BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 11 2013, pages 1424–1432
doi:10.1093/bioinformatics/btt160

Data and text mining

Advance Access publication April 9, 2013

Measuring gene functional similarity based on group-wise comparison of GO terms

Zhixia Teng[1,2], Maozu Guo[1,*], Xiaoyan Liu[1], Qiguo Dai[1], Chunyu Wang[1] and Ping Xuan[1,3]

[1]Department of Computer Science and Engineering, Harbin Institute of Technology, Harbin 150001, P.R. China, [2]Department of Information Management and Information System, Northeast Forestry University, Harbin 150040, P.R. China and [3]Department of Computer Science and Technology, Heilongjiang University, Harbin 150080, P.R. China

Associate Editor: Jonathan Wren

ABSTRACT

Motivation: Compared with sequence and structure similarity, functional similarity is more informative for understanding the biological roles and functions of genes. Many important applications in computational molecular biology require functional similarity, such as gene clustering, protein function prediction, protein interaction evaluation and disease gene prioritization. Gene Ontology (GO) is now widely used as the basis for measuring gene functional similarity. Some existing methods combined semantic similarity scores of single term pairs to estimate gene functional similarity, whereas others compared terms in groups to measure it. However, these methods may make error-prone judgments about gene functional similarity. It remains a challenge that measuring gene functional similarity reliably.

Result: We propose a novel method called SORA to measure gene functional similarity in GO context. First of all, SORA computes the information content (IC) of a term making use of semantic specificity and coverage. Second, SORA measures the IC of a term set by means of combining inherited and extended IC of the terms based on the structure of GO. Finally, SORA estimates gene functional similarity using the IC overlap ratio of term sets. SORA is evaluated against five state-of-the-art methods in the file on the public platform for collaborative evaluation of GO-based semantic similarity measure. The carefully comparisons show SORA is superior to other methods in general. Further analysis suggests that it primarily benefits from the structure of GO, which implies expressive information about gene function. SORA offers an effective and reliable way to compare gene function.

Availability: The web service of SORA is freely available at http://nclab.hit.edu.cn/SORA/.

Contact: maozuguo@hit.edu.cn

Received on November 12, 2012; revised on March 10, 2013; accepted on March 27, 2013

1 INTRODUCTION

In recent years, gene functional similarity has become a main hotspot in biology research. Because it is important for a variety of applications such as gene clustering (Brameier and Wiuf, 2007; Cho *et al*., 2009; Qu and Xu, 2004; Yang *et al*., 2008), protein interaction prediction and evaluation (Li *et al*., 2008; Jain and Bader, 2010; Schlicker *et al*., 2007;), gene function prediction (Chen and Xu, 2004; Jensen *et al*., 2003; Nariai *et al*., 2007) and disease gene prioritization (Chen *et al*., 2009; Mathur and Dinakarpandian, 2011; Ortutay and Vihinen, 2009; Schlicker *et al*., 2010; Yilmaz *et al*., 2009). Moreover, compared with sequence and structure similarity, functional similarity is more informative for understanding the biological roles and functions of genes.

Gene Ontology (GO) is a controlled vocabulary of terms for describing behavior of genes and their products (GO-Consortium, 2004), which is valuable to measure gene functional similarity. Gene and its products, which are collectively called gene to simplify in this article, are usually annotated with multiple terms. Functional similarity between genes can be inferred from the semantic relationships of their terms. It is considered that two genes are similar in function if their terms are similar in semantics. Accordingly, many methods based on semantic similarity have been put forward to estimate gene functional similarity. These methods could be generally classified into two categories: pairwise and group-wise (Pesquita *et al*., 2009a).

Pairwise methods measure gene functional similarity through two steps. The first step is measuring semantic similarity scores of term pairs using term comparison techniques. The most typical term comparison techniques used by these methods are Resnik's (1999), Lin's (1998), Jiang and Conrath's (1998). The second step is computing gene functional similarity based on the semantic similarity scores calculated in the first step. Some rules such as average rule (AVG), maximum rule (MAX) and best-match average rule (BMA) are used in the last step. The methods based on AVG regard the average of semantic similarity scores of all term pairs as gene functional similarity. The methods based on MAX take the maximal semantic similarity score of all term pairs as gene functional similarity. The methods based on BMA find all the best matches between the term sets and take the average of semantic similarity scores of these best matches as gene functional similarity. As Lord *et al*. (2003) made use of GO and AVG to estimate gene functional similarity, great efforts have been made in this field. In 2005, Sevilla *et al*. (2005) and Azuaje *et al*. (2005) introduced methods like Lord's, but they used MAX and BMA rather than AVG. Meanwhile, many variants of aforementioned typical term comparison techniques like GraSM (Couto *et al*., 2005), Wang's (Wang *et al*., 2007) and Pozo's (Pozo *et al*., 2008) were proposed. Recently, Couto *et al*. (2011) exploited DiShIn to update GraSM, and Yang *et al*. (2012) improved the semantic similarity between two terms by considering their common ancestors and descendants.

*To whom correspondence should be addressed.

Although pairwise methods are used widely for measuring gene functional similarity, they suffer from some limitations of combining rules. Methods based on AVG will underestimate gene functional similarity. For instance, if two genes both are annotated with two same terms, which are unrelated to each other, their functional similarity is 0.5 by these methods. In fact, they are exactly matched, and their functional similarity should be 1. Methods based on MAX will overestimate gene functional similarity. An example is that, the functional similarity between genes, which share common terms, is 1, regardless of the different terms of them. Unlike the methods aforementioned, methods based on BMA make a balance between them. Nevertheless, the pairwise methods are affected by how well the semantic similarity of single term pair is measured. The detailed discussion of these methods can be referred to several reviews (Pesquita *et al*., 2009a; Guzzi *et al*., 2011).

Group-wise methods estimate gene functional similarity by comparing the terms in groups. These methods are categorized as follwos: set-based, graph-based and vector-based. Set-based methods (Batet *et al*., 2011; Gentleman *et al*., 2005; Lee *et al*., 2004; Martin *et al*., 2004; Mistry and Pavlidis, 2008; Pesquita *et al*., 2008) put terms and their ancestors into term set to denote gene firstly. Then, they compute semantic similarity score between the term sets using Tversky's ratio model (Tversky, 1977). Finally, the semantic similarity score between the term sets is regarded as gene functional similarity. Graph-based methods make use of GO sub-graph to describe gene, in which nodes are terms and arcs represent relationships between terms. These methods estimate gene functional similarity by means of graph matching (Alvarez and Yan, 2011; Cho *et al*., 2007; Gentleman *et al*., 2005; Lin *et al*., 2004; Sheehan *et al*., 2008; Ye *et al*., 2005; Yu *et al*., 2007). Vector-based methods represent each gene as a vector where each dimension corresponds to a term and 1 means the specific term occurs while 0 otherwise. They measure the gene functional similarity through calculating the cosine similarity of vector (Huang *et al*., 2007) or the probability of co-occurrence of the terms (Chabalier *et al*., 2007).

