Benchmarking gene ontology function predictions using negative annotations

Alex Warwick Vesztrocy1,2,3,* and Christophe Dessimoz1,2,3,4,5,*

1Department of Genetics, Evolution and Environment, University College London, London, WC1E 6BT, UK, 2SIB Swiss Institute of Bioinformatics, 1015 Lausanne, Switzerland, 3Department of Computational Biology, University of Lausanne, 1015 Lausanne, Switzerland, 4Department of Computer Science, University College London, London, WC1E 6BT, UK and 5Centre for Integrative Genomics, University of Lausanne, 1015 Lausanne, Switzerland

*To whom correspondence should be addressed.

Abstract

Motivation: With the ever-increasing number and diversity of sequenced species, the challenge to characterize genes with functional information is even more important. In most species, this characterization almost entirely relies on automated electronic methods. As such, it is critical to benchmark the various methods. The Critical Assessment of protein Function Annotation algorithms (CAFA) series of community experiments provide the most comprehensive benchmark, with a time-delayed analysis leveraging newly curated experimentally supported annotations. However, the definition of a false positive in CAFA has not fully accounted for the open world assumption (OWA), leading to a systematic underestimation of precision. The main reason for this limitation is the relative paucity of negative experimental annotations.

Results: This article introduces a new, OWA-compliant, benchmark based on a balanced test set of positive and negative annotations. The negative annotations are derived from expert-curated annotations of protein families on phylogenetic trees. This approach results in a large increase in the average information content of negative annotations. The benchmark has been tested using the naıve and BLAST baseline methods, as well as two orthology-based methods. This new benchmark could complement existing ones in future CAFA experiments.

Contact: alex.warwickvesztrocy@unil.ch or christophe.dessimoz@unil.ch

Availability and Implementation: All data, as well as code used for analysis, is available from https://lab.dessimoz.org/20_not.

Supplementary information: Supplementary data are available at Bioinformatics online.

1 Introduction

According to the GOLD database, hundreds of thousands of genomes have already been sequenced, including close to ten thousand eukaryotes (Mukherjee et al., 2019). Within one decade, the Earth BioGenome consortium aims to sequence 1.5 million eukaryotic sequences (Lewin et al., 2018). At a molecular level, however, nearly all biological knowledge is concentrated in human and a handful of model species. Strikingly, in UniProt-GOA (Huntley et al., 2015), over 80% of all Gene Ontology annotations supported by direct experimental evidence are concentrated in just seven species. Thus, for the overwhelming majority of species, functional characterization is almost entirely reliant on automated computational methods (Cozetto and Jones, 2017).

As such, it is critical to benchmark the various computational methods. The Critical Assessment of protein Function Annotation algorithms (CAFA) series of community experiments have provided the most comprehensive benchmark, with a time-delayed analysis leveraging new experimentally supported annotations (Jiang et al., 2016; Radivojac et al., 2013; Zhou et al., 2019).

One major complication in assessing protein function predictions is that proteins typically possess multiple ‘functions’ [sensu Gene Ontology (GO) (Thomas, 2017)], and knowledge of these functions, even for well-known genes in model species, is typically incomplete (Huttenhower et al., 2009). This incomplete state of knowledge is referred to as the open world assumption (OWA) (Skaqna et al., 2017; Thomas et al., 2012). This has previously been shown to affect the performance measures of conventional benchmarks (Huttenhower et al., 2009). Whilst CAFA benchmarks have been shown to be relatively stable in the short term (Jiang et al., 2014), they do not fully account for the OWA. This leads to a systematic underestimation of precision (Dessimoz et al., 2013). For example, consider the human gene Serotonin N-acetyltransferase (SNAT_HUMAN) which controls the night/day rhythm of melatonin production in the pineal gland. When this protein had no GO annotations, a method may have predicted ‘circadian rhythm’ (GO:0007623), ‘rhythmic process’ (GO:0048511) and ‘indolalkylamine biosynthetic process’ (GO:0046219). Then, when ‘circadian rhythm’ and ‘rhythmic process’ were experimentally associated with
this gene, they would both be considered true positives and ‘indolylalkylamine biosynthetic process’ as a false positive. Several years later, however, this term was also associated with this protein—contradicting the assertion that it was a false positive and demonstrating the problem with assuming a ‘closed world’ of complete knowledge.

To be compliant with the OWA during benchmarking, explicit negative annotations are desirable—those that state a particular gene does not have a particular function—thus making it possible to classify computational predictions of the contrary as a false positive (Dessimoz et al., 2013). Yet currently, in UniProt-GOA, less than 2.5% of all experimentally annotated proteins have a Gene Ontology annotation which is negatively qualified, indicated by the use of the ‘NOT’ tag in the qualifier field of a GAF file (Gaudet et al., 2017).

