BIOINFORMATICS

Vol. 29 ISMB/ECCB 2013, pages i53–i61
doi:10.1093/bioinformatics/btt228

# Information-theoretic evaluation of predicted ontological annotations

Wyatt T. Clark and Predrag Radivojac*

Department of Computer Science and Informatics, Indiana University, Bloomington, IN 47405, USA

**ABSTRACT**

**Motivation:** The development of effective methods for the prediction of ontological annotations is an important goal in computational biology, with protein function prediction and disease gene prioritization gaining wide recognition. Although various algorithms have been proposed for these tasks, evaluating their performance is difficult owing to problems caused both by the structure of biomedical ontologies and biased or incomplete experimental annotations of genes and gene products.

**Results:** We propose an information-theoretic framework to evaluate the performance of computational protein function prediction. We use a Bayesian network, structured according to the underlying ontology, to model the prior probability of a protein's function. We then define two concepts, misinformation and remaining uncertainty, that can be seen as information-theoretic analogs of precision and recall. Finally, we propose a single statistic, referred to as semantic distance, that can be used to rank classification models. We evaluate our approach by analyzing the performance of three protein function predictors of Gene Ontology terms and provide evidence that it addresses several weaknesses of currently used metrics. We believe this framework provides useful insights into the performance of protein function prediction tools.

**Contact:** predrag@indiana.edu

**Supplementary information:** Supplementary data are available at *Bioinformatics* online.

## 1 INTRODUCTION

Ontological representations have been widely used in biomedical sciences to standardize knowledge representation and exchange (Robinson and Bauer, 2011). Modern ontologies are typically viewed as graphs in which vertices represent terms or concepts in the domain of interest, and edges represent relational ties between terms (e.g. is-a, part-of). Although, in theory, there are no restrictions on the types of graphs used to implement ontologies, hierarchical organizations, such as trees or directed acyclic graphs, have been frequently used in the systematization of biological experiments, organismal phenotypes or structural and functional descriptions of biological macromolecules.

In molecular biology, one of the most frequently used ontologies is the Gene Ontology (GO) (Ashburner *et al.*, 2000), which standardizes the functional annotation of genes and gene products. The development of GO was based on the premise that the genomes of all living organisms are composed of genes whose products perform functions derived from a finite molecular repertoire. In addition to knowledge representation, GO has also facilitated large-scale analyses and automated annotation of gene product function (Radivojac *et al.*, 2013). As the rate of accumulation of uncharacterized sequences far outpaces the rate at which biological experiments can be carried out to characterize those sequences, computational function prediction has become increasingly useful for the global characterization of genomes and proteomes as well as for guiding biological experiments via prioritization (Rentzsch and Orengo, 2009; Sharan *et al.*, 2007).

The growing importance of tools for the prediction of GO annotations, especially for proteins, presents the problem of how to accurately evaluate such tools. First, because terms can automatically be associated with their ancestors in the GO graph, the task of an evaluation procedure is to compare the predicted graph with the true experimental annotation. Furthermore, the structure of the ontology introduces dependence between terms, which must be appropriately considered when comparing two graphs. Second, GO, as most current ontologies, is generally unfinished and contains a range of specificities of functional descriptions at the same depth of the ontology (Alterovitz *et al.*, 2010). Third, protein function is complex and context dependent; thus, a single biological experiment rarely results in complete characterization of a protein's function. This is particularly evident in cases when only high-throughput experiments are used for functional characterization, leading to shallow annotation graphs. This poses a problem in evaluation, as the ground truth is incomplete and noisy. Finally, different computational models produce different outputs that must be accounted for. For example, some models simply predict an annotation graph, possibly associating it with a numerical score, whereas others assign a score to potentially each node in the ontology, with an expectation that a good decision threshold would be applied to provide useful annotations.

There are two important factors related to the development of evaluation metrics. First, because both the experimental and predicted annotation of genes can be represented as subgraphs of the generally much larger GO graph, it is unlikely that a given computational method will provide an exact prediction of the experimental annotation. Thus, it is necessary to develop metrics that facilitate calculating degrees of similarity between pairs of graphs and appropriately address dependency between nodes. Ideally, such a measure of similarity would be able to characterize not only the level of correct prediction of the true (albeit incomplete) annotation but also the level of misannotation. The second important factor related to the evaluation metric is its interpretability. This is because characterizing the predictor's performance should be meaningful to a downstream user. Ideally, an evaluation metric would have a simple probabilistic interpretation.

In this article, we develop an information-theoretic framework for evaluating the prediction accuracy of computer-generated ontological annotations. We first use the structure of the

*To whom correspondence should be addressed.

ontology to probabilistically model, via a Bayesian network, the prior distribution of protein experimental annotation. We then apply our metric to three protein function prediction algorithms selected to highlight the limitations of typically considered evaluation metrics. We show that our metrics provide added value to the current analyses of the strengths and weaknesses of computational tools. Finally, we argue that our framework is probabilistically well founded and show that it can also be used to augment already existing evaluation metrics.

## 2 BACKGROUND

The issue of performance evaluation is closely related to the problems of measuring similarity between pairs of graphs or sets. First, we note that a protein's annotation (experimental or predicted) is a graph containing a subset of nodes in the ontology together with edges connecting them. We use the term *leaf node* to describe a node that has no descendants in the annotation graph, although it is allowed to have descendants in the ontology. A set of leaf terms completely describes the annotation graph.

We roughly group both graph similarity and performance evaluation metrics into topological and probabilistic categories and note that a particular metric may combine aspects from both. More elaborate distinctions are provided by Guzzi *et al.* (2012) and Pesquita *et al.* (2009). Topological metrics rely on the structure of the ontology to perform evaluation and typically use metrics that operate on sets of nodes and/or edges. A number of topological measures have been used, including the Jaccard and cosine similarity coefficients (the cosine approach initially maps the binary term designations into a vector space), the shortest path-based distances (Rada *et al.*, 1989) and so forth. In the context of classifier performance analysis, two common 2D metrics are the precision/recall curve and the Receiver Operating Characteristic (ROC) curve. Both curves are constructed based on the overlap in either edges or nodes between true and predicted terms and have been widely used to evaluate the performance of tools for the inference of GO annotations. They can also be used to provide a single statistic to rank classifiers through the maximum F-measure in the case of precision/recall curve or the area under the ROC curve. The area under the ROC curve has a limitation arising from the fact that the ontology is relatively large, but that the number of terms associated with a typical protein is relatively small. In practice, this results in specificities close to one, regardless of the prediction, as long as the number of predicted terms is relatively small.

