtKB/Swiss-Prot we obtain an $(n \times t)$ matrix H of predictions (the HMMER seed set). Running InterPro we obtain m matrices of predictions $R^{(m)}$, one per InterPro model, each of size $(n \times t)$. These matrices are then combined into a single $(n \times t)$ matrix R (the InterPro seed set). The combined seed set Y, that will be used for the label propagation, is a linear combination of H and R. **Network Transfer**. A collection of networks is built by our interaction transfer procedure using known functional relationships between proteins in every organism from the STRING database. **Network Combination.** Transferred networks are linearly combined into a single network W. The weights of the linear combination are learnt using an auxiliary target network built from R. **Prediction**. The network W and the seed set Y are fed into our label propagation algorithm that outputs the protein function prediction F, an $(n \times t)$ matrix where each row corresponds to a protein, each column corresponds to a GO term and each entry $F_{i,j}$ is related to the probability for protein i to have function j. For a given protein i, its labels $F_{i,\cdot}$ are guaranteed to be consistent, i.e., they satisfy the GO "true path rule".

Figure 2 S_{min} metric for every organism per-gene (left) and per-term (right), lower values are better. Comparison of HMMER, InterPro, HMMER + InterPro, S2F, Argot 2.5, DeepGOPlus, GOLabeler, and NetGO. Values indicate the mean of the metric over genes or terms, and error bars indicate a confidence interval of 95%, estimated using 10,000 bootstrap iterations on the gene set or term set, respectively.

Figure 3 F_{max} for every organism per-gene (left) and per-term (right), higher values are better. Comparison of HMMER, InterPro, HMMER + InterPro, S2F, Argot 2.5, DeepGOPlus, GOLabeler, and NetGO. Values indicate the mean of the metric over genes or terms, and error bars indicate a confidence interval of 95%, estimated using 10,000 bootstrap iterations on the gene set or term set, respectively.

Figure 4 AUC-ROC for every organism per-gene (left) and per-term (right), higher values are better. Comparison of HMMER, InterPro, HMMER + InterPro, S2F, Argot 2.5, DeepGOPlus, GOLabeler, and NetGO. Values indicate the mean of the metric over genes or terms, and error bars indicate a confidence interval of 95%, estimated using 10,000 bootstrap iterations on the gene set or term set, respectively.