To our knowledge, the group-wise methods also have some shortcomings. The set-based and vector-based methods ignore some valuable information implicit in the semantics and relationships of terms. The graph-based methods are limited by the complexity of graph matching.

In general, some error-prone judgments about gene functional similarity may be raised by existing methods. In our views, it primarily results from the inappropriate computing of the information content (IC) of terms and unreasonable conversion from semantic similarity into functional similarity. For the effective comparison of gene function, we design a novel method based on Semantic Overlap Ratio of Annotations, namely SORA. Section 2 illustrates the details of our method, and the experimental results are shown and discussed in Section 3. Finally, Section 4 presents some concluding remarks.

2 METHODS

The process of measuring gene functional similarity by SORA is displayed in Figure 1. At first, to quantify the semantics of the terms, SORA infers the IC of the terms from their location in the GO hierarchy. Meanwhile, the inherited and extended IC values of the terms are computed separately. Next, for the semantics of a term set, SORA calculates the IC of the term set by combining the inherited and extended IC values of its members. Finally, the functional similarity between two genes is computed on the basis of the IC values of their term sets by a simple reciprocal average method.

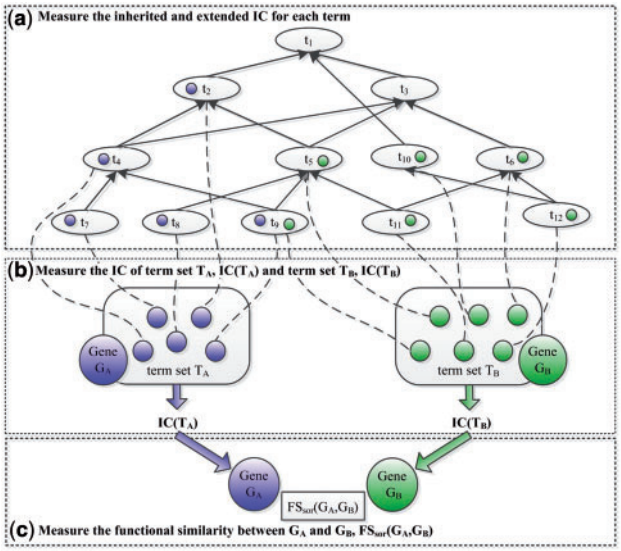

Fig. 1. Measuring gene functional similarity by SORA

2.1 Measure the inherited and extended IC of terms

2.1.1 Related works There are two approaches, corpus-based and structure-based, to compute the IC of a term. Under the corpus-based approach, the IC of the term t_i is defined as

$$IC_{corpus}(t_i) = -\log(p(t_i)) \quad (1)$$

In the Equation (1), $p(t_i)$ is the occurrence probability of t_i and its descendants in the specified GO annotation (GOA) corpus.

Considering a GOA corpus includes 50 distinct annotated genes, in which 15 genes are annotated with term t_i or t_i's descendants, the IC of the term t_i is

$$IC_{corpus}(t_i) = -\log\left(\frac{15}{50}\right) \approx 0.5229.$$

However, it becomes 0.3802 when annotation information about additional 10 genes annotated with the term t_i is added to the GOA corpus. It can be found that IC for the same term depends on the number of genes annotated with it. As argued by Guzzi *et al*. (2011), the semantics of GO terms should be independent of the annotation distribution. This approach suffers corpus bias and may not reflect the semantics of the term objectively.

Alternatively, the IC of the term can also be computed from the number of its descendants in the GO structure (Seco *et al*., 2004). We refer this approach as a structural IC approach. Under this approach, the IC of the term t_i is defined as

$$\begin{aligned} IC_{\text{structure}}(t_i) &= \frac{\log((\text{desc}(t_i)+1)/total\_terms)}{\log(1/total\_terms)} \\ &= 1 - \frac{\log(\text{desc}(t_i)+1)}{\log(total\_terms)} \end{aligned} \quad (2)$$

where $desc(t_i)$ means the number of descendants of term t_i, and $total\_terms$ is the number of terms in GO. This measure produces consistent IC of the term over different annotation corpus, which seems more reasonable than corpus-based approach. However, a new problem is that the IC

values of the terms without descendant are all 1 under this approach. Actually, the IC of these terms may be not entirely same. Hence, Equation (2) is also unreasonable for measuring the IC of the terms.

Besides, some measures (Gentleman *et al*., 2005; Ye *et al*., 2005) considered that the IC of the term is proportional to its depth in the hierarchy, which premised that the semantic of term is finer and finer details as one descends the hierarchy. However, these approaches may not distinguish the differences between the terms, which are at the same level but differ in the number of descendants. Meanwhile, we noticed that some works, which focused on the semantic distances of terms, achieved their goals through exploiting the information contained in the GO hierarchy. For example, to measure distance between linked terms, Jiang and Conrath (1998) weighted the edges along shortest path linking the terms based on the link density, term depth and the difference of their IC. Inspired by these works, we consider that the semantics of the term may be tightly related to its location in the GO hierarchy, which could be characterized by term depth (specificity) and the number of descendants (coverage). Accordingly, a novel approach is proposed to overcome the limitations suffered by aforementioned measures.

2.1.2 Inherited and extended IC of the term It assumed that the IC of the term is not only proportional to its depth but also inversely to the number of its descendants because more descendants the term has, less specific the semantics is. Therefore, the IC of the term is computed by Equation (3).

$$\begin{aligned} IC(t_i) &= Specificity(t_i) \times C\text{overage}(t_i) \\ &= Specificity(t_i) \times \left(1 - \frac{\log(\text{desc}(t_i) + 1)}{\log(\text{total\_terms})}\right) \end{aligned} \quad (3)$$

In Equation (3), the semantic specificity of term t_i, $Specificity(t_i)$ is computed by its depth in the GO hierarchy. The maximum depth of the term is taken as its depth. The semantic coverage of term t_i, $Coverage(t_i)$ is measured by the number of its descendants in GO, like Equation (2). Under this approach, the terms at lower levels are more specific with bigger IC, whereas the terms with more descendants are more generic with smaller IC.