Although these statistics provide good feedback regarding multiple aspects of a predictor's performance, they do not always address node dependency or the problem of unequal specificity of functional annotations found at the same depth of the graph. Coupled with a large bias in the distribution of terms among proteins, prediction methods that simply learn the prior distribution of terms in the ontology could appear to have better performance than they actually do.

The second class of similarity/performance measures is probabilistic or information-theoretic metrics. Such measures assume an underlying probabilistic model over the ontology and use a database of proteins to learn the model. Similarity is then assessed by measuring the information content of the shared terms in the ontology but can also take into account the information content of the individual annotations. Unlike with topological measures where updates to the ontology affect similarity between objects, information-theoretic measures are also affected by changes in the underlying probabilistic model even if the structure of the ontology remains the same.

Probabilistic metrics closely follow and extend the methodology laid out by Resnik (1995), which is based on the notion of information content between a pair of individual terms. These measures overcome biases related to the structure of the ontology; however, they have several drawbacks of their own. One that is especially important in the context of analyzing the performance of a predictor is that they only report a single statistic, namely, the similarity or distance between two terms or sets of terms. This ignores the tradeoff between precision and recall that any predictor has to make. In the case of Resnik's metric, a prediction by any descendant of the true term will be scored as if it is an exact prediction. Similarly, a shallow prediction will be scored the same as a prediction that deviates from the true path at the same point, regardless of how deep the erroneous prediction might be. Although some of these weaknesses have been corrected in subsequent work (Jiang and Conrath, 1997; Lin, 1998; Schlicker *et al.*, 2006), there remains the issue that the available probabilistic measures of semantic similarity resort to *ad hoc* solutions to address the common situation where proteins are annotated by graphs that contain multiple leaf terms (Clark and Radivojac, 2011). Various approaches have been taken, including averaging between all pairs of leaf terms (Lord *et al.*, 2003), finding the maximum among all pairs (Resnik, 1999) or finding the best-match average, but each such solution lacks strong justification in general. For example, all-pair averaging leads to anomalies where the exact prediction of an annotation containing a single leaf term $u$ would be scored higher than the exact prediction of an annotation containing two distinct leaf terms $u$ and $v$ of equal information content, when it is more natural to think that the latter prediction should be scored higher. Finally, certain semantic similarity metrics that incorporate pairwise matching between leaf terms tacitly assume that the objects to be compared are annotated by similar numbers of leaf terms. As such, they could produce undesirable solutions when applied to a wide range of prediction algorithms such as those outputting a large number of predicted terms.

## 3 METHODS

Our objective here is to introduce information-theoretic metrics for evaluating classification performance in protein function prediction. In this learning scenario, the input space $\mathcal{X}$ represents proteins, whereas the output space $\mathcal{Y}$ contains directed acyclic graphs describing protein function according to GO. Because of the hierarchical nature of GO, both experimental and computational annotations need to satisfy the *consistency requirement*, i.e. if an object $x \in \mathcal{X}$ is assigned a node (functional term) $v$ from the ontology, it must also be assigned all of the ancestors of $v$ up to the root(s). Therefore, the task of a classifier is to assign the best consistent subgraph of the ontology to each new protein and output a prediction score for this subgraph and/or each predicted term.

We only consider consistent subgraphs as descriptions of function and simplify the exposition by referring to such graphs as prediction or annotation graphs. In addition, we frequently treat consistent graphs as sets of nodes or functional terms and use set operations to manipulate them.

We now proceed to provide a definition for the information content of a (consistent) subgraph in the ontology. Then, using this definition, we derive information-theoretic performance evaluation metrics for comparing pairs of graphs.

### 3.1 Calculating the information content of a graph

Let each term in the ontology be a binary random variable and consider a fixed but unknown probability distribution over $\mathcal{X}$ and $\mathcal{Y}$ according to which the quality of a prediction process will be evaluated. We shall assume that the prior distribution of a target can be factorized according to the structure of the ontology, i.e. we assume a Bayesian network as the underlying data generating process for the target variable. According to this assumption, each term is independent of its ancestors, given its parents and, thus, the full joint probability can be factorized as a product of individual terms obtained from the set of conditional probability tables associated with each term (Koller and Friedman, 2009). Here, we are only interested in marginal probabilities that a protein is experimentally associated with a consistent subgraph $T$ in the ontology. This probability can be expressed as

$$\Pr(T) = \prod_{v \in T} \Pr(v|\mathcal{P}(v)), \tag{1}$$

where $v$ denotes a node in a graph and $\mathcal{P}(v)$ is the set of parent nodes of $v$. Here, Equation (1) can be derived from the full joint factorization by first marginalizing over the leaves of the ontology and then moving towards the root(s) for all nodes not in $T$.

The information content of a subgraph can be thought of as the number of bits of information one would receive about a protein if it were annotated with that particular subgraph. We calculate the information content of a subgraph $T$ in a straightforward manner as

$$i(T) = \log \frac{1}{\Pr(T)}$$

and use a base 2 logarithm as a matter of convention. The information content of a subgraph $T$ can now be expressed by combining the previous two equations as

$$\begin{aligned} i(T) &= \sum_{v \in T} \log \frac{1}{\Pr(v|\mathcal{P}(v))} \\ &= \sum_{v \in T} ia(v), \end{aligned}$$

where, to simplify the notation, we use $ia(v)$ to represent the negative logarithm of $\Pr(v|\mathcal{P}(v))$. Term $ia(v)$ can be thought of as the increase, or accretion, of information obtained by adding a child term to a parent term, or set of parent terms, in an annotation. We will refer to $ia(v)$ as *information accretion* (perhaps information gain would be a better term, but because it is frequently used in other applications to describe an expected reduction in entropy, we avoid it in this situation).

A simple ontology containing five terms together with a conditional probability table associated with each node is shown in Figure 1A. Because of the graph consistency requirement, each conditional probability table is limited to a single number. For example, at node $b$ in the graph, the probability $\Pr(b = 1|a = 1)$ is the only one necessary because $\Pr(b = 0|a = 1) = 1 - \Pr(b = 1|a = 1)$ and because $\Pr(b = 1|a = 0)$ is guaranteed to be 0. In Figure 1B, we show a sample dataset of four proteins functionally annotated according to the distribution defined by the Bayesian network. In Figure 1C, we show the total information content for each of the four annotation graphs.