Furthermore, reasoning on ontologies when using negative annotations requires different treatment than with positive annotations. Thus, the information content (IC) associated with negative annotations needs to be computed differently. As is elaborated below, this has not been accounted for in benchmarks to date.

Previous work to identify negative annotations tends to focus on their use as negative examples in machine learning methods. For example, NoGO (Youngs et al., 2014) generated a database of negative annotations based on annotated and unannotated examples using methods for relevance feedback from the field of information retrieval—the Rocchio, 1-DNF and AGPS algorithms. These methods can suffer from predicting overly specific terms. This has the same issues as only having positive annotations to very general terms, in that overly specific negative annotations carry little information. NegGOA (Fu et al., 2016) aimed to overcome this using the ontology structure, random walks and co-occurrence of terms to model the potentiality of missing annotations.

This article introduces an approach to derive a large number of negatively qualified annotations from expertly curated gene phylogenies. Using these, a framework for OWA-compliant benchmarking was developed, based on a balanced test set of positive and negative annotations. This benchmark has been tested on the naïve and BLAST baseline methods, GOtcha and an orthology-based method. This new benchmarking framework could complement existing ones in future CAFA experiments.

2 Materials and methods

This section begins by highlighting the differences in benchmarking GO annotations with explicit negative annotations, over the current practice. This requires a large number of negative annotations—a method is then presented to derive many negative annotations based on expertly curated gene phylogenies. These can then be used in an OWA-compliant benchmarking framework, illustrated with a method comparison.

2.1 Benchmarking gene ontology annotation with explicit negative annotations

A large amount of explicit negative annotations would help to address the OWA in benchmarking. Further, benchmarking using these negative annotations requires different handling. It is customary to assess automated function predictors in a protein-centric sense. This means computing some measure of quality—for example precision-recall—for each protein, with an average taken over the proteins tested. A set of true annotations is required, that are not available to the predictor, to properly assess the method. It is currently commonplace to identify the false-positive GO terms as those that have been predicted, but not in the set of true annotations (Table 1a). When there are sufficient negative annotations in the true annotation set for a given protein, the false positives can then be identified as overlapping with these (Table 1b).

Furthermore, because different terms vary in their IC—for example a positive association with a term such as ‘root hair elongation’ (GO:0048767) is more informative than the more general term ‘growth’ (GO:0009753)—it is common to compute weighted precision-recall curves, to correct for the bias towards general terms. For instance, Clark and Radivojac (2013) proposed to weight by the information accretion—the increase in information that a particular term gives, relative to all parent terms. This approach was subsequently implemented in CAFA 2 (Jiang et al., 2016). To compute the IC of GO terms, the probability is required. This can be estimated using the empirical annotation frequency of each term.

However, it is important to acknowledge that the IC of a single term is not the same if it is negatively or positively qualified. For example it is easier to show that a gene should be annotated with the general metabolic process term (GO:0008152) than a particular metabolic process, for instance lactose biosynthetic process (GO:0009899). In contrast, it is exceptionally challenging to show that a gene is not associated with any metabolic process, in comparison to showing that it is not involved in a very specific one. Thus, more general terms in the GO have a lower IC than more specific ones when a positive association is made. However, the inverse is true for negative associations—general terms have a greater IC than those that are more specific.

Hence, it is necessary to estimate the IC of negative annotations separately—ensuring to propagate term counts to children instead of the parents, unlike for positive annotations (Gaudet and Dessimoz, 2017).

2.1.1 Information content computation

IC can be estimated by computing the frequency of a particular GO term in a given database of annotations. The IC that an individual term holds is then computed as

\[ ic_t(t) = -\log_2(\frac{P[t]}{\frac{1}{C_1}TT}) \]

where \( t \) is a single GO term and \( P[t] \) is the empirical probability of observing said term. The logarithm is taken base 2 by convention, with the units of information as Shannons or bits (Shannon, 1948). Then, the IC of a set of terms \( T \), can be computed as

\[ ic_T(TT) = -\log_2(\frac{1}{C_1}TT) \]

where \( P[TT] \) is the empirical joint probability, calculated directly from the annotation matrix \( P \), considering for co-occurrence of the annotations. Note, proteins were considered annotated if they had at least one annotation in at least one aspect of the GO, lower than the root term, listed in the UniProt-GOA database (Barrell et al., 2009).