According to the true path rule of GO, if a gene is annotated with a term, it is also annotated with the ancestors of the term. That is to say, the semantics of the ancestor term is generalized from that of its descendants, and the latter is extended from the former. In light of this, the semantics of the term is divided into two parts: one is inherited semantics, which is same as the semantics of its ancestors, and the other is extended semantics, which is special in itself. For measuring IC of a term set, the inherited IC and extended IC of each term, which represent the inherited and the extended semantics of the term respectively, are computed. Supposed that the term t_j is one ancestor of the term t_i, the inherited IC of the term t_i from the term t_j is actually equal to the IC of term t_j, $IC(t_j)$. The extended IC of the term t_i from the term t_j is defined as

$$IC_{\text{extended}}(t_j \rightarrow t_i) = IC(t_i) - IC(t_j). \quad (4)$$

Likewise, given the ancestor set of the term t_i, $AS(t_i)$, the inherited IC of the term t_i from $AS(t_i)$ equals the IC of $AS(t_i)$, $IC(AS(t_i))$. The extended IC of the term t_i from $AS(t_i)$, $IC_{extended}(AS(t_i) \rightarrow t_i)$, is

$$IC_{\text{extended}}(AS(t_i) \rightarrow t_i) = IC(t_i) - IC(AS(t_i)). \quad (5)$$

2.2 MEASURE THE IC OF TERM SET BY COMBINING THE INHERITED AND THE EXTENDED IC OF ITS MEMBERS

Regarding the IC of the term set, a simple method is summing up the IC of the terms in the set. Take an example, the IC of term set ts, which just contains two terms t_1 and t_2, is the summation of the $IC(t_1)$ and $IC(t_2)$. However, as discussed by Couto *et al*. (2005), the terms may share IC because of the inheritance nature of GO. Take the term set ts again, considering the term t_c is one common ancestor of t_1 and t_2, they share the inherited IC from t_c, $IC(t_c)$ but differ in the extended IC from t_c. Accordingly, the $IC(ts) = IC(t_1) + IC(t_2) = [IC(t_c) + IC_{\text{extended}}(t_c \rightarrow t_1)] + [IC(t_c) + IC_{\text{extended}}(t_c \rightarrow t_2)] = 2IC(t_c) + IC_{\text{extended}}(t_c \rightarrow t_1) + IC_{\text{extended}}(t_c \rightarrow t_2)$ in term of the Equation (4). Actually, IC(ts) should be $IC(t_c) + IC_{\text{extended}}(t_c \rightarrow t_1) + IC_{\text{extended}}(t_c \rightarrow t_2)$ because the IC shared by terms should not cumulatively contribute to the IC of the set. It is not hard to imagine that the IC of the set would be larger than reality since more shared IC exists. To overcome this limitation, it is necessary to remove the shared IC between the terms, which is summed repeatedly.

In fact, the calculation of the shared IC has been already proposed by GraSM (Couto *et al*., 2005) and DiShIn (Couto *et al*., 2011). These works focused on dealing with the shared IC when measuring semantic similarity between terms. GraSM defined the shared IC between terms as the average of their common disjunctive ancestors while DiShIn redefined it as the average of their all disjunctive ancestors. As verified, both of them could improve the performance of the semantic similarity measures. However, in our opinion, the shared IC between terms could be measured alternatively by the IC of their common ancestors set. Similarly, the shared IC between the term sets could be measured by the IC of their intersection.

Subsequently, we put forward an algorithm for computing the IC of the term set, as illustrated in Figure 2, which combines inherited and extended IC values of its members according to the structure of GO. To simplify the description of the algorithm, some notations are used in the algorithm: considering a term set X, $CET(X)$ consists of the terms without descendants in X; t_{extend} is used to extend term set X in each round, which is selected from $CET(X)$; ES_{extend} consists of the t_{extend} and its ancestors; $ES_i(X)$ is the extended term set X after the ith round extension and $IC_i(X)$ is the IC of $ES_i(X)$; OTS_i is the overlapped term set between ES_{extend} and $ES_i(X)$; $ES(X)$ is the final term set X after all extensions, and $IC(X)$ is IC of the term set X.

The process of measuring the IC of the term set is demonstrated by an example shown in Figure 3. Gene Q9BPW9 is annotated with manually assigned term set $X_g = \{$GO: 0004022, GO: 0004745, GO: 0047035, GO: 0016854$\}$ in molecular function sub-ontology. The initial $CET(X_g)$ is {GO: 0004022, GO: 0004745, GO: 0047035, GO: 0016854}. The process of computing the IC of the term set X_g includes several rounds and each round consists of four main steps:

(1) Select t_{extend} to extend $ES_i(X_g)$;
(2) Generate ES_{extend} and OTS_i;
(3) Calculate $IC_{extended}(OTS_i \rightarrow t_{extend})$ and $IC_i(X_g)$;
(4) Update $CET(X_g)$ and $ES_i(X_g)$.

As displayed in Figure 3, each term is represented by an oval with a GO identifier and IC value. In each round, the term t_{extend} is denoted by an oval with octagon. The terms of ES_{extend} are marked by the ovals with asterisks. The terms of $ES_i(X_g)$ are labeled with symbols like t_j, $j \in N$ in the circles. The overlapped

ALGORITHM: Measuring the IC of the term set based on the inherited and extended IC.
Input: Term set $X = \{t_1, t_2, \ldots, t_n\}$
Output: The IC of the term set X, $IC(X)$

```
Initialize CET(X)
For each t_i ∈ CET(X) do
   Calculate IC(t_i) using Eq.(3)
End for
IC_0(X)=0, IC(X)=0
ES_0(X)←Φ
n=|CET(X)|, i=1
While i ≤ n
    T_max←{t_j | t_j∈CET(X), ∀t_q∈CET(X), IC(t_j)≥IC(t_q)}
    t_extend=null
    If |T_max|=1
        t_extend←{t_j | t_j∈T_max}
    else
        {
            T_ancestors←{t_k | t_k∈T_max, ∀t_q∈T_max, |AS(t_k)|≥|AS(t_q)|}
            t_extend←t_k, where t_k is selected from T_ancestors randomly.
        }
    End if
    If t_extend =null then continue
    ES_extend←AS(t_extend) ∪{t_extend}
    OTS_i←ES_extend∩ES_i(X)
    If OTS_i ≠Φ then
        IC_i(X)←IC_i-1(X)+ IC_extended(OTS_i→t_extend)
    else
        IC_i(X)←IC_i-1(X)+ IC(textend)
    End if
    ES_i(X) ←ES_i-1(X) ∪ES_extend
    CET(X)←CET(X) −{t_extend}
    i++
End while
IC(X) = IC_n(X)
Return IC(X)
```

Fig. 2. Algorithm for measuring the IC of the term set

terms between $ES_1(X_g)$ and ES_{extend} are shown by the ovals with circles and asterisks.