### 3.2 Comparing two annotation graphs

We now consider a situation in which a protein's true and predicted function is represented by graphs $T$ and $P$, respectively. We define two metrics that can be thought of as the information-theoretic analogs of

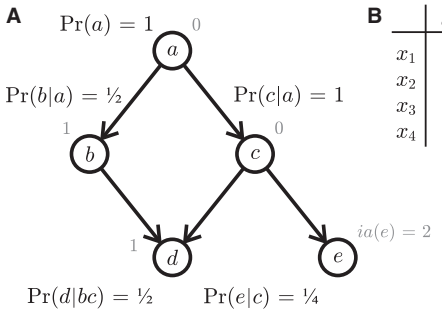


**Fig. 1.** An example of an ontology, dataset and calculation of information content. (**A**) An ontology viewed as a Bayesian network together with a conditional probability table assigned to each node. Each conditional probability table is limited to a single number owing to the consistency requirement in assignments of protein function. Information accretion calculated for each node, e.g. $ia(e) = -\log \Pr(e|c) = 2$, are shown in gray next to each node. (**B**) A dataset containing four proteins whose functional annotations are generated according to the probability distribution from the Bayesian network. (**C**) The total information content associated with each protein found in panel (B); e.g. $i(ace) = ia(a) + ia(c) + ia(e) = 2$. Note that $i(ab) = 1$ and $i(abcde) = 4$, although proteins with such annotation have not been observed in part (B)

recall and precision and refer to them as remaining uncertainty and misinformation, respectively.

Definition 1. The *remaining uncertainty* about the protein's true annotation corresponds to the information about the protein that is not yet provided by the graph $P$. More formally, we express the remaining uncertainty ($ru$) as

$$ru(T, P) = \sum_{v \in T - P} ia(v)$$

which is simply the total information content of the nodes in the ontology that are contained in true annotation $T$, but not in the predicted annotation $P$. In a slight abuse of notation, we apply set operations to graphs to manipulate only the vertices of these graphs.

Definition 2. The *misinformation* introduced by the classifier corresponds to the total information content of the nodes along incorrect paths in the prediction graph $P$. More formally, the misinformation is expressed as

$$mi(T, P) = \sum_{v \in P - T} ia(v),$$

which quantifies how misleading a predicted annotation is.

Here, a perfect prediction (one that achieves $P = T$) leads to $ru(T, P) = 0$ and $mi(T, P) = 0$. However, both $ru(T, P)$ and $mi(T, P)$ can be infinite in the limit. In practice, though, $ru(T, P)$ is bounded by the information content of the particular annotation, whereas $mi(T, P)$ is only limited by the particular annotations a predictor chooses to return.

To illustrate calculation of remaining uncertainty and misinformation, in Figure 2, we show a sample ontology where the true annotation of a protein $T$ is determined by the two leaf terms $t_1$ and $t_2$, whereas the predicted subgraph $P$ is determined by the leaf terms $p_1$ and $p_2$. The remaining uncertainty $ru(T, P)$ and misinformation $mi(T, P)$ can now be calculated by adding the information accretion corresponding to the nodes circled in gray.

Finally, this framework can be used to define the similarity between the protein's true annotation and the predicted annotation without relying on identifying an individual common ancestor between pairs of leaves (this node is usually referred to as the maximum informative common

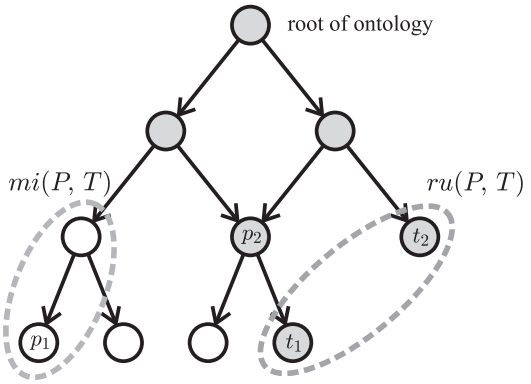


**Fig. 2.** Illustration of calculating remaining uncertainty and misinformation, given a predicted annotation graph $P$ and a graph of true annotations $T$. Graphs $P$ and $T$ are uniquely determined by the leaf nodes $p_1$, $p_2$, $t_1$, and $t_2$, respectively. Nodes colored in gray represent graph $T$. Nodes circled in gray are used to determine remaining uncertainty (*ru*; right side) and misinformation (*mi*; left side) between $T$ and $P$

ancestor; Guzzi *et al*., 2012). The information content of the subgraph shared by $T$ and $P$ is one such possibility; i.e. $s(T,P) = \sum_{v \in T \cap P} ia(v)$.

## 3.3 Measuring the quality of function prediction

A typical predictor of protein function usually outputs scores that indicate the strength (e.g. posterior probabilities) of predictions for each term in the ontology. To address this situation, the concepts of remaining uncertainty and misinformation need to be considered as a function of a decision threshold $\tau$. In such a scenario, predictions with scores greater than or equal to $\tau$ are considered positive predictions, whereas the remaining associations are considered negative (if the strength of a prediction is expressed via *P*-values or E-values, values lower than the threshold would indicate positive predictions). Regardless of the situation, every decision threshold results in a separate pair of values corresponding to the remaining uncertainty $ru(T, P(\tau))$ and misinformation $mi(T, P(\tau))$.

The remaining uncertainty and misinformation for a previously unseen protein can be calculated as expectations over the data generating probability distribution. Practically, this can be performed by averaging over the entire set of proteins used in evaluation, i.e.

$$ru(\tau) = \frac{1}{n}\sum_{i=1}^{n} ru(T_i, P_i(\tau)) \tag{2}$$

and

$$mi(\tau) = \frac{1}{n}\sum_{i=1}^{n} mi(T_i, P_i(\tau)) \tag{3}$$

where $n$ is the number of proteins in the dataset, $T_i$ is the true set of terms for protein $x_i$, and $P_i(\tau)$ is the set of predicted terms for protein $x_i$, given decision threshold $\tau$. Once the set of terms with scores greater than or equal to $\tau$ is determined, the set $P_i(\tau)$ is composed of the unique union of the ancestors of all predicted terms. As the decision threshold is moved from its minimum to its maximum value, the pairs of $(ru(\tau), mi(\tau))$ will result in a curve in 2D space. We refer to such a curve using $(ru(\tau), mi(\tau))_\tau$. Removing the normalizing constant $(\frac{1}{n})$ from the aforementioned equations would result in the total remaining uncertainty and misinformation associated with a database of proteins and a set of predictions.