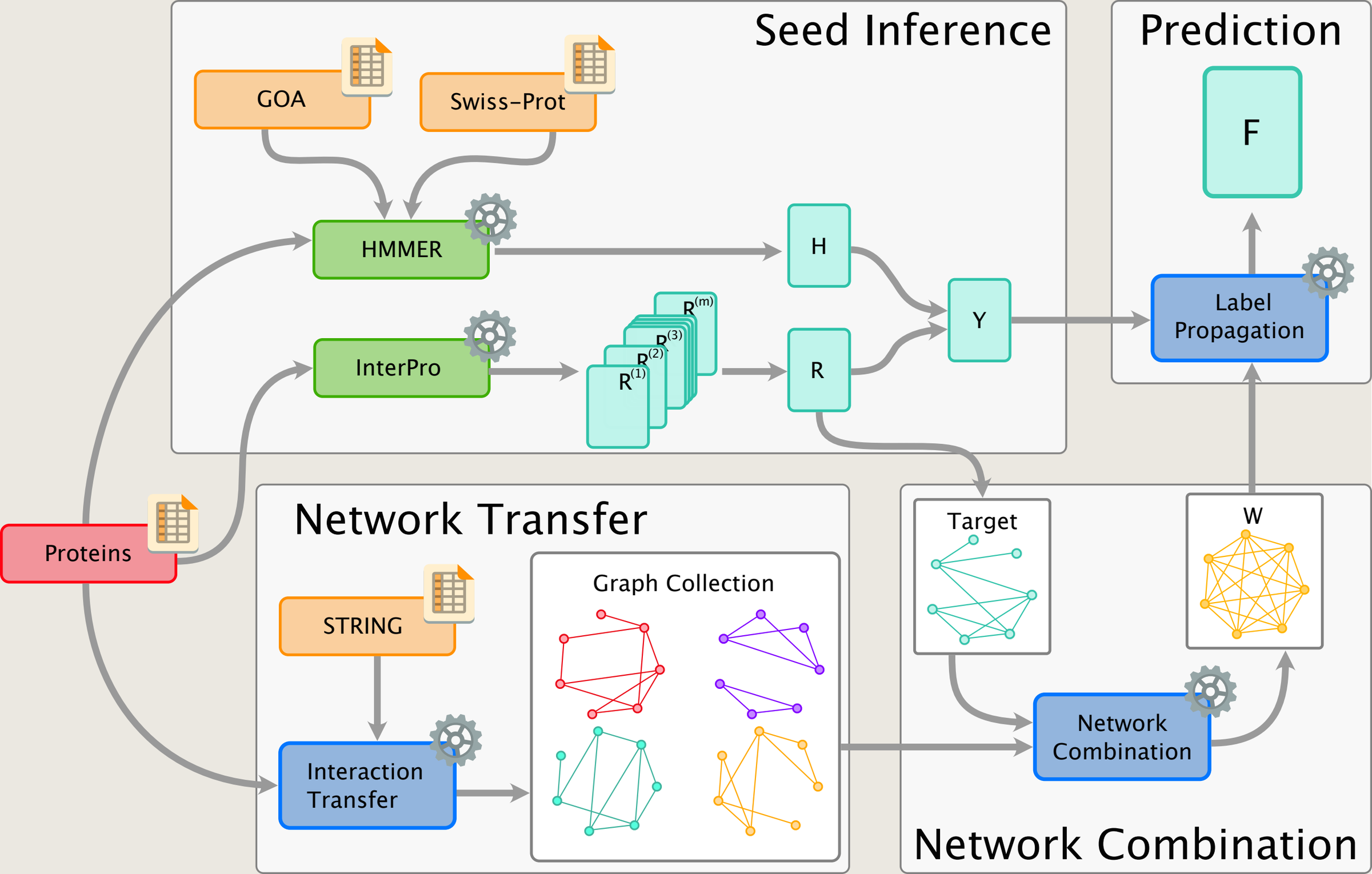
Figure 5 AUC-PR for every organism per-gene (left) and per-term (right), higher values are better. Comparison of HMMER, InterPro, HMMER + InterPro, S2F, Argot 2.5, DeepGOPlus, GOLabeler, and NetGO. Values indicate the mean of the metric over genes or terms, and error bars indicate a confidence interval of 95%, estimated using 10,000 bootstrap iterations on the gene set or term set, respectively.