In the first round, as shown in Figure 3a, GO: 0047035 is selected as t_{extend} to extend $ES_1(X_g)$. Because the initial $ES(X_g)$ is null, OTS_1 is null and $IC_1(X_g) = IC(t_{extend}) = 0.42857$ in term of Equation (4). According to the true path rule, X_g can also be annotated with the ancestors of the term t_{extend}. Therefore, the term t_{extend} and its all ancestors should be added into $ES(X_g)$. Then GO: 0047035 is removed from $CET(X_g)$. At the end of the round, $ES_1(X_g)$ and $CET(X_g)$ become $\{t_1, t_2, t_3, t_4, t_5, t_6, t_7\}$ and {GO: 0004745, GO: 0004022, GO: 0016854}, respectively.

In the second round, as illustrated by Figure 3b, GO: 0004745 is selected as t_{extend} to extend $ES_2(X_g)$. The overlapped terms between ES_{extend} and $ES_1(X_g)$ are t_1, t_2, t_3, t_4 and t_5. To measure $IC_{extended}(OTS_2 \rightarrow t_{extend})$, it is necessary to measure $IC(OTS_2)$. Because t_5 is the only member of the $CET(OTS_2)$, $IC(OTS_2) = IC(t_5)$, i.e. 0.10474. According to Equation (5), $IC_{extended}(OTS_2 \rightarrow t_{extend}) = IC_{extended}(t_5 \rightarrow t_{extend}) = IC(t_{extend}) - IC(t_5) = 0.35714 - 0.10474 = 0.25240$. For $IC_2(X_g) = IC_1(X_g) + IC_{extended}(OTS_2 \rightarrow t_{extend})$, $IC_2(X_g)$ becomes 0.68097. Then, the terms of ES_{extend} are added into $ES_1(X_g)$ and GO: 0004745 is removed from $CET(X_g)$. At the end of the second round, $ES_2(X_g)$ and $CET(X_g)$ are $\{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$ and {GO: 0004022, GO: 0016854}, respectively.

In the third round, as shown in Figure 3c, GO: 0004022 is selected as t_{extend} to extend $ES_3(X_g)$. The overlapped terms between ES_{extend} and $ES_2(X_g)$ are t_1, t_2, t_3, t_4 and t_5. The following process is similar to that of the second round. In the following process, $IC_{extended}(OTS_3 \rightarrow t_{extend})$ is calculated, i.e. 0.18875. Thus, $IC_3(X_g) = IC_2(X_g) + IC_{extended}(OTS_3 \rightarrow t_{extend}) = 0.68097 + 0.18875 = 0.86972$. At the end of the third round, $ES_3(X_g)$ and $CET(X_g)$ become $\{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9\}$ and {GO: 0016854}, respectively.

In the fourth round, as seen in Figure 3d, GO: 0016854 is selected as t_{extend} to extend $ES_4(X_g)$. The overlapped terms between ES_{extend} and $ES_3(X_g)$ are t_1 and t_2. For t_2 is one child of t_1, $IC(OTS_4) = IC(t_2)$, i.e. 0.00316. Thus, $IC_{extended}(OTS_4 \rightarrow t_{extend}) = 0.1152$ and $IC_4(X_g) = 0.98492$. Next, the terms of ES_{extend} are added into $ES_3(X_g)$, and GO: 0016854 is removed from $CET(X_g)$. Here, it is found that $CET(X_g)$ is null; thus, the iteration is finished.

After iteration is finished, the $IC_4(X_g)$ and $ES_4(X_g)$ are returned as $IC(X_g)$ and $ES(X_g)$, respectively. As shown in Figure 3e, the $IC(X_g)$ is 0.98492. The final $ES(X_g)$ is $\{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}, t_{11}, t_{12}\}$, which is consistent with the true path rule of GO.

Besides, we find that the key terms of which the IC could represent the shared IC between two term sets such as t_2 and t_5 in our strategy are coincidently the common disjunctive ancestors of the terms in the set like t_8, t_9, t_{10} and t_{12} in Figure 3. From this point, the IC of term set can also be given alternatively by summing the IC of the terms and remove the repeatedly summed IC of their common disjunctive ancestors.

2.3 MEASURE THE FUNCTIONAL SIMILARITY BETWEEN GENES

To compute gene functional similarity, set-based methods usually make use of Tversky's ratio model or its variants. Assuming that genes G_A and G_B are annotated with term sets $T_A = \{t_1, t_2, \ldots, t_m\}$ and $T_B = \{t_1, t_2, \ldots, t_n\}$, respectively, simUI (Gentleman *et al.*, 2005) defined the functional similarity between G_A and G_B as follows:

$$FS_{\text{simUI}}(G_A, G_B) = \frac{|T_A \cap T_B|}{|T_A \cup T_B|} \tag{6}$$

$|.|$ is the number of terms in the specified set. This method neglected the differences of the terms; simGIC (*Pesquita et al.*, 2008) improved simUI by the IC of the terms. In simGIC, the functional similarity between G_A and G_B is

$$FS_{\text{simGIC}}(G_A, G_B) = \frac{\sum_{t_i \in T_A \cap T_B} f(t_i)}{\sum_{t_j \in T_A \cup T_B} f(t_j)} \tag{7}$$

where $f(.)$ is the IC of the term. However, the shared IC of the terms was also summed repeatedly under this method. In fact, repeated summing of the shared IC is common in set-based methods. It may also result in misjudgments of gene functional similarity.