| Table 1. Definitions of true positive (TP), false positive (FP) and false negative (FN) for a single GO term on a single protein used in (a) CWA benchmarks (current benchmarks) and (b) OWA benchmarks (in this article) for no-knowledge targets |
| --- |
| (a) CWA Benchmark | (b) OWA Benchmark |
| **Pred.** | **True** | **Pred.** | **True** |
| TP | FP | TP | FP |
| FN | TN | FN | TN |
Evidence to the contrary, the function of a gene is maintained through the extant genes. Ancestral annotations are then propagated down the phylogeny to (‘NOT’-qualified) annotations are recorded in ancestral states. These lies (Muruganujan et al. 2015, 2016) using the Phylogenetic Annotation and Inference Tool (PAINT) (Gaudet et al., 2017, 2018). Irrespective of the reason, an expert curator has deemed that there is currently a lack of evidence to annotate the root node with this term. As such, it can be argued that an automated predictor should incur a penalty for predicting such terms.

By scanning the PAINT annotations for such instances, it is possible to derive many pairs, \((p, t)\), where \(p\) is a protein which is member of a family where an ancestral node, not in its direct lineage, has been annotated to a GO term \(t\). That is, \(p\) is not covered by a PAINT annotation (positive or negative) for a GO term \(t\), but other members of its family are. Negative annotations are only derived for terms \(t\) for which \(i_{\text{c}}(t) \geq 5\). This aims to reduce the number of incorrect derivations from cautious curation (as elaborated upon in the discussion below).

2.3 Balanced benchmarking
In general, approaches to benchmarking GO annotations acknowledge that some aspects of function are easier to predict than others. Thus, they typically consider the IC of each annotation. Furthermore, since the IC for the same term varies whether it is associated positively or negatively with a given target (see above), this difference should also be considered. One such way to account for differences in IC amongst annotations is by weighting predictions by their IC. However, this only works up to a point: if there are no, or very few, annotations with high IC, the results will have a very large variance and thus not be particularly informative. To avoid this, it is possible to design a benchmark to test GO terms for which there are informative positive and negative examples. Henceforth, this design shall be referred to as a ‘weighted and balanced’ benchmark.

To investigate the two approaches (weighted-only, as well as weighted and balanced), two test sets were generated that represent each case.

2.3.1 Weighted-only
For the weighted-only case, the test set includes one pair of proteins per family, for which it is possible to choose a protein with positive annotations \((p_+\) and one with negative annotations \((p_-)\). This resulted in 2,292 protein-pairs used for this benchmark. True-positive and false-negative terms are identified with the positive protein, \(p_+\), and false positives with \(p_-\).

Denote the sets of terms classified as true positive, false negative and false positive as \(TP, FN, FP\), respectively. In the OWA-compliant benchmarking framework, the weighted metric representing each of these is computed by calculating the IC of the terms in each set. For true-positive and false-negative terms, that is \(TP_w = i_{\text{c}}(TP)\) and \(FN_w = i_{\text{c}}(FN)\). For false positives, this is instead calculated as \(FP_w = i_{\text{c}}(FP)\).

As the protein pairs in the test set are chosen without stipulation on the depth or amount of information that each gene has per aspect or overall. Weighting is then required to correct the bias due to the differences in IC—both within and between the positive and negative annotation sets. To balance within, the IC of the terms inside each set. For true-positive and false-negative terms, that is \(TP_w = i_{\text{c}}(TP)\) and \(FN_w = i_{\text{c}}(FN)\). For false positives, this is instead calculated as \(FP_w = i_{\text{c}}(FP)\).
balance between the positive and negative sets a normalized measure is computed for each of the gene sets (e.g. normalized true positive), normalizing by the total IC of the positive or negative example genes. That is, the weighted-normalized measures for computing precision and recall are

\[
\begin{align*}
\hat{TP}_w^t &= \frac{\sum_{\rho \in \mathcal{P}} \text{ic}_z(TP^t_\rho)}{\sum_{\rho \in \mathcal{P}} \text{ic}_z(A^+_\rho)}, \\
\hat{FN}_w^t &= \frac{\sum_{\rho \in \mathcal{P}} \text{ic}_z(FN^t_\rho)}{\sum_{\rho \in \mathcal{P}} \text{ic}_z(A^-_\rho)}, \\
\text{and } \hat{FP}_w^t &= \frac{\sum_{\rho \in \mathcal{P}} \text{ic}_z(FP^t_\rho)}{\sum_{\rho \in \mathcal{P}} \text{ic}_z(A^-_\rho)},
\end{align*}
\]

where \(TP^t_\rho, FN^t_\rho, FP^t_\rho\) are the sets of true-positive and false-negative GO terms for \(\rho\), \(p\), the false positive for \(p\), both with confidence cut-off \(t\). \(A^+_\rho\) is the truth set of positive annotations for \(p\), and \(A^-_\rho\) is the truth set of negative annotations for \(p\).