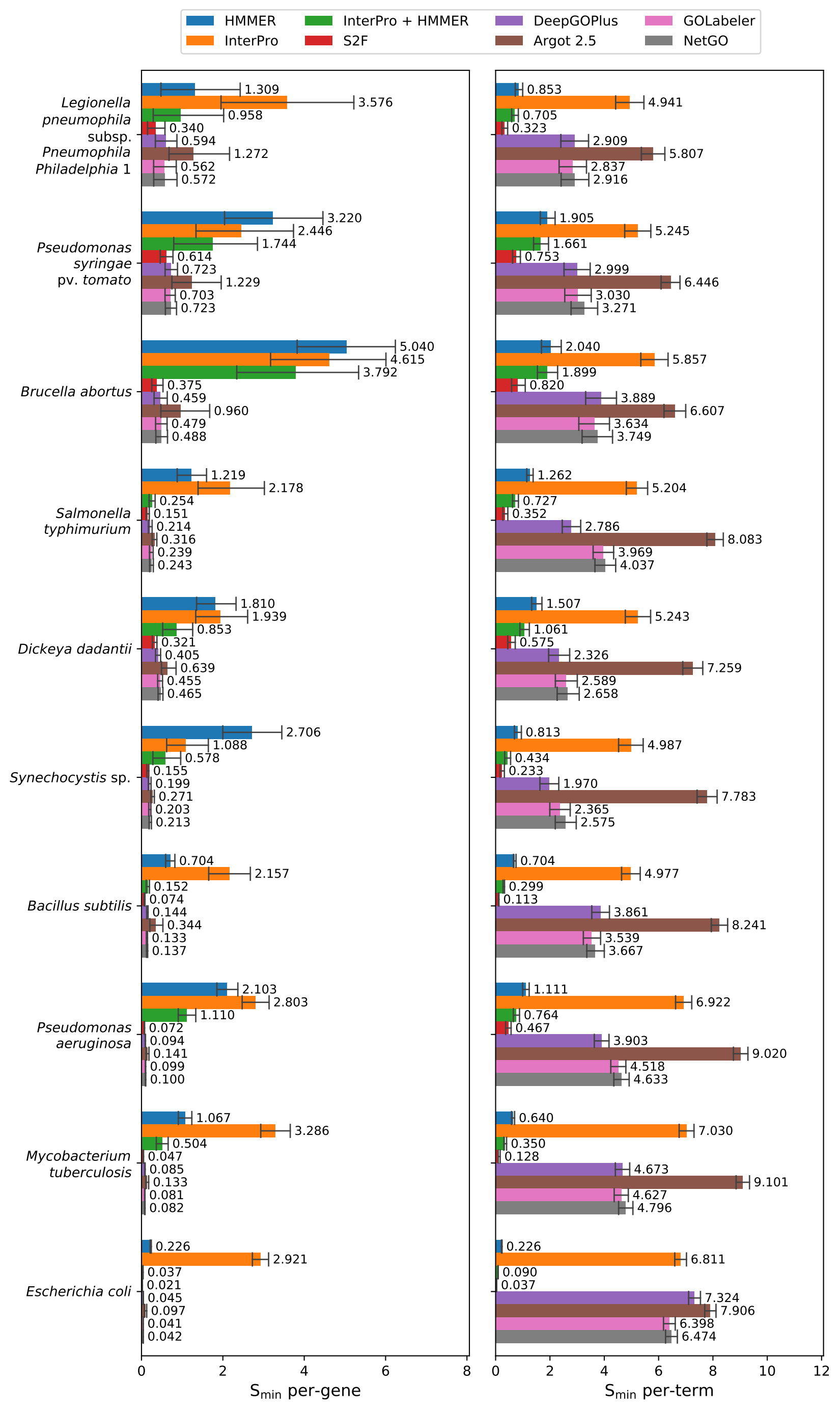
References

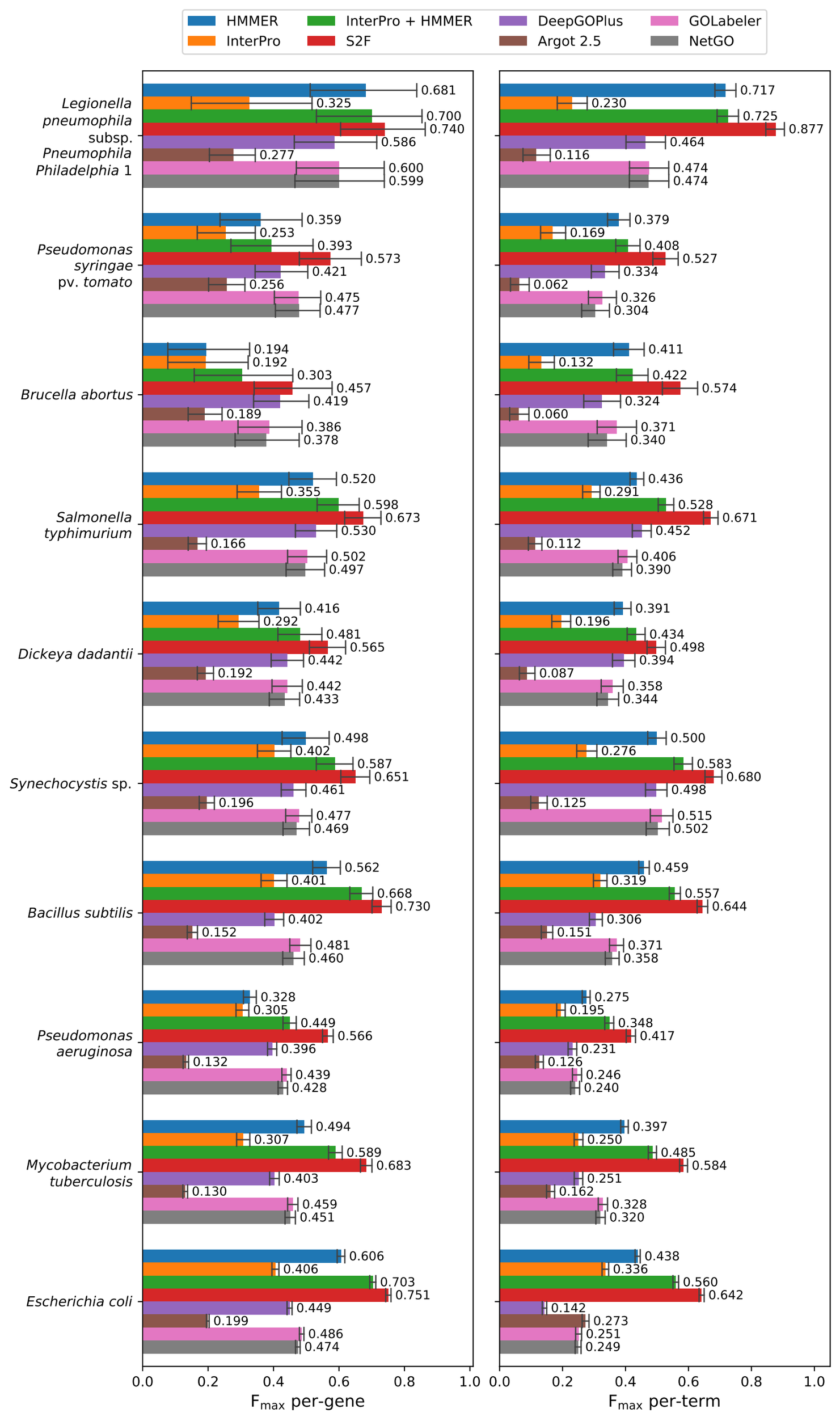
1. Cruz, L. M., Trefflich, S., Weiss, V. A. & Castro, M. A. A. Protein Function Prediction. *Methods Mol. Biol. Clifton NJ* **1654**, 55–75 (2017).
2. Shehu, A., Barbará, D. & Molloy, K. A Survey of Computational Methods for Protein Function Prediction. in *Big Data Analytics in Genomics* (ed. Wong, K.-C.) 225–298 (Springer International Publishing, 2016). doi:10.1007/978-3-319-41279-5_7.
3. Jiang, Y. *et al.* An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biol.* **17**, 184 (2016).
4. Ashburner, M. *et al.* Gene Ontology: tool for the unification of biology. *Nat. Genet.* **25**, 25–29 (2000).
5. Cowen, L., Ideker, T., Raphael, B. J. & Sharan, R. Network propagation: a universal amplifier of genetic associations. *Nat. Rev. Genet.* **18**, 551–562 (2017).
6. Zhou, N. *et al.* The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol.* **20**, 244 (2019).
7. Valentini, G. True Path Rule Hierarchical Ensembles for Genome-Wide Gene Function Prediction. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **8**, 832–847 (2011).
8. Friedberg, I. & Radivojac, P. Community-Wide Evaluation of Computational Function Prediction. in *The Gene Ontology Handbook* (eds. Dessimoz, C. & Škunca, N.) 133–146 (Springer New York, 2017). doi:10.1007/978-1-4939-3743-1_10.
9. Obozinski, G., Lanckriet, G., Grant, C., Jordan, M. I. & Noble, W. S. Consistent probabilistic outputs for protein function prediction. *Genome Biol.* **9**, S6 (2008).
10. Mitchell, A. L. *et al.* InterPro in 2019: improving coverage, classification and access to protein sequence annotations. *Nucleic Acids Res.* **47**, D351–D360 (2019).