Inspired by Chen *et al.* (2012), the functional similarity between two genes is defined as the IC overlap ratio (ICOR) between their term sets as Equation (8).

$$FS_{sor}(G_A, G_B) = \left(\frac{IC(T_A \cap T_B)}{IC(T_A)} + \frac{IC(T_A \cap T_B)}{IC(T_B)}\right)/2 \tag{8}$$

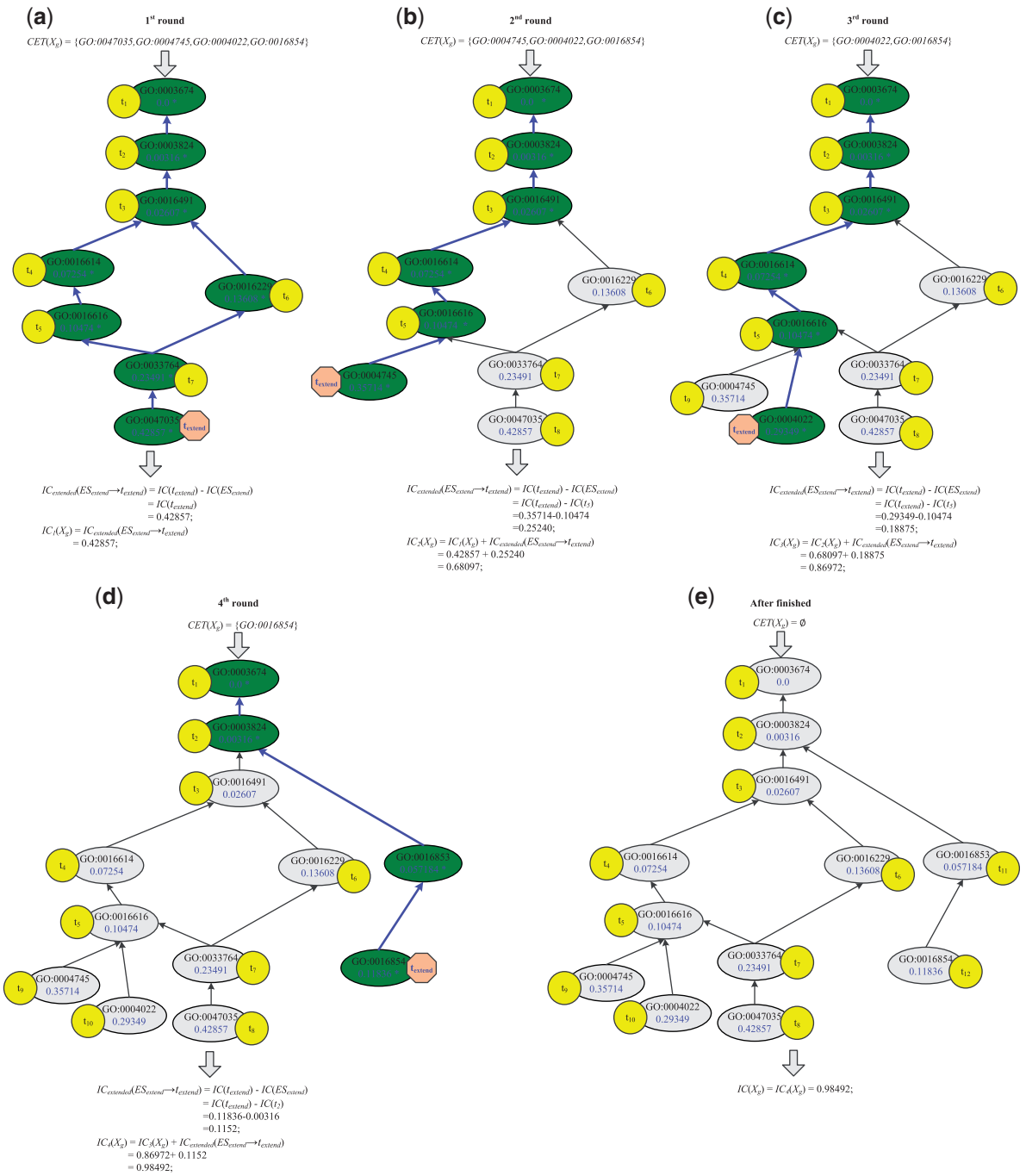

Fig. 3. The process of measuring the IC of the term set. Each term is represented by an oval node with GO identifier and the IC value. In each round, the term t_{extend} is denoted by an oval with the octagon. The terms of ES_{extend} are marked by ovals with asterisks. And the terms of $ES_i(X_g)$ are labeled with symbols like t_j, $j \in N$ in yellow circles. Overlapped terms between $ES_i(X_g)$ and ES_{extend} are shown by the ovals with circles and asterisks. The process includes four rounds corresponding to (**a–d**), respectively. The final $ES(X_g)$ and $IC(X_g)$ are shown by (**e**)

As known, the GOAs of genes are currently incomplete and suffer from a large research bias (Wang *et al.*, 2010, Yang *et al.*, 2012). To reduce the effects of annotation bias and imperfection, a simple reciprocal average method is used to make a balance between shallow and well annotated genes. In the Equation (8), the first item on the right of the equation reflects the proportion of the shared IC between T_A and T_B to the IC of T_A, and the second item reflects the proportion of the shared IC between T_A and T_B to the IC of T_B. The shared IC between the term sets is measured by the IC of the intersection between them $IC(T_A \cap T_B)$. To avoid repeated summing of shared IC, the IC of the term set T_A,T_B and $T_A \cap T_B$ are computed by the algorithm described in Figure 2.

3 VALIDATION AND RESULTS

To validate the performance of our method, SORA is implemented, and its web service can available at http://nclab.hit.edu.cn/SORA/.

SORA is compared on a widely used platform for Collaborative Evaluation of GO-based Semantic Similarity Measure (CESSM) (Pesquita *et al*., 2009b). The task is to measure functional similarity of 13 430 protein pairs, which involved 1039 proteins, in GO database and GOA released in August, 2008. According to the resources, terms in the GO are classified as Electronic-assigned terms (E-terms) and Manually assigned terms (M-Terms). E-terms are inferred from electronic annotations, whereas M-terms are inferred from experiments, computational analysis, author statements and curatorial statements. Considering GO aspects and the electronic annotations may influence performances of methods, validation experiments are conducted on six GOAs: AMF, ABP, ACC, MMF, MBP and MCC. The details of the six experimental GOAs are listed in Table 1.

As for the performance criteria, CESSM provides the Pearson correlations with sequence similarity (Seq), protein family similarity (Pfam), enzyme commission classification similarity (ECC) and Resolution (Res) to evaluate measures. Sequence similarity is computed by dividing the sum of their reciprocal BlAST bit scores by the sum of their self-BLAST bit scores. The Pfam similarity between two proteins is the ratio between the number of domains they share and the total number of those they have. ECC similarity is measured by the digits of the enzyme commission number shared by the proteins. The larger Pearson correlations with them suggest that the semantic similarities reflect the functional closeness of proteins better. Resolution is the relative intensity with which values in the sequence similarity scale are translated into the semantic similarity (Pesquita *et al*., 2008). A higher resolution indicates the method is more sensitive to the differences in annotations. It is noteworthy that, as reported by Pesquita *et al*. (2008), the relationship between semantic and sequence similarity is not linear, and the resolution was verified more appropriate to depict the intrinsic relationship between them than the correlation.