*3.3.1 Weighted metrics* One disadvantage of definitions in Equations (2) and (3) is that an equal weight is given to proteins with low and high information content annotations when averaging. To address this, we assign a weight to each protein according to the information content of its experimental annotation. This formulation naturally downweights proteins with less informative annotations compared with proteins with rare, and therefore more informative (surprising), annotations. In biological datasets, frequently seen annotations have a tendency to be incomplete or shallow annotation graphs and arise owing to the limitations or high-throughput nature of some experimental protocols. We define *weighted remaining uncertainty* as

$$wru(\tau) = \frac{\sum_{i=1}^{n} i(T_i) \cdot ru(T_i, P_i(\tau))}{\sum_{i=1}^{n} i(T_i)} \tag{4}$$

and *weighted misinformation* as

$$wmi(\tau) = \frac{\sum_{i=1}^{n} i(T_i) \cdot mi(T_i, P_i(\tau))}{\sum_{i=1}^{n} i(T_i)} \tag{5}$$

*3.3.2 Semantic distance* Finally, to provide a single performance measure, which can be used to rank and evaluate protein function prediction algorithms, we introduce *semantic distance* as the minimum distance from the origin to the curve $(ru(\tau), mi(\tau))_\tau$. More formally, the semantic distance $S_k$ is defined as

$$S_k = \min_{\tau}(ru^k(\tau) + mi^k(\tau))^{\frac{1}{k}}, \tag{6}$$

where $k$ is a real number greater than or equal to one. Setting $k = 2$ results in the minimum Euclidean distance from the origin. The preference for Euclidean distance ($k = 2$) over say Manhattan distance ($k = 1$) is to penalize unbalanced predictions with respect to the depth of predicted and experimental annotations.

## 3.4 Precision and recall

To contrast the semantic distance-based evaluation with more conventional performance measures, in this section, we briefly introduce precision and recall for measuring functional similarity. As before, we consider a set of propagated experimental terms $T$ and predicted terms $P(\tau)$ and define precision as the fraction of terms predicted correctly. More specifically,

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

where $|\cdot|$ is the set cardinality operator. Only proteins for which the prediction set is non-empty can be used to calculate average precision. To address this issue, the root term is counted as a prediction for all proteins. Similarly, recall is defined as the fraction of experimental (true) terms, which were correctly predicted, i.e.

$$rc(T, P(\tau)) = \frac{|T \cap P(\tau)|}{|T|}.$$

As before, precision $pr(\tau)$ and recall $rc(\tau)$ for the entire dataset are calculated as averages over the entire set of proteins [an alternative definition of precision and recall is given by Verspoor *et al*. (2006)]. Finally, to provide a single evaluation measure, we use the maximum F-measure over all decision thresholds. For a particular set of terms $T$ and $P(\tau)$, F-measure is calculated as the harmonic mean of precision and recall. More formally, the final evaluation metric is calculated as

$$F_{\max} = \max_{\tau}\left\{2 \cdot \frac{pr(\tau) \cdot rc(\tau)}{pr(\tau) + rc(\tau)}\right\}$$

where $pr(\tau)$ and $rc(\tau)$ are calculated by averaging over the dataset.

*3.4.1 Information-theoretic weighted formulation* The definition of information accretion and the use of a probabilistic framework defined by the Bayesian network enables the straightforward application of information accretion to weight each term in the ontology. Therefore, it is easy to generalize the definitions of precision and recall from the previous section into a weighted formulation. Here, weighted precision and weighted recall can be expressed as

$$wpr(T, P(\tau)) = \frac{\sum_{v \in T \cap P(\tau)} ia(v)}{\sum_{v \in P(\tau)} ia(v)}$$

and

$$wrc(T, P(\tau)) = \frac{\sum_{v \in T \cap P(\tau)} ia(v)}{\sum_{v \in T} ia(v)}.$$

Weighted precision $wpr(\tau)$ and recall $wrc(\tau)$ can then be calculated as weighted averages over the database of proteins, as in Equations (4) and (5).

## 4 EXPERIMENTS AND RESULTS

In this section, we fist analyze the average information content in a dataset of experimentally annotated proteins and then evaluate performance accuracy of different function prediction methods using both topological and probabilistic metrics. Each experiment was conducted on all three categories of the GO: Molecular Function (MFO), Biological Process (BPO) and Cellular Component (CCO) ontologies. To avoid cases where the information content of a term is infinite, a pseudo-count of one was added to each term, and the total number of proteins in the dataset was incremented when calculating term frequencies.

### 4.1 Data, prediction models and evaluation

We first collected all proteins with GO annotations supported by experimental evidence codes (EXP, IDA, IPI, IMP, IGI, IEP, TAS, IC) from the January 2011 version of the Swiss-Prot database (29 699 proteins in MFO, 31 608 in BPO and 30 486 in CCO). We then generated three simple function annotation models: Naive, BLAST and GOtcha, to assess the ability of performance metrics to accurately reflect the quality of a predicted set of annotations. In addition to these three methods, we generated another set of 'predictions' by collecting experimental annotations for the same set of proteins from a database generated by the GO Consortium released at about the same time as our version of Swiss-Prot. This was done to quantify the variability of experimental annotation across different databases using the same set of metrics. In addition, this comparison can be used to estimate the empirical upper limit of prediction accuracy because the observed performance is limited by the noise in experimental data. All computational methods were evaluated using 10-fold cross-validation.

The Naive model was designed to reflect biases in the distribution of terms in the dataset and was the simplest annotation model we used. It was generated by first calculating the relative frequency of each term in the training dataset. This value was then used as the prediction score for every protein in the test set; thus, every protein in the test partition was assigned an identical set of predictions over all functional terms. The performance of the Naive model reflects what one could expect when annotating a protein with no knowledge about that protein.

The BLAST model was generated using local sequence identity scores to annotate proteins. Given a target protein sequence $x$, a particular functional term $v$ in the ontology, and a set of sequences $S_v = \{s_1, s_2, \ldots\}$ annotated with term $v$, we determine the BLAST predictor score for function $v$ as $\max\{sid(x, s) : s \in S_v\}$, where $sid(x, s)$ is the maximum sequence identity returned by the BLAST package (Altschul *et al*., 1997) when the two sequences are aligned. We chose this method to mimic the performance one would expect if they simply used BLAST to transfer annotations between similar sequences.