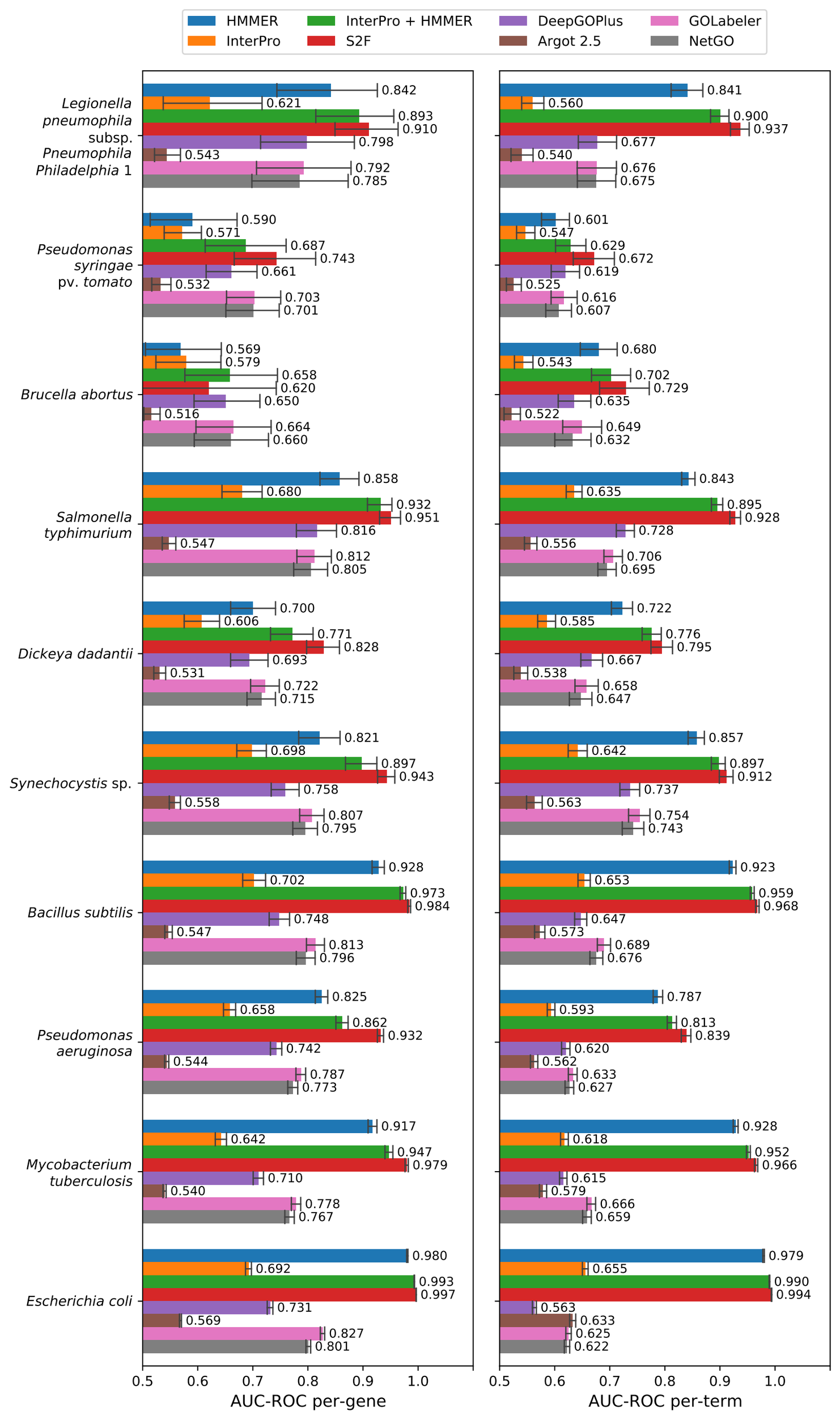
11. HMMER. http://hmmer.org/.
12. Walhout, A. J. *et al.* Protein interaction mapping in C. elegans using proteins involved in vulval development. *Science* **287**, 116–122 (2000).
13. Yu, H. *et al.* Annotation Transfer Between Genomes: Protein–Protein Interologs and Protein–DNA Regulogs. *Genome Res.* **14**, 1107–1118 (2004).
14. Ben-Hur, A. & Noble, W. S. Kernel methods for predicting protein–protein interactions. *Bioinformatics* **21**, i38–i46 (2005).
15. Sharan, R. *et al.* Conserved patterns of protein interaction in multiple species. *Proc. Natl. Acad. Sci.* **102**, 1974–1979 (2005).
16. Szklarczyk, D. *et al.* STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Res.* **47**, D607–D613 (2019).
17. Mostafavi, S., Ray, D., Warde-Farley, D., Grouios, C. & Morris, Q. GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome Biol.* **9**, S4 (2008).
18. Zhou, N. *et al.* The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *bioRxiv* 653105 (2019) doi:10.1101/653105.
19. Huntley, R. P. *et al.* The GOA database: gene Ontology annotation updates for 2015. *Nucleic Acids Res.* **43**, D1057-1063 (2015).
20. Lavezzo, E., Falda, M., Fontana, P., Bianco, L. & Toppo, S. Enhancing protein function prediction with taxonomic constraints – The Argot2.5 web server. *Methods* **93**, 15–23 (2016).
21. Kulmanov, M. & Hoehndorf, R. DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics* **36**, 422–429 (2020).