To evaluate the impact of the term IC, we measure the functional similarities of the protein pairs specified by CESSM using the methods based on the structural IC and that based on the term IC computed by our strategy (called SORA IC), respectively. These two approaches are evaluated on CESSM, and the results are displayed in Table 2. As suggested by the results, the method based on the SORA IC performs identically better than the other with respect to Seq, Pfam and ECC in the experiments. However, it is also found that the performance of the method based on SORA IC is not as good as the one based on structural IC on Res in some cases. It suggests that the differences of SORA IC may be not as obvious as those of structural IC, but the former reflect the reality better than the latter in terms of other metrics. On the whole, the SORA IC has more positive impacts on functional comparison of protein.

To validate the effects of the converting strategy, we convert the semantic similarity into function similarity using Jaccard and ICOR, respectively. The functional similarity scores measured with the two converting strategies are compared on CESSM.

Table 1. Descriptions of the six experimental GOA

GOA	Components	Number of terms
AMF	M-terms and E-terms of MF sub-ontology	9375
ABP	M-terms and E-terms of BP sub-ontology	9235
ACC	M-terms and E-terms of CC sub-ontology	5163
MMF	Only M-terms of MF sub-ontology	4437
MBP	Only M-terms of BP sub-ontology	6291
MCC	Only M-terms of CC sub-ontology	3343

Table 2. The impacts of the term IC

GOA	Strategy	Seq	Res	Pfam	ECC
AMF	SORA IC	**0.5949**	**0.9762**	**0.5765**	**0.6726**
	Structural IC	0.5528	0.9720	0.5247	0.6056
ABP	SORA IC	**0.7293**	0.9076	**0.4679**	**0.4648**
	Structural IC	0.6374	**0.9229**	0.4297	0.4618
ACC	SORA IC	**0.6549**	0.9371	**0.4960**	**0.3741**
	Structural IC	0.6472	**0.9447**	0.4790	0.3603
MMF	SORA IC	**0.6443**	**0.9605**	**0.5703**	**0.6502**
	Structural IC	0.5539	0.9520	0.4686	0.5859
MBP	SORA IC	**0.6754**	0.8966	**0.4171**	**0.4311**
	Structural IC	0.5810	**0.9079**	0.3688	0.4172
MCC	SORA IC	**0.6875**	0.9110	**0.4725**	**0.3512**
	Structural IC	0.6406	**0.9221**	0.4613	0.3429

The best results are in bold.

As listed in the Table 3, the method with ICOR gets higher Res and ECC, whereas it is comparable with the other one on Pfam in most experiments. On all of the experimental datasets, the scores computed by ICOR show lower correlation with sequence similarities. This may illustrate that the distribution of the scores converted by Jaccard matches better with that of sequence similarities than by ours. According to Res, the scores derived by Jaccard are less capable to capture the differences in the annotations of the proteins than by our strategy. Overall, the results indicate that ICOR is more discriminating for gene functional comparison.

To evaluate effectiveness of our method, SORA is performed on the six experimental GOAs separately. The functional similarities of the 13 430 protein pairs computed by SORA are compared with other methods on CESSM after every experiment. The CESSM enables the comparison of new methods against 11 pairwise and group-wise functional similarity methods. SORA is compared against typical methods of them including simUI, simGIC as well as Resnik's (RB), Lin's (LB) and Jiang and Conrath's (JB) based on BMA, respectively. Table 4 shows the Seq, Res, Pfam, ECC, average and the improvement on respective average level of them computed by different methods. The negative values, signed with '↓' in Table 4, imply that the method is under average level with respect to the specific metric.

As for Seq, simGIC shows consistently better performance than others on the six experimental datasets, whereas SORA is

slightly superior to the average level. Regarding Res and ECC, SORA outperforms to others in most cases. When performed on AMF, SORA is the best with improvements in the average level against Res and ECC, by 25.47 and 8%, respectively. When applied to MMF, SORA has significant improvements in the average level against Res and ECC by 13.85 and 16.98%, respectively. Referring to average levels of the Res and ECC, SORA improves them by 14.04 and 10.86% when conducted on ABP and improves by 11.81 and 5.14% when performed on the MBP. SORA running on the terms of CC sub-ontology is the best. Regarding Pfam, SORA is comparable with the best and has significant improvements in the average level of Pfam. Moreover, SORA outperforms average level of these methods in terms of almost all of the metrics in the experiments. From these results, SORA is outstanding than others while measuring gene functional similarity.

To evaluate SORA against each metric, the average improvements of them in six experiments are calculated and shown in Table 5. Regarding Seq, simGIC is the best by 12%, and SORA has a positive effect on it, whereas some others have a negative impact on it. As for Res and ECC, SORA shows the best performances with 15.37 and 8.82% improvement on average level, respectively. In terms of Pfam, SORA gets a significant improvement and performs comparably with the best, simGIC. It reveals that SORA has improved the performances of gene functional comparison.

Furthermore, to provide an intuitive measure of relative performance, we summarize the comparison results by ranking performances of the concerned methods in the six experiments.

Table 3. The effects of the converting strategies

GOA	Strategy	Seq	Res	Pfam	ECC
AMF	ICOR	0.5949	**0.9762**	0.5765	**0.6726**
	Jaccard	**0.6629**	0.9625	**0.6122**	0.6378
ABP	ICOR	0.7293	**0.9076**	**0.4679**	**0.4648**
	Jaccard	**0.7778**	0.8555	**0.4679**	0.4104
ACC	ICOR	0.6549	**0.9371**	**0.4960**	**0.3741**
	Jaccard	**0.7621**	0.8920	0.4865	0.3587
MMF	ICOR	0.6443	**0.9605**	0.5703	**0.6502**
	Jaccard	**0.6988**	0.9368	**0.5895**	0.5943
MBP	ICOR	0.6754	**0.8966**	0.4171	**0.4311**
	Jaccard	**0.7374**	0.8561	**0.4339**	0.3916
MCC	ICOR	0.6875	**0.9110**	**0.4725**	**0.3512**
	Jaccard	**0.7391**	0.8680	0.4633	0.3406

The best results are in bold.