The third method, GOtcha (Martin *et al*., 2004), was selected to incorporate not only sequence identity between protein sequences but also the structure of the ontology (technically, BLAST also incorporates structure of the ontology but in a relatively trivial manner). Specifically, given a target protein $x$, a particular functional term $v$, and a set of sequences $S_v = \{s_1, s_2, \ldots\}$ annotated with function $v$, one first determines the r-score for function $v$ as $r_v = c - \sum_{s \in S_v} \log(e(x, s))$, where $e(x, s)$ represents the E-value of the alignment between the target sequence $x$ and sequence $s$, and $c = 2$ is a constant added to the given quantity to ensure all scores were above 0. Given the r-score for function $v$, i-scores were then calculated by dividing the r-score of each function by the score for the root term $i_v = r_v / r_{root}$. As such, GOtcha is an inexpensive and robust predictor of function.

### 4.2 Average information content of a protein

We first examined the distribution of the information content per protein for each of the three ontologies (Fig. 3). We observe a wide range of information contents in all ontologies, reaching over 128 bits in case of BPO (which corresponds to a factor of 128 in the probability of observing particular annotation graphs). The distributions for MFO and CCO show unusual peaks for low information contents, suggesting that a large fraction of annotation graphs in these ontologies are low quality. One such anomaly is created by the term 'binding' in MFO that is associated with 72% of proteins. Furthermore, 41% of proteins are annotated with its child 'protein binding' as a leaf term, and 26% are annotated with it as their sole leaf term. Such annotations, which are clearly a consequence of high-throughput experiments, present a significant difficulty in method evaluation.

Previously, we showed that the distribution of leaf terms in protein annotation graphs exhibits scale-free tendencies (Clark and Radivojac, 2011). Here, we also analyzed the average number of leaf terms per protein and compared it with the information content of that protein. We estimate the average number of leaf terms to be 1.6 (std. 1.0), 3.0 (std. 3.6) and 1.6 (std. 1.0) for MFO, BPO and CCO, respectively, and calculate Pearson correlation between the information content and the number of leaf terms for a protein (0.80, 0.92 and 0.71). Such high level of correlation suggests that proteins annotated with a small number of leaf terms are generally annotated by shallow graphs. This is particularly evident in the case of 'protein binding' annotations that can be derived from yeast-2-hybrid experiments but provide little insight into the functional aspects of these complexes when only viewed as GO annotations. We believe the wide range of information contents coupled

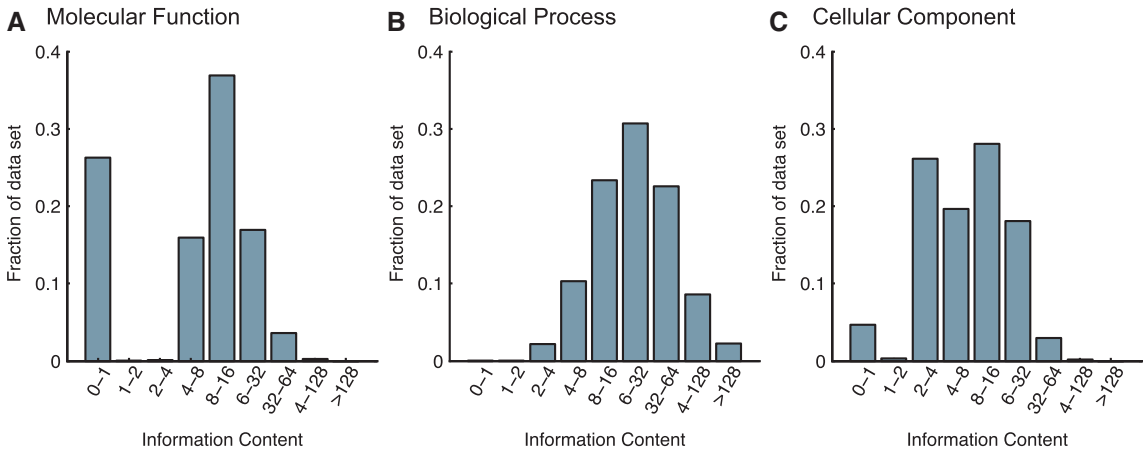


**Fig. 3.** Distribution of information content (in bits) over proteins annotated by terms for each of the three ontologies. The average information content of a protein was estimated at 10.9 (std. 10.2), 32.0 (std. 33.6) and 10.4 (std. 9.2) bits for MFO, BPO and CCO, respectively

with the fact that a large fraction of proteins were essentially uninformative, justifies the weighting proposed in this work.

### 4.3 2D plots

To assess how each metric evaluated the performance of the four prediction methods, we generated 2D plots. Figure 4 shows the performance of each predictor using precision/recall and ru-mi curves, as well as their weighted variants [additional precision/recall curves using the definition by Verspoor *et al*. (2006) as well as additional ru-mi curves are provided in Supplementary Materials]. The performance of the GO/Swiss-Prot annotation is represented as a single point because it compares two databases of experimental annotations.

When looking at the precision/recall curves, we first observe an unusually high area under the curve associated with the Naive model. This is a result of a significant fraction of low information content annotations that are relatively easy to predict by simply using prior probabilities of terms as prediction values. In addition, these biases lead to a biologically unexpected result where the predictor based on the BLAST algorithm performs on par with the Naive model, e.g. $F_{\max}$(BLAST, MFO) $= 0.65$ and $F_{\max}$(Naive, MFO) $= 0.60$, whereas $F_{\max}$(BLAST, CCO) $= 0.63$; $F_{\max}$(Naive, CCO) $= 0.64$. The largest difference between the BLAST and Naive models was observed for BPO, which has a Gaussian-like distribution of information contents in the logarithmic scale (Fig. 3). The second column of plots in Figure 4 shows the weighted precision/recall curves. Here, we observe large changes in the performance accuracy, especially for the Naive model, in MFO and CCO categories, whereas the BPO category was, for the most part, not impacted. We believe that the information-theoretic weighting of precision and recall resulted in more meaningful evaluation.