22. You, R. *et al.* GOLabeler: improving sequence-based large-scale protein function prediction by learning to rank. *Bioinformatics* **34**, 2465–2473 (2018).
23. You, R. *et al.* NetGO: improving large-scale protein function prediction with massive network information. *Nucleic Acids Res.* **47**, W379–W387 (2019).
24. Makrodimitris, S., van Ham, R. C. H. J. & Reinders, M. J. T. Automatic Gene Function Prediction in the 2020's. *Genes* **11**, 1264 (2020).
25. Cao, M. *et al.* Going the Distance for Protein Function Prediction: A New Distance Metric for Protein Interaction Networks. *PLOS ONE* **8**, e76339 (2013).
26. Zhou, D., Bousquet, O., Lal, T. N., Weston, J. & Schölkopf, B. Learning with Local and Global Consistency. in 8 (2004).
27. Torres, M., Haixuan Yang, Romero, A. E. & Paccanaro, A. Input Data for 'Protein Function Prediction for newly sequenced organisms'. (2021) doi:10.5281/ZENODO.5514323.
28. Torres, M., Haixuan Yang, Romero, A. E. & Paccanaro, A. *Source code for 'Protein Function Prediction for newly sequenced organisms'*. (Zenodo, 2021). doi:10.5281/ZENODO.5513071.
29. UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Res.* **47**, D506–D515 (2019).


Seed Inference
GOA
Swiss-Prot
HMMER
InterPro
H
R
Y
Prediction
F
Label Propagation
Proteins
Network Transfer
STRING
Interaction Transfer
Graph Collection
Target
W
Network Combination
Network Combination