### 2.3.2 Weighted and balanced

For the ‘weighted and balanced’ case, proteins were chosen for every GO term that have positive and negative examples within a protein family, resulting in 12,613 protein pairs (with associated GO term). In this case, it is still necessary to weight, to account for variation of information among GO terms for positive or negative annotations (Fig. 2 and Section 2.1.2).

These are computed over all families \((f \in \mathcal{F})\) for all terms \(t\) for which there are positive and negative examples in the family, \(p^{(f)}\) and \(p^{(f)}\), or more formally the terms for each family are defined as

\[
T_f = \{ t : \exists p^{(f)} \text{ s.t. } t \in A^+_p \land \exists p^{(f)} \text{ s.t. } t \in A^-_p \},
\]

where, as previously, \(A^+_p\) is the truth set of positive annotations for \(p\), and \(A^-_p\) is the truth set of negative annotations for \(p\).

The weighted and normalized measures for true positive, false negative and false positive are then

\[
\begin{align*}
\hat{TP}_w^f &= \frac{\sum_{t \in T_f} \sum_{\rho \in \mathcal{P}} 1_{TP}(t,f) \cdot \text{ic}_z(t)}{\sum_{t \in T_f} \sum_{\rho \in \mathcal{P}} \text{ic}_z(t)}, \\
\hat{FN}_w^f &= \frac{\sum_{t \in T_f} \sum_{\rho \in \mathcal{P}} 1_{FN}(t,f) \cdot \text{ic}_z(t)}{\sum_{t \in T_f} \sum_{\rho \in \mathcal{P}} \text{ic}_z(t)}, \\
\text{and } \hat{FP}_w^f &= \frac{\sum_{t \in T_f} \sum_{\rho \in \mathcal{P}} 1_{FP}(t,f) \cdot \text{ic}_z(t)}{\sum_{t \in T_f} \sum_{\rho \in \mathcal{P}} \text{ic}_z(t)},
\end{align*}
\]

where

\[
1_{TP}(t,f) = \begin{cases} 1 & \text{if } t \in TP^t_\rho \\ 0 & \text{if } t \notin TP^t_\rho \end{cases},
\]

and similarly,

\[
1_{FN}(t,f) = \begin{cases} 1 & \text{if } t \in FN^t_\rho \\ 0 & \text{if } t \notin FN^t_\rho \end{cases}
\]

### 2.3.3 Comparison benchmark

The benchmark set of proteins \(\mathcal{P}\) was chosen subject to routines described above (Sections 2.3.1 and 2.3.2). All existing knowledge was removed from the annotation data provided to the methods. Each predictor outputs in the form \((p, t, x)\), where \(p \in \mathcal{P}\) is a protein identifier, \(t\) a GO term and \(x \in [0,1]\) the method’s confidence in its prediction. Precision–recall curves were computed for both benchmarks, by varying the confidence cut-off \((x \geq \tau, \tau \in [0,1])\) that each method reports in its predictions in 100 equal steps of \((0.01)\), as in CAFA.

For comparison to benchmarks under the CWA, the positive example genes from the weighted-only benchmark were used to identify true positives and weighting by information accretion. The CWA benchmark presented then corresponds to the weighted precision–recall benchmark in CAFA (Jiang et al., 2016; Radivojac et al., 2013; Zhou et al., 2019).

Predictors for which it was possible to provide custom training data were used: the two baseline methods included in CAFA (naive and BLAST), GOtcha (Martin et al., 2004) and HOGPROP (DessimozLab in the third CAFA).

#### 2.3.3.1 Naïve predictor

The naive predictor assigns the same \((t, x)\) for all \(p \in \mathcal{P}\). The confidence score is the frequency of the term in the database (that is, the proportion of annotations with this term). This is computed using only experimentally verified annotations on proteins in UniProtKB/Swiss-Prot (The UniProt Consortium, 2017, 2018).

#### 2.3.3.2 BLAST predictor

For each term, the confidence is defined as the maximum percentage identity [identified using BLAST+ (Camacho et al., 2009)] to a sequence that has been annotated with this term. Again, only experimentally verified annotations to proteins in UniProtKB/Swiss-Prot (The UniProt Consortium, 2017, 2018) were used.
2.3.3.3 GOTcha. GOTcha (Martin et al., 2004) is a more sophisticated predictor, making use of not only sequence homology but also the structure of the GO whilst combining BLAST hits. Consider a target protein \( p \), GO term \( t \) and a set of sequences associated with said term \( S_t \). Then, first an \( r \)-score is computed as \( r_{r} = -\sum_{i=0}^{\infty} \log(e(p, s)) \) where \( e(p, s) \) represents the E-value of the alignment between the target sequence \( p \) and sequence \( s \). \( r \)-scores are then calculated by dividing the \( r \)-scores by the score for the root term of the relevant aspect—that is \( r_t = r_{r}/r_{\text{root}} \). GOTcha was included in the assessment of Clark and Radijojac (2013) as an example of a good predictor, performing better than the baseline methods.