The information-theoretic measures are shown in the last two columns of Figure 4. One useful property of ru-mi plots is that they explicitly illustrate how many bits of information are yet to be revealed about a protein (on average) as a function of misinformation that is introduced by over-prediction or misannotation. In all three categories, the amount of misinformation being introduced increases rapidly; quickly obtaining a rate that is twice the amount of expected information for an average protein. We believe these plots shed new light into how much information overload a researcher can be presented with by drawing predictions at a particular threshold. Looking from right to left in each plot, we observe an elbow in each of the curves (at ~3 bits for MFO and CCO and 12 bits for BPO; Fig. 4) after which the remaining uncertainty barely decreases, whereas misinformation grows out of control.

### 4.4 Comparisons of single statistics

Here, we analyze the ability of the single measures to rank predictors and lead to useful evaluation insights. We compare the performance of semantic distance to several other methods that calculate either topological or semantic similarities. For each evaluation method, the decision threshold was varied for each of the prediction methods, and the threshold providing the best performance was selected as optimal. We then analyze and discuss the performance of these metrics at those optimal thresholds.

We implemented the semantic similarity metrics of Jiang and Conrath (1997), Lin (1998), Resnik (1995) and Schlicker *et al*. (2006), as detailed in Supplementary Materials. Because each of these measures is defined for a pair of terms in the ontology, scores between two protein annotation graphs (true graph $T$ versus predicted graph $P$) were obtained by averaging scores over all pairs of leaf terms $(t, p)$ such that $t \in T$ and $p \in P$. We refer to such scoring as all-pair averaging and note that the all-pair averaging using Resnik's term similarity was implemented by Lord *et al*. (2003) in the context of GO annotations. The results for a best-match averaging (also referred to as max-average method) are presented in the Supplementary Materials. In addition to these semantic measures, we also implemented the Jaccard similarity coefficient between the sets of vertices in the two annotation graphs (Supplementary Materials). In terms of precision/recall curve and ru-mi curve, we used $F_{\max}$ and $S_2$ measures to obtain optimal thresholds.

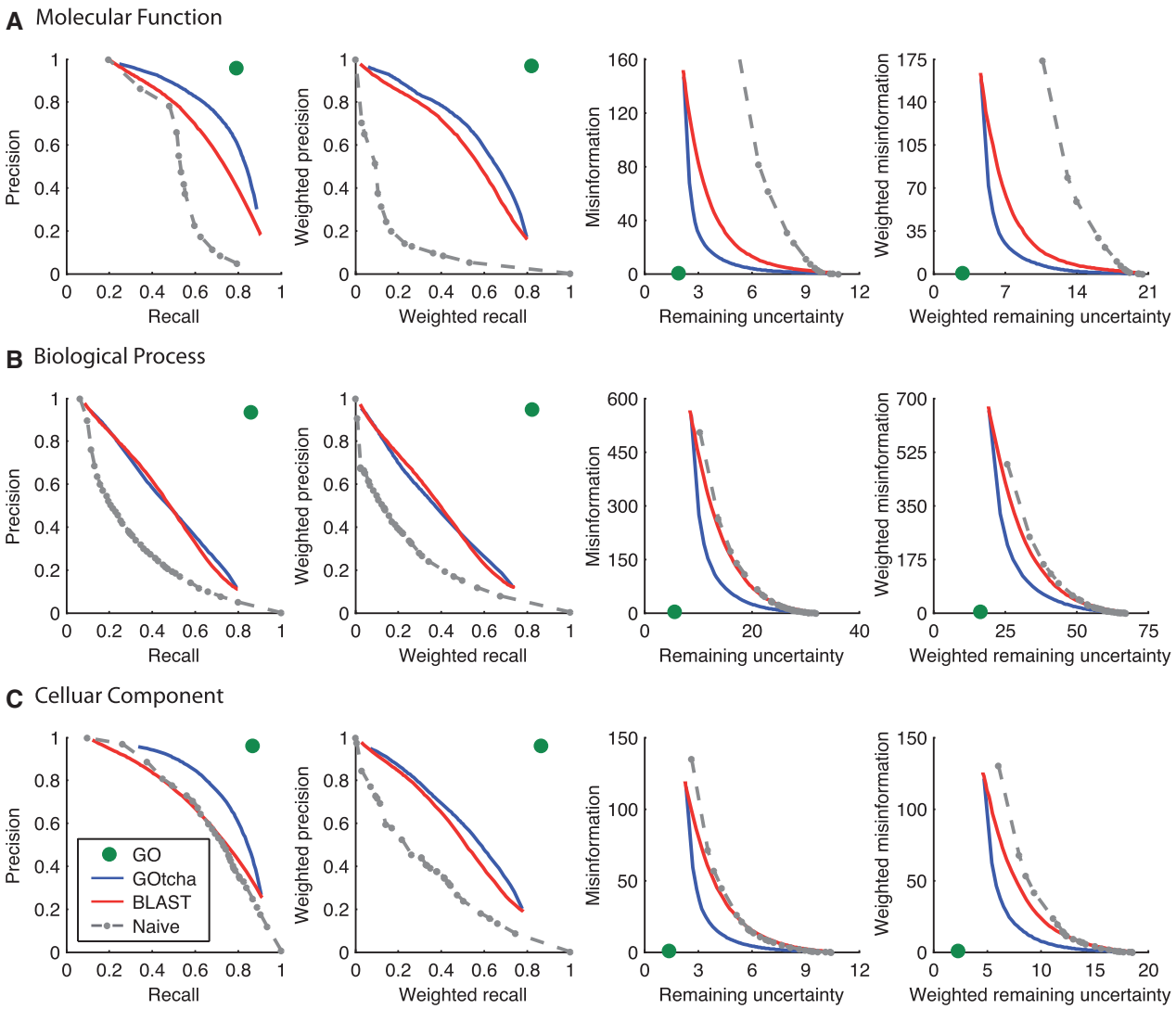


**Fig. 4.** The 2D evaluation plots. Each plot shows three prediction methods: Naive (gray, dashed), BLAST (red, solid) and GOtcha (blue, solid) constructed using cross-validation. Green point labeled GO shows the performance evaluation between two databases of experimental annotations, downloaded at the same time. The rows show the performance for different ontologies (MFO, BPO, CCO). The columns show different evaluation metrics: $(pr(\tau), rc(\tau))_\tau$, $(wpr(\tau), wrc(\tau))_\tau$, $(ru(\tau), mi(\tau))_\tau$ and $(wru(\tau), wmi(\tau))_\tau$

Table 1 shows the maximum similarity, or minimum distance in the case of Jiang and Conrath's and semantic distance, that each metric obtained for each of our classification models. In addition to reporting the maximum similarity, we also report the decision threshold at which that value was obtained along with the associated level of remaining uncertainty and misinformation at that threshold. The first interesting observation is that all metrics, aside from that of Jiang and Conrath, obtain optimal thresholds that result in relatively similar levels of remaining uncertainty and misinformation for the GOtcha model. However, all metrics, aside from semantic distance and Jiang and Conrath's distance, seem to favor extremely high levels of misinformation at the reported decision thresholds for the BLAST model. For MFO and CCO, the semantic similarity measures of Lord *et al.*, Lin and Sclicker *et al.* report misinformation levels that are more than twice the information content of the average protein in that ontology for the BLAST model. In BPO, those are even more extreme. We believe this is a direct consequence of the pairwise term averaging applied in these methods.