HMMER
InterPro
InterPro + HMMER
S2F
DeepGOPlus
Argot 2.5
GOLabeler
NetGO
Legionella pneumophila subsp. Pneumophila Philadelphia 1
1.309
3.576
0.958
0.340
0.594
1.272
0.562
0.572
0.853
4.941
0.705
0.323
2.909
5.807
2.837
2.916
Pseudomonas syringae pv. tomato
3.220
2.446
1.744
0.614
0.723
1.229
0.703
0.723
1.905
5.245
1.661
0.753
2.999
6.446
3.030
3.271
Brucella abortus
5.040
4.615
3.792
0.375
0.459
0.960
0.479
0.488
2.040
5.857
1.899
0.820
3.889
6.607
3.634
3.749
Salmonella typhimurium
1.219
2.178
0.254
0.151
0.214
0.316
0.239
0.243
1.262
5.204
0.727
0.352
2.786
8.083
3.969
4.037
Dickeya dadantii
1.810
1.939
0.853
0.321
0.405
0.639
0.455
0.465
1.507
5.243
1.061
0.575
2.326
7.259
2.589
2.658
Synechocystis sp.
2.706
1.088
0.578
0.155
0.199
0.271
0.203
0.213
0.813
4.987
0.434
0.233
1.970
7.783
2.365
2.575
Bacillus subtilis
0.704
2.157
0.152
0.074
0.144
0.344
0.133
0.137
0.704
4.977
0.299
0.113
3.861
8.241
3.539
3.667
Pseudomonas aeruginosa
2.103
2.803
1.110
0.072
0.094
0.141
0.099
0.100
1.111
6.922
0.764
0.467
3.903
9.020
4.518
4.633
Mycobacterium tuberculosis
1.067
3.286
0.504
0.047
0.085
0.133
0.081
0.082
0.640
7.030
0.350
0.128
4.673
9.101
4.627
4.796
Escherichia coli
0.226
2.921
0.037
0.021
0.045
0.097
0.041
0.042
0.226
6.811
0.090
0.037
7.324
7.906
6.398
6.474
0
2
4
6
8
S$_{min}$ per-gene
0
2
4
6
8
10
12
S$_{min}$ per-term


HMMER
InterPro
InterPro + HMMER
S2F
DeepGOPlus
Argot 2.5
GOLabeler
NetGO
Legionella pneumophila subsp. Pneumophila Philadelphia 1
0.681
0.325
0.700
0.740
0.586
0.277
0.600
0.599
0.717
0.230
0.725
0.877
0.464
0.116
0.474
0.474
Pseudomonas syringae pv. tomato
0.359
0.253
0.393
0.573
0.421
0.256
0.475
0.477
0.379
0.169
0.408
0.527
0.334
0.062
0.326
0.304
Brucella abortus
0.194
0.192
0.303
0.457
0.419
0.189
0.386
0.378
0.411
0.132
0.422
0.574
0.324
0.060
0.371
0.340
Salmonella typhimurium
0.520
0.355
0.598
0.673
0.530
0.166
0.502
0.497
0.436
0.291
0.528
0.671
0.452
0.112
0.406
0.390
Dickeya dadantii
0.416
0.292
0.481
0.565
0.442
0.192
0.442
0.433
0.391
0.196
0.434
0.498
0.394
0.087
0.358
0.344
Synechocystis sp.
0.498
0.402
0.587
0.651
0.461
0.196
0.477
0.469
0.500
0.276
0.583
0.680
0.498
0.125
0.515
0.502
Bacillus subtilis
0.562
0.401
0.668
0.730
0.402
0.152
0.481
0.460
0.459
0.319
0.557
0.644
0.306
0.151
0.371
0.358
Pseudomonas aeruginosa
0.328
0.305
0.449
0.566
0.396
0.132
0.439
0.428
0.275
0.195
0.348
0.417
0.231
0.126
0.246
0.240
Mycobacterium tuberculosis
0.494
0.307
0.589
0.683
0.403
0.130
0.459
0.451
0.397
0.250
0.485
0.584
0.251
0.162
0.328
0.320
Escherichia coli
0.606
0.406
0.703
0.751
0.449
0.199
0.486
0.474
0.438
0.336
0.560
0.642
0.142
0.273
0.251
0.249
0.0
0.2
0.4
0.6
0.8
1.0
F$_{max}$ per-gene
F$_{max}$ per-term