Note, as will become relevant in the results, due to the combination of BLAST scores the confidence assigned by GOTcha (the \( r \)-score) tends to only predict general terms. As all annotated hits will be associated with at least one aspect’s root term, for most terms \( t \) \( r_{\text{root}} \gg r_t \) and so \( r_t \rightarrow 0 \) for all but the most frequent terms. As the lowest cut-off \( \tau \) is 0.01, any predictions with a score less than this will not be considered.

2.3.3.4 HOGPROP. This was submitted to the third CAFA as DessimozLab and uses the hierarchical orthologous groups (HOGs) from the OMA project (Altenhoff et al., 2018), with the same algorithm previously applied to predicting potential causal genes in QTL experiments (Warwick Vesztrony et al., 2018). Two variants are included in this article—HOGPROP1 uses experimentally derived annotations, as well as a sub-set of the electronic annotations deemed to be ‘trusted’ [see (Warwick Vesztrony et al., 2018) for details]; HOGPROP2 uses all annotations, except for electronic ones which are filtered to only include the ‘trusted’ ones.

A subset of GO annotations (including some electronic annotations [based on Škunca et al. (2012)]) are given a score dependent on their evidence code. These terms, with scores, are then associated with the leaves of the hierarchical structure (genes), before being pushed up and pulled down the hierarchy. The score decays across each edge (fixed rate of 20%), with a penalty when propagating over paralogous relationships of a double decay. Scores are combined at each node (using summation) during the up-propagation, whilst the maximum score is taken during down-propagation.

3 Results

This section first gives an outline of the additional negative annotations, derived from expertly curated gene phylogenies. After which, to illustrate the differences in a balanced OWA-compliant benchmark, the results of the method comparison are given.

3.1 Derived negative annotations

A large number of negative annotations were required to proceed with the balanced benchmarking. One such source is described in Section 2.2. Here, PANTHER families were scanned for instances of proteins where an ancestral node, not in its direct lineage, had been annotated to a particular GO term (as in Fig. 1c). That is many pairs can be derived from the PAINT annotations \( (p, t) \), where \( p \) is a protein which is member of a family where an ancestral node, not in its direct lineage, has been annotated to a GO term \( t \).

Negative annotations were curated using the ancestral annotations from PAINT on PANTHER 13.1 families, provided in personal correspondence on August 21, 2018. At this time, 5,664 PANTHER families contained annotations, for which it was possible to derive at least one extra negative annotation on 2,894. In order not to make too general negative assertions, only GO terms for which the ‘positive’ IC was greater than 5 bits were used.

The number of such pairs is shown in Figure 3 for each aspect of the Gene Ontology—biological process (BP), cellular component (CC) and Molecular Function (MF). In the database, only 11,635 proteins were covered by a negative annotation in UniProt-GOA—consisting of 4,911 with BP annotations, 4,619 with CC and 5,068 with MF. After including the derived negative annotations, this increased to 330,635—98,848 with BP, 268,831 with CC and 192,307 with MF. This is more than the number of proteins with at least one positive (non-IEA) annotation (323,438) as well as more than those with only at least one CC positive (non-IEA) annotation (266,638). Further, the IC of the derived negative annotations is similar to those already in UniProt-GOA (Supplementary Fig. S1).
set consists of pairs of proteins, a positive and negative example, for each gene family containing both types. This tests a predictor’s ability to discriminate between homologous proteins.

With a balanced test set, the na"ıve predictor performs much worse than in conventional CWA tests. This is because very general predictions, which are very easy to prove but nearly impossible to disprove, are by design not considered here. In other words, when na"ıve is evaluated on testable predictions, it makes many mistakes, which is reflected in the OWA benchmark. The recall too is markedly lower, which is to be expected with a method inherently limited to predicting the most frequent terms only.

Likewise, results obtained for the BLAST predictor are more reasonable than in conventional CWA benchmarks: precision is very high where recall is low, but degrades steeply when recall increases. This makes sense, as the confidence score is based on the percentage sequence identity, high-precision-low-recall results are obtained when sequence identity is close to 100%, and where one would expect functions to be highly conserved. Increasing recall requires more permissive thresholds, which also results in more false positives.