It is particularly interesting to analyze the optimal thresholds obtained for the BLAST model. These thresholds can be interpreted as the level of sequence identity above which each metric reports functional transfer can be made. For example, because their optimal BLAST thresholds are relatively low, the levels of misinformation provided by the similarities of Lord *et al.*, Lin and Schlicker *et al.* are rather large. $F_{\max}$ and Jaccard approaches also report low threshold values for all ontologies, whereas Jiang and Conrath's distance selects the optimal threshold at an overly restrictive 100% sequence identity. We believe that the semantic distance $S_2$ provides more reasonable values for functional transfer, finding an optimal distance at 77, 88 and 78% for MFO, BPO and CCO, respectively.

**Table 1.** Performance evaluation of several information-theoretic and topological metrics

| | Molecular Function | | | | Biological Process | | | | Cellular Component | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Lord *et al.* (2003) | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* |
| GOtcha | 2.34 | 0.47 | 6.34 | 3.20 | 1.95 | 0.40 | 23.36 | 11.90 | 1.80 | 0.36 | 5.88 | 4.58 |
| BLAST | 1.61 | 0.43 | 4.69 | 27.90 | 1.40 | 0.43 | 16.73 | 139.57 | 1.27 | 0.38 | 4.42 | 37.24 |
| Naive | 0.46 | 0.09 | 9.56 | 4.23 | 0.63 | 0.01 | 10.35 | 504.88 | 0.75 | 0.07 | 5.81 | 16.34 |
| Lin (1998) | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* |
| GOtcha | 0.44 | 0.52 | 6.67 | 2.67 | 0.26 | 0.46 | 24.43 | 9.40 | 0.41 | 0.50 | 6.71 | 2.76 |
| BLAST | 0.22 | 0.43 | 4.69 | 27.90 | 0.16 | 0.43 | 16.73 | 139.57 | 0.23 | 0.40 | 4.78 | 30.45 |
| Naive | 0.37 | 0.30 | 10.39 | 0.21 | 0.12 | 0.12 | 24.92 | 23.14 | 0.26 | 0.31 | 8.98 | 1.32 |
| Schlicker *et al.* (2006) | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* |
| GOtcha | 0.29 | 0.51 | 6.60 | 2.76 | 0.23 | 0.42 | 23.73 | 10.99 | 0.30 | 0.43 | 6.31 | 3.56 |
| BLAST | 0.17 | 0.44 | 4.83 | 25.39 | 0.14 | 0.43 | 16.73 | 139.57 | 0.18 | 0.43 | 5.26 | 23.26 |
| Naive | 0.14 | 0.30 | 10.39 | 0.21 | 0.08 | 0.12 | 24.92 | 23.14 | 0.13 | 0.31 | 8.98 | 1.32 |
| Jiang and Conrath (1997) | Min | Threshold | *ru* | *mi* | Min | Threshold | *ru* | *mi* | Min | Threshold | *ru* | *mi* |
| GOtcha | 5.74 | 0.75 | 8.20 | 1.27 | 8.38 | 0.98 | 30.88 | 1.22 | 4.83 | 0.76 | 8.21 | 1.19 |
| BLAST | 6.34 | 1.00 | 10.62 | 0.43 | 8.39 | 1.00 | 31.31 | 1.40 | 5.20 | 1.00 | 10.16 | 0.35 |
| Naive | 6.19 | 0.63 | 10.53 | 0.13 | 8.24 | 0.50 | 31.75 | 0.07 | 5.01 | 0.61 | 10.13 | 0.08 |
| Jaccard | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* |
| GOtcha | 0.57 | 0.46 | 6.29 | 3.32 | 0.31 | 0.34 | 22.24 | 15.24 | 0.56 | 0.43 | 6.31 | 3.56 |
| BLAST | 0.37 | 0.50 | 5.74 | 14.72 | 0.19 | 0.50 | 19.68 | 76.98 | 0.34 | 0.43 | 5.26 | 23.26 |
| Naive | 0.46 | 0.30 | 10.39 | 0.21 | 0.17 | 0.19 | 27.53 | 9.22 | 0.47 | 0.31 | 8.98 | 1.32 |
| $F_{max}$ | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* | Max | Threshold | *ru* | *mi* |
| GOtcha | 0.72 | 0.43 | 6.12 | 3.68 | 0.49 | 0.32 | 21.84 | 16.69 | 0.73 | 0.43 | 6.31 | 3.56 |
| BLAST | 0.64 | 0.48 | 5.42 | 17.89 | 0.49 | 0.50 | 19.68 | 76.98 | 0.63 | 0.45 | 5.57 | 19.42 |
| Naive | 0.60 | 0.29 | 9.87 | 1.44 | 0.33 | 0.19 | 27.53 | 9.22 | 0.64 | 0.33 | 9.22 | 0.80 |
| $S_2$ | Min | Threshold | *ru* | *mi* | Min | Threshold | *ru* | *mi* | Min | Threshold | *ru* | *mi* |
| GOtcha | 7.11 | 0.47 | 6.34 | 3.20 | 26.14 | 0.43 | 23.91 | 10.56 | 7.23 | 0.46 | 6.48 | 3.21 |
| BLAST | 9.13 | 0.77 | 8.25 | 3.90 | 29.89 | 0.88 | 28.28 | 9.69 | 9.08 | 0.78 | 8.51 | 3.15 |
| Naive | 9.98 | 0.10 | 9.72 | 2.80 | 29.00 | 0.22 | 27.67 | 8.72 | 8.79 | 0.21 | 7.71 | 4.95 |

*Note*: For each measure, the decision threshold was varied across the entire range of predictions to obtain the maximum or minimum value (shown in column 1). The threshold at which each method reached the best value is shown in column 2. Columns 3 and 4 show the remaining uncertainty (*ru*) and misinformation (*mi*) calculated according to the Bayesian network. Each semantic similarity metric was calculated according to the relative frequencies of observing each term in the database.