HMMER
InterPro
InterPro + HMMER
S2F
DeepGOPlus
Argot 2.5
GOLabeler
NetGO
Legionella pneumophila subsp. Pneumophila Philadelphia 1
0.842
0.621
0.893
0.910
0.798
0.543
0.792
0.785
0.841
0.560
0.900
0.937
0.677
0.540
0.676
0.675
Pseudomonas syringae pv. tomato
0.590
0.571
0.687
0.743
0.661
0.532
0.703
0.701
0.601
0.547
0.629
0.672
0.619
0.525
0.616
0.607
Brucella abortus
0.569
0.579
0.658
0.620
0.650
0.516
0.664
0.660
0.680
0.543
0.702
0.729
0.635
0.522
0.649
0.632
Salmonella typhimurium
0.858
0.680
0.932
0.951
0.816
0.547
0.812
0.805
0.843
0.635
0.895
0.928
0.728
0.556
0.706
0.695
Dickeya dadantii
0.700
0.606
0.771
0.828
0.693
0.531
0.722
0.715
0.722
0.585
0.776
0.795
0.667
0.538
0.658
0.647
Synechocystis sp.
0.821
0.698
0.897
0.943
0.758
0.558
0.807
0.795
0.857
0.642
0.897
0.912
0.737
0.563
0.754
0.743
Bacillus subtilis
0.928
0.702
0.973
0.984
0.748
0.547
0.813
0.796
0.923
0.653
0.959
0.968
0.647
0.573
0.689
0.676
Pseudomonas aeruginosa
0.825
0.658
0.862
0.932
0.742
0.544
0.787
0.773
0.787
0.593
0.813
0.839
0.620
0.562
0.633
0.627
Mycobacterium tuberculosis
0.917
0.642
0.947
0.979
0.710
0.540
0.778
0.767
0.928
0.618
0.952
0.966
0.615
0.579
0.666
0.659
Escherichia coli
0.980
0.692
0.993
0.997
0.731
0.569
0.827
0.801
0.979
0.655
0.990
0.994
0.563
0.633
0.625
0.622
0.5
0.6
0.7
0.8
0.9
1.0
AUC-ROC per-gene
AUC-ROC per-term

HMMER
InterPro
InterPro + HMMER
S2F
DeepGOPlus
Argot 2.5
GOLabeler
NetGO

Legionella pneumophila subsp. *Pneumophila Philadelphia* 1
0.003 0.156 0.138 0.623 0.526 0.362 0.353 0.337
0.009 0.155 0.166 0.212 0.166 0.142 0.131 0.130

Pseudomonas syringae pv. *tomato*
0.083 0.184 0.224 0.408 0.307 0.322 0.284 0.282
0.076 0.161 0.142 0.212 0.222 0.148 0.201 0.197

Brucella abortus
0.051 0.142 0.118 0.315 0.286 0.154 0.185 0.180
0.036 0.132 0.111 0.166 0.150 0.108 0.123 0.119

Salmonella typhimurium
0.004 0.224 0.293 0.528 0.410 0.231 0.183 0.177
0.003 0.124 0.145 0.204 0.159 0.067 0.120 0.111

Dickeya dadantii
0.041 0.184 0.234 0.437 0.338 0.224 0.200 0.195
0.054 0.155 0.173 0.252 0.228 0.104 0.150 0.145

Synechocystis sp.
0.006 0.186 0.255 0.505 0.305 0.220 0.161 0.157
0.014 0.113 0.114 0.206 0.135 0.072 0.118 0.095

Bacillus subtilis
0.000 0.191 0.264 0.596 0.280 0.215 0.194 0.181
0.000 0.134 0.163 0.222 0.109 0.063 0.104 0.100

Pseudomonas aeruginosa
0.014 0.158 0.202 0.421 0.265 0.167 0.170 0.164
0.017 0.108 0.118 0.155 0.104 0.056 0.084 0.082

Mycobacterium tuberculosis
0.002 0.198 0.270 0.580 0.314 0.211 0.200 0.198
0.003 0.124 0.146 0.215 0.101 0.062 0.104 0.103

Escherichia coli
0.000 0.248 0.347 0.621 0.374 0.272 0.244 0.245
0.000 0.155 0.192 0.241 0.072 0.096 0.091 0.094

0.0 0.2 0.4 0.6 0.8 1.0
AUC-PR per-gene
0.0 0.2 0.4 0.6 0.8 1.0
AUC-PR per-term