One last finding of note is that GOtcha, a method which combines BLAST results, performs particularly well under the CWA benchmark. For instance, on the MF aspect, GOtcha achieves an $F_{\text{max}}$ of 0.65 compared to the next best method of 0.58 (HOGPROP2). However, in the weighted and balanced OWA benchmark, it performs worse than BLAST ($F_{\text{max}}$ of 0.52 versus 0.55 in MF). This large discrepancy appears to be due to two main factors. First, the internal scoring scheme of GOtcha strongly favours general terms (see Section 2.3.3). As seen with the na"ıve predictor, predictions of general GO terms tend to be rewarded in conventional benchmarks [corroborated by Clark and Radivojac (2013) and Jiang et al. (2016)]. However, being practically impossible to disprove, they are by design not considered in the balanced benchmark. Second, given a target protein to be annotated, although GOtcha uses the $E$-values of BLAST matches to the target to assess the relative plausibility of the GO annotations associated with each match, it then normalizes the scores obtained for each target by the maximum score of that target. As a result, predictions for a target for which the best functionally annotated BLAST match is, say, 100% identical could receive the same confidence as a prediction for a target for which the best is only 40% identical. Indeed, by removing this normalization, a substantial improvement for GOtcha was observed in the weighted and balanced OWA benchmark (Supplementary Fig. S4).

4 Discussion and conclusion

Current benchmarks make an assumption that proteins are fully annotated, by identifying false positives as all the predicted terms...
that are not confirmed by experimentally backed annotations. This assumes that the proteins used for benchmarking are exhaustively annotated (‘Closed World Assumption’, CWA). By contrast, this work does not assume exhaustive annotations (OWA), and instead relies on explicit negative annotations to assess the accuracy of predicted annotations.

This work makes two main contributions. First, it provides a methodological framework to benchmark using negative annotations. Second, it provides a way to obtain substantially more negative annotations for benchmarking. The latter is needed because—even after applying the ‘true-path rule’ (that is, propagating annotations according to the GO hierarchy) (Valentini, 2009)—there are currently only few curated negative annotations in databases.

To overcome the relative paucity of negative annotations (Fig. 3), this study identified a substantial source of negative annotations derived from the expertly curated annotation of gene phylogenies in the PAINT project. After performing this procedure, when considering all genes which are members of PANTHER families that have been annotated in PAINT, there is roughly the same number of genes that have at least one positive annotation to that with at least one negative annotation.

Although PAINT has been used as the main source of negative annotations, the methodology is general. Other sources, such as from NoGO (Youngs et al., 2014) or NegGOA (Fu et al., 2016), could be used instead.

As the negative annotations used in this study are derived from expertly curated gene phylogenies, they are of higher quality than negative electronic assertions. However, they are less so than direct negative annotations performed manually by an expert. Phylogenetic reconstruction can be difficult, particularly around short internal branches and in the deeper parts of tree. Functional annotation of ancestral nodes requires careful judgement by the curator. The curator has to decide on the most appropriate level of specificity of the term used in ancestral annotations. If a curator, in an abundance of caution, assigns an overly general term to a subset of the gene family, the lack of this annotation will be indicative of the absence of that function. Note, however, that the assumption is made within the specific context of phylogenies which have been annotated and reviewed as a whole by expert curators. Furthermore, there is restraint in the procedure from deriving negative annotations of general terms (i.e. < 5, see Section 3.1), because curators occasionally use general terms to convey uncertainty in their annotations. While such behaviour is prudent in terms of the positive annotations, applying this derivation procedure would result in imprudent negative annotations.

Despite the plethora of methods developed and submitted to the CAFA challenge, only a few of them are available as standalone software. This makes it difficult to test them on newly developed benchmarks, such as the one introduced here. Note that web-based services, while convenient for end-users, are difficult to include in such a benchmark due to the lack of control over the input—it is important that the ontology definition and existing protein annotations are carefully controlled during training, to avoid circular evaluation.

Time-lapsed studies, such as CAFA, are by design less prone to circular evaluation. However, they require a steady supply of new annotations. For the derived negative annotations introduced here, time-lapsed studies would require steady supply of gene families newly annotated by PAINT or a similar curated approach. This may seem more constraining than merely annotating individual gene targets using the literature. However, family-wise annotation is also more consistent and scalable than the inconsistent process of annotating individual targets; their value in benchmarking based on negative examples is an additional incentive for this curation effort.