Table 4. The performances of different methods in six experiments

GOA	Metric	Original value						Average value	Improvement in average level (%)					
		simGIC	simUI	RB	LB	JB	SORA		simGIC	simUI	RB	LB	JB	SORA
AMF	Seq	**0.7172**	0.5925	0.6683	0.6063	0.5459	0.5949	0.6209	15.52	4.57	7.64	−2.34↓	−12.07↓	−4.18↓
	Res	0.9559	0.9671	0.9577	0.5705	0.2409	**0.9762**	0.7781	22.85	24.30	23.09	−26.67↓	−69.04↓	25.47
	Pfam	**0.6380**	0.6181	0.5718	0.5639	0.4908	0.5765	0.5765	10.67	7.21	−0.82↓	−2.19↓	−14.86↓	0
	ECC	0.6219	0.6365	0.6027	0.6417	0.5612	**0.6726**	0.6228	−0.14↓	2.21	−3.23↓	3.04	−9.88↓	8
ABP	Seq	**0.7732**	0.7304	0.7397	0.6369	0.5864	0.7293	0.6993	10.56	4.44	5.77	−8.93↓	−16.15↓	4.29
	Res	0.8373	0.8628	0.9004	**0.9326**	0.3345	0.9076	0.7959	5.2	8.41	13.13	17.18	−57.97↓	14.04
	Pfam	0.4547	0.4505	0.4587	0.3727	0.3318	**0.4679**	0.4227	7.55	6.57	8.52	−11.84↓	−21.5↓	10.69
	ECC	0.3981	0.4022	0.4444	0.4352	0.3707	**0.4648**	0.4192	−5.05↓	−4.05↓	6	3.81	−11.57↓	10.86
ACC	Seq	**0.7500**	0.6721	0.7113	0.6398	0.5014	0.6549	0.6549	14.52	2.62	8.61	−2.31↓	−23.44↓	0
	Res	0.9001	0.9337	0.9167	0.9359	0.3098	**0.9371**	0.8222	9.47	13.56	11.5	13.82	−62.31↓	13.97
	Pfam	0.4974	**0.5214**	0.4930	0.4850	0.3123	0.4960	0.4675	6.39	11.52	5.46	3.74	−33.2↓	6.09
	ECC	0.3612	0.3757	**0.3776**	0.3683	0.2598	0.3741	0.3528	2.39	6.49	7.03	4.39	−26.35↓	6.04
MMF	Seq	**0.6665**	0.5907	0.6512	0.5976	0.5219	0.6443	0.6120	8.90	−3.49↓	6.40	−2.36↓	−14.73↓	5.27
	Res	0.9358	0.9304	0.9335	0.9376	0.3641	**0.9605**	0.8437	10.92	10.28	10.65	11.13	−56.84↓	13.85
	Pfam	**0.5824**	0.5504	0.5221	0.5148	0.4503	0.5703	0.5317	9.54	3.51	−1.81↓	−3.18↓	−15.32↓	7.26
	ECC	0.5874	0.5782	0.4841	0.5161	0.5189	**0.6502**	0.5558	5.68	4.02	−12.9↓	−7.14↓	−6.64↓	16.98
MBP	Seq	**0.7359**	0.6949	0.7267	0.6269	0.5333	0.6754	0.6655	10.58	4.42	9.19	−5.80↓	−19.87↓	1.49
	Res	0.8697	0.8831	0.8929	**0.9117**	0.3573	0.8966	0.8019	8.46	10.12	11.35	13.7	−55.44↓	11.81
	Pfam	0.4383	0.4253	**0.4506**	0.3810	0.2740	0.4171	0.3977	10.19	6.93	13.3	−4.20↓	−31.09↓	4.88
	ECC	0.3887	0.3818	0.4257	0.4216	0.4113	**0.4311**	0.4100	−5.21↓	−6.9↓	3.83	2.83	0.31	5.14
MCC	Seq	**0.7348**	0.6499	0.7214	0.6441	0.5013	0.6875	0.6565	11.93	−1.01↓	9.89	−1.89↓	−23.64↓	4.72
	Res	0.8691	0.9072	0.8921	0.9102	0.3441	**0.9110**	0.8056	7.88	12.61	10.73	12.98	−57.28↓	13.08
	Pfam	0.4681	**0.4872**	0.4676	0.4562	0.3321	0.4725	0.4473	4.66	8.93	4.54	1.99	−25.76↓	5.64
	ECC	0.3502	**0.3527**	0.3443	0.3390	0.2519	0.3512	0.3316	5.63	6.37	3.86	2.25	−24.01↓	5.90

Original values show Seq, Res, Pfam and ECC provided by CESSM. Average values present the average level on each metric. Improvements in the average level (%) display the improvement on average level with respect to each metric. Symbol '↓' denotes that the method is under average level in term of the specific metric. The best levels of each metric are in bold.

Table 5. Performances of different methods in term of the metrics

Metric	simGIC	simUI	RB	LB	JB	SORA
Seq	**12**	0.4	7.92	−3.94↓	−18.32↓	1.93
Res	10.8	13.21	13.41	7.02	−59.8↓	**15.37**
Pfam	**8.17**	7.44	4.86	−2.61↓	−23.62↓	5.76
ECC	0.55	1.36	0.76	1.53	−13.02↓	**8.82**

The best results are in bold.

To simplify, we define the ranking of a given method m_i with respect to an assigned performance metric p_j in a specific experiment E as $rank(m_i,p_j,E)$. As these methods are compared in the same task, the comprehensive ranking of m_i, $RS(m_i)$, is measured by

$$RS(m_i) = \sum rank(m_i, p_j, E) \tag{9}$$

Sorting $RS(m_i)$ in increasing order gives the final ranking of the concerned methods. The rankings of different methods are listed in Table 6. It suggests that SORA is at the top of the list by smallest comprehensive ranking of 54. SORA is still the best among these methods. The second is simGIC and RB is the third.

Generally, SORA is able to obtain better results and perform better than other methods. The structure of GO has a great contribution to its success, as it implies expressive information about gene function. Further analysis indicates that the group-wise methods show better overall performances than pairwise methods. It may be related to the ways of converting semantic similarity into gene functional similarity. The pairwise methods combine semantic similarity of terms into gene functional similarity with the help of BMA. The group-wise methods take semantic similarity between the term sets as gene functional similarity in a single step. The way of converting in the latter may be closer to reality than that in the former.

4 CONCLUSION

In this article, we put forward a novel method, namely SORA, to measure gene functional similarity. It was evaluated against typical pairwise and group-wise methods on CESSM. From the experimental results, SORA is a more effective and reliable way to estimate gene functional similarity than other tested methods. The success of SORA may be related to the following characteristics.

First, SORA makes use of semantic specificity and coverage to measure the IC of the term. The term IC is determined by its location in the GO hierarchy rather than the number of proteins annotated with it. Thus, it can overcome the limitation of GOA corpus bias, which affects the corpus-based approach heavily. With the help of both semantic specificity and coverage, our strategy could reflect the differences in semantics of terms more objectively than the structural IC.