## 5 DISCUSSION

In this work, we propose an information-theoretic framework for evaluating the performance of computational protein function prediction. We frame protein function prediction as a structured-output learning problem in which the output space is represented by consistent subgraphs of the GO graph. We argue that our approach directly addresses evaluation in cases where there are multiple true and predicted (leaf) terms associated with a protein by taking the structure of the ontology and the dependencies between terms induced by a hierarchical ontology into account. Our method also facilitates accounting for the high level of biased and incomplete experimental annotations of proteins by allowing for the weighting of proteins based on the information content of their annotations. Because we maintain an information-theoretic foundation, our approach is relatively immune to the potential dissociation between the depth of a term and its information content, a weakness of often-used topological metrics in this domain such as precision/recall or ROC-based evaluation. At the same time, because we take a holistic approach to considering a protein's potentially large set of true or predicted functional associations, we resolve many of the problems introduced by the practice of aggregating multiple pairwise similarity comparisons common to existing semantic similarity measures.

Although there is a long history (Resnik, 1999) and a significant body of work in the literature regarding the use of semantic similarity measures (Guzzi *et al.*, 2012; Pesquita *et al.*, 2009), to the best of our knowledge, all such metrics are based on single

statistics and are unable to provide insight into the levels of remaining uncertainty and misinformation that every predictor is expected to balance. Therefore, the methods proposed in this work extend, modify and formalize several useful information-theoretic metrics introduced during the past decades. In addition, both remaining uncertainty and misinformation have natural information-theoretic interpretations and can provide meaningful information to the users of computational tools. At the same time, the semantic distance based on these concepts facilitates not only the use of a single performance measure to evaluate and rank predictors but can also be exploited as a loss function during training.

One limitation of the proposed approach is grounded in the assumption that a Bayesian network, structured according to the underlying ontology, will perfectly model the prior probability distribution of a target variable. An interesting anomaly with this approach is that the marginal probability, and subsequently the information content, of a single term (i.e. consistent graph with a single leaf term) calculated from a Bayesian network does not necessarily match the relative term frequency in the database (instead, the conditional probability tables are estimated as relative frequencies). *Ad hoc* solutions that maintain the term information content are possible but would result in sacrificed interpretability of the metric itself. One such solution can be obtained via a recursive definition $ia(v) = i(v) - \sum_{u \in \mathcal{P}(v)} ia(u)$ and $ia(root) = 0$, where $i(v)$ is estimated directly from the database.

Finally, rationalizing between evaluation metrics is a difficult task. The literature presents several strategies where protein sequence similarity, protein–protein interactions or other data are used to assess whether a performance metric behaves according to expectations (Guzzi *et al.*, 2012). In this work, we took a somewhat different approach and showed that the demonstrably biased protein function data can be shown to provide surprising results with well-understood prediction algorithms and conventional evaluation metrics. Thus, we believe that our experiments provide evidence of the usefulness of the new evaluation metric.

## ACKNOWLEDGEMENT

The authors thank Prof. David Crandall for his comments on the manuscript, Prof. Iddo Friedberg for stimulating discussions about semantic similarity measures and four anonymous reviewers for their suggestions that improved the quality of this study.

*Funding*: This work was supported by the National Science Foundation grant DBI-0644017 and National Institutes of Health grant R01 LM009722-06A1.

*Conflict of Interest*: none declared.

## REFERENCES

Alterovitz,G. *et al.* (2010) Ontology engineering. *Nat. Biotechnol.*, **28**, 128–130.
Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
Ashburner,M. *et al.* (2000) Gene ontology: tool for the unification of biology. The gene ontology consortium. *Nat. Genet.*, **25**, 25–29.
Clark,W.T. and Radivojac,P. (2011) Analysis of protein function and its prediction from amino acid sequence. *Proteins*, **79**, 2086–2096.
Guzzi,P.H. *et al.* (2012) Semantic similarity analysis of protein data: assessment with biological features and issues. *Brief. Bioinform.*, **13**, 569–585.
Jiang,J.J. and Conrath,D.W. (1997) Semantic similarity based on corpus statistics and lexical taxonomy. In: *Proceedings of the International Conference on Research in Computational Linguistics*. Taiwan. pp. 19–33.
Koller,D. and Friedman,N. (2009) *Probabilistic Graphical Models*. The MIT Press, Cambridge, MA.
Lin,D. (1998) An information-theoretic definition of similarity. In: *Proceedings of the 15th International Conference on Machine Learning*. Morgan Kaufmann, San Francisco, CA, pp. 296–304.
Lord,P.W. *et al.* (2003) Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation. *Bioinformatics*, **19**, 1275–1283.
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.
Pesquita,C. *et al.* (2009) Semantic similarity in biomedical ontologies. *PLoS Comput. Biol.*, **5**, e1000443.
Rada,R. *et al.* (1989) Development and application of a metric on semantic nets. *IEEE Trans. Syst. Man Cybern.*, **19**, 17–30.
Radivojac,P. *et al.* (2013) A large-scale evaluation of computational protein function prediction. *Nat. Methods*, **10**, 221–227.
Rentzsch,R. and Orengo,C. (2009) Protein function prediction–the power of multiplicity. *Trends Biotechnol.*, **27**, 210–219.
Resnik,P. (1995) Using information content to evaluate semantic similarity in a taxonomy. In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, San Francisco, CA, pp. 448–453.
Resnik,P. (1999) Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.*, **11**, 95–130.
Robinson,P.N. and Bauer,S. (2011) *Introduction to Bio-Ontologies*. CRC Press, Boca Raton, FL, USA.
Schlicker,A. *et al.* (2006) A new measure for functional similarity of gene products based on gene ontology. *BMC Bioinformatics*, **7**, 302.
Sharan,R. *et al.* (2007) Network-based prediction of protein function. *Mol. Syst. Biol.*, **3**, 88.
Verspoor,K. *et al.* (2006) A categorization approach to automated ontological function annotation. *Protein Sci.*, **15**, 1544–1549.