Fig. 5. Sub-family of PANTHER family PTHR10686—the root term is annotated to transmembrane transport, whilst particular sub-families have been annotated to folic acid transmembrane transporter activity and thiamine transmembrane transporter activity. This implies that for example proteins outside of that annotated with folic acid transmembrane transporter activity (green) should not be annotated with this term.
Table 2. Results for subset of tests performed on PANTHER family FTHR10686 in the weighted and balanced OWA benchmark, at the $F_{\text{max}}$ point

| GO term | Name | Aspect | Positive | Negative | Method predictions |
|---------|------|--------|----------|----------|--------------------|
| GO: 0015234 | Thiamine transmembrane transporter activity | MF | F6SXG7 (Sub-Fam. C) | F1N2M7 (Sub-Fam. A) | Naive + | BLAST + | GOncha + | HOGPROP1 + | HOGPROP2 + |
| GO: 0008517 | Folic acid transmembrane transporter activity | MF | F1P9N8 (Sub-Fam. A) | F6SXG7 (Sub-Fam. C) | Naive + | BLAST + | GOncha + | HOGPROP1 + | HOGPROP2 + |
| GO: 0098838 | Folate transmembrane transporter | BP | F7EDM0 (Sub-Fam. A) | C32IU7 (Sub-Fam. –) | Naive + | BLAST + | GOncha + | HOGPROP1 + | HOGPROP2 + |

Note: For each method, predictions are listed—tick indicates the method predicted, cross that it did not. Green/red colouring indicates correct/incorrect classification, respectively. Those for both thiamine and folic acid transmembrane transporter activity show that all methods fail to discriminate between these two terms. Whereas, on the term for folate transmembrane transport both BLAST and HOGPROP2 correctly classify the two proteins. These terms all have too low a frequency in UniProtKB/Swiss-Prot for the naive predictor to make predictions. Protein are referred to with UniProt identifiers, and subfamilies refer to Figure 5.

‘deep learning’ machine-learning methods show promising results, but rely heavily on training sets consisting of both positive and negative examples.

More specifically, this study also provides guidance to curation, by quantifying which individual Gene Ontology terms—positive or negative—are most valuable for benchmarking. Whilst positive associations become more informative the further they are away from the root-terms, negative annotations are more informative the closer they are to the root-terms. Negating particularly general terms may prove prohibitively difficult to experimentally validate. This also explains why only using general terms in a benchmark is not merely uninformative (Clark and Radivojac, 2013; Gaudet et al., 2017; Pesquita, 2017; Skunca et al., 2017), but misleading.

When weighting by IC it is possible to correct for differences within and between protein annotation sets. It does not, however, provide a balanced test—especially if only general terms are used. The balanced OWA-compliant benchmark provides a balanced test set such that methods are only rewarded for predicting terms that can be disproved. This, alongside the relatively low IC of annotations considered in the benchmark under the closed world assumption, explains why the naive predictor performs so well in CAFA.

Finally, this work highlights the importance of the methodological details underpinning benchmarking. The absolute and relative performance of methods is enormously affected by seemingly technical decisions. Overcoming the limitations of the current benchmarks should be an overriding priority for the function prediction community.

Acknowledgements

The authors thank Pascale Gaudet, HuaIyu Mi and Paul D. Thomas for providing the relevant data from PAINT, and for their helpful feedback on the work. The authors also thank Monique Zahn for her suggestions on the manuscript.

Funding

The authors acknowledge funding by the Swiss National Science Foundation (grants 150654 and 183723) and UK BBSRC grant BB/M015009/1.

Conflict of Interest: none declared.

References

Altenhoff, A.M. et al. (2018) The OMA orthology database in 2018: retrieving evolutionary relationships among all domains of life through richer web and programmatic interfaces. Nucleic Acids Res., 46, D477–D485.

Ashburner, M. et al. (2000) Gene Ontology: tool for the unification of biology. Nat. Genet., 25, 25–29.

Barrell, D. et al. (2009) The GOA database in 2009—an integrated Gene Ontology Annotation resource. Nucleic Acids Res., 37, D396–D403.

Camacho, C. et al. (2009) BLAST+: architecture and applications. BMC Bioinformatics, 10, 421.

Clark, W.T. and Radivojac, P. (2013) Information-theoretic evaluation of predicted ontological annotations. Bioinformatics, 29, i53–i61.

Cuzzotto, D. and Jones, D.T. (2017) Computational methods for annotation transfers from sequence. In: Dessiou, C. and Skunca, N. (eds.) The Gene Ontology Handbook. Springer, New York, pp. 55–67.

Deegan, J.I. et al. (2010) Formalization of taxon-based constraints to detect inconsistencies in annotation and ontology development. BMC Bioinformatics, 11, 530.

Dessiou, C. et al. (2013) CAFA and the Open World of protein function predictions. Trends Genet. TIG, 29, 609–610.

Fu, G. et al. (2016) NegGOA: negative GO annotations selection using ontology structure. Bioinformatics, 32, 2996–3004.

Gaudet, P. and Dessiou, C. (2017) Gene ontology: pitfalls, biases, and remedies. In: Dessiou, C. and Skunca, N. (eds.) The Gene Ontology Handbook. Springer, New York, pp. 189–205.

Gaudet, P. et al. (2011) Phylogenetic-based propagation of functional annotations within the Gene Ontology Consortium. Brief. Bioinf., 12, 449–462.

Gaudet, P. et al. (2017) Primer on the gene ontology. In: Dessiou, C. and Skunca, N. (eds.) The Gene Ontology Handbook. Springer, New York, pp. 25–37.

Huntley, R.P. et al. (2014) Understanding how and why the Gene Ontology and its annotations evolve: the GO within UniProt. GigaScience, 3, 2047–2217.

Huntley, R.P. et al. (2015) The GOA database: gene ontology annotation updates for 2015. Nucleic Acids Res., 43, D1057–D1063.

Hüntener, C. et al. (2009) The impact of incomplete knowledge on evaluation: an experimental benchmark for protein function prediction. Bioinformatics, 25, 2404–2410.

Jiang, Y. et al. (2014) The impact of incomplete knowledge on the evaluation of protein function prediction: a structured-output learning perspective. Bioinformatics, 30, 609–616.

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

Lewin, H.A. et al. (2018) Earth BioGenome Project: sequencing life for the future of life. Proc. Natl. Acad. Sci. USA, 115, 4325–4333.

Martin, D.M. et al. (2004) Gotcha: a new method for prediction of protein function assessed by the annotation of seven genomes. BMC Bioinformatics, 5, 178.

Mukherjee, S. et al. (2019) Genomes OnLine database (GOLD) v.7: updates and new features. Nucleic Acids Res., 47, D649–D659.

Muruganujan, A. et al. (2012) PANTHER in 2013: modeling the evolution of gene function, and other gene attributes, in the context of phylogenetic trees. Nucleic Acids Res., 41, D377–D386.
Pesquita, C. (2017). Semantic similarity in the gene ontology. In: Dessimoz, C. and Skunca, N. (eds.) The Gene Ontology Handbook. Springer, New York, pp. 161–173.

Radivojac, P. et al. (2013) A large-scale evaluation of computational protein function prediction. Nat. Methods, 10, 221–227.

Shannon, C. E. (1948) A mathematical theory of communication. Bell Syst. Tech. J., 27, 379–423.

Skunca, N. et al. (2012) Quality of computationally inferred gene ontology annotations. PLoS Comput. Biol., 8, e1002533.

Skunca, N. et al. (2017) Evaluating computational gene ontology annotations. In: Dessimoz, C. and Skunca, N. (eds.) The Gene Ontology Handbook. Springer, New York, pp. 97–109.

Tang, H. et al. (2018) Gotaxon: representing the evolution of biological functions in the gene ontology. arXiv preprint arXiv:1802.06004.

The Gene Ontology Consortium. (2017) Expansion of the Gene Ontology knowledgebase and resources. Nucleic Acids Res., 45, D331–D338.

The Gene Ontology Consortium. (2018) The gene ontology resource: 20 years and still going strong. Nucleic Acids Res., 47, D330–D338.

The UniProt Consortium. (2017) UniProt: the universal protein knowledgebase. Nucleic Acids Res., 45, D158–D169.

UniProt Consortium. (2018) Uniprot: a worldwide hub of protein knowledge. Nucleic Acids Res., 47, D506–D515.

Thomas, P. D. (2017). The gene ontology and the meaning of biological function. In: Dessimoz, C. and Skunca, N. (eds.) The Gene Ontology Handbook. Springer, New York, pages 15–24.

Thomas, P. D. et al. (2012) On the use of gene ontology annotations to assess functional similarity among orthologs and paralogs: a short report. PLoS Comput. Biol., 8, e1002386.

Valentini, G. (2009) True path rule hierarchical ensembles. In: Benediktsson J.A., Kittler J., Roli F. (eds) Multiple Classifier Systems. MCS 2009. Lecture Notes in Computer Science, vol 5519. Springer, Berlin, Heidelberg.

Warwick Vesztrocy, A. et al. (2018) Prioritising candidate genes causing QTL using hierarchical orthologous groups. Bioinformatics, 34, i612–i619.

Youngs, N. et al. (2014) Negative example selection for protein function prediction: the NoGO database. PLoS Comput. Biol., 10, e1003644.

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