Second, SORA computes the IC of annotating term set by combining the inherited and extended IC of the terms based on the structure of GO. It can effectively avoid repeated summing of the shared IC of terms, which is the key point for estimating the IC of the term set correctly.

Table 6. The rankings of the concerned methods

Rank	$RS(m_i)$	Method
1	54	SORA
2	69	simGIC
3	72	RB
4	76	simUI
5	92	LB
6	140	JB

Third, SORA uses simple reciprocal ICOR between the term sets as gene functional similarity. It is an appropriate description of functional relationship between genes. As discussed before, SORA measures semantic similarity in a single step, regardless of the number of annotations per protein, which is essential for combining similarities of term pairs in pairwise approach. This strategy has positive impacts on gene function comparison.

Moreover, from the results of our experiments, all of the methods performed better with E-terms than without. We consider that sometimes the E-terms may provide new knowledge about protein function, which has not been confirmed by manual means. High quality computational inferring of annotations would promote the gene function comparison, which is one of our interests in the future.

Funding: This work was supported by the Natural Science Foundation of China (60932008, 61172098, 61271346); Specialized Research Fund for the Doctoral Program of Higher Education of China (20112302110040).

Conflict of Interest: none declared.

REFERENCES

Alvarez,M.A. and Yan,C. (2011) A graph-based semantic similarity measure for the gene ontology. *J. Bioinform. Comput. Biol.*, **9**, 681–695.

Azuaje,F. *et al.* (2005) Ontology-driven similarity approaches to supporting gene functional assessment. In: *Proceeding of the ISMB 2005 SIG Meeting on Bio-ontologies*. MI, USA, pp. 9–10.

Batet,M. *et al.* (2011) An ontology-based measure to compute semantic similarity in biomedicine. *J. Biomed. Inform.*, **44**, 118–125.

Brameier,M. and Wiuf,C. (2007) Co-clustering and visualization of gene expression data and gene ontology terms for Saccharomyces cerevisiae using self-organizing maps. *J. Biomed. Inform.*, **40**, 160–173.

Chabalier,J. *et al.* (2007) A transversal approach to predict gene product networks from ontology-based similarity. *BMC Bioinformatics*, **8**, 235.

Chen,J. *et al.* (2009) Disease candidate gene identification and prioritization using protein interaction networks. *BMC Bioinformatics*, **10**, 73.

Chen,X. *et al.* (2012) A sensitive method for computing GO-based functional similarities among genes with 'shallow annotation'. *Gene*, **509**, 131–135.

Chen,Y. and Xu,D. (2004) Genome-scale protein function prediction in yeast Saccha-romyces cerevisiae through integrating multiple sources of high throughput data. *Nucleic Acids Res.*, **32**, 6414–6424.

Cho,Y.R. *et al.* (2007) Semantic integration to identify overlapping functional modules in protein interaction networks. *BMC Bioinformatics*, **8**, 265.

Cho,Y.R. *et al.* (2009) Semantic similarity based feature extraction from microarray expression data. *Int. J. Data Min. Bioinform.*, **3**, 333–345.

Couto,F.M. *et al.* (2005) Semantic similarity over the gene ontology: Family correlation and selecting disjunctive ancestors. In: *Proceedings of the 14th ACM International Conference on Information and knowledge Management*. Bermen, Germany, pp. 343–344.

Couto,F.M. *et al.* (2011) Disjunctive shared information between ontology concepts: application to Gene Ontology. *J. Biomed. Semantics*, **2**, 5–21.

Gentleman,R. *et al.* (2005) Visualizing distances. In: *Bioinformatics and computational biology solutions using R and Bioconductor*. Springer, New York, pp. 170–173.

GO-Consortium. (2004) The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res.*, **32**, D258–D261.

Guzzi,P.H. *et al.* (2011) Semantic similarity analysis of protein data: assessment with biological features and issues. *Brief. Bioinform.*, **5**, 569–585.

Huang,D.W. *et al.* (2007) David gene functional classification tool: A novel biological module centric algorithm to functionally analyze large gene list. *Genome Biol.*, **8**, R183.

Jain,S. and Bader,G.D. (2010) An improved method for scoring protein-protein interactions using semantic similarity within the gene ontology. *BMC Bioinformatics*, **11**, 562–575.

Jensen,L.J. *et al.* (2003) Prediction of human protein function according to Gene Ontology categories. *Bioinformatics*, **19**, 635–642.

Jiang,J.J. and Conrath,D.W. (1998) Semantic similarity based on corpus statistics and lexical taxonomy. In: *Proceedings of International Conference on Research in Computational Linguistics*. Taiwan, China, pp. 19–33.

Lee,H.K. *et al.* (2004) Coexpression analysis of human genes across many microarray data sets. *Genome Res.*, **14**, 1085–1094.

Li,D. *et al.* (2008) PRINCESS, a protein interaction confidence evaluation system with multiple data sources. *Mol. Cell. Proteomics*, **7**, 1043–1052.

Lin,D. (1998) An information-theoretic definition of similarity. In: *Proceeding of the 15th International Conference on Machine learning*. Morgan Kaufmann, San Fran-cisco, CA, pp. 296–304.

Lin,N. *et al.* (2004) Information assessment on predicting protein-protein interactions. *BMC Bioinformatics*, **5**, 154.

Lord,P. *et al.* (2003) Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation. *Bioinformatics*, **19**, 1275–1283.

Nariai,N. *et al.* (2007) Probabilistic protein function prediction from heterogeneous genome-wide data. *PLoS One*, **2**, e337.

Mathur,S. and Dinakarpandian,D. (2011) Finding disease similarity based on implicit semantic similarity. *J. Biomed. Inform.*, **45**, 363–371.

Martin,D. *et al.* (2004) GOToolBox: functional analysis of gene datasets based on Gene Ontology. *Genome Biol.*, **5**, R101.

Mistry,M. and Pavlidis,P. (2008) Gene ontology term overlap as a measure of gene functional similarity. *BMC Bioinformatics*, **9**, 327.

Ortutay,C. and Vihinen,M. (2009) Identification of candidate disease genes by integrating Gene Ontologies and protein interaction networks: case study of primary immune deficiencies. *Nucleic Acids Res.*, **37**, 622–628.

Pesquita,C. *et al.* (2008) Metrics for go based protein semantic similarity: a systematic evaluation. *BMC Bioinformatics*, **9** (**Suppl. 5**), S4.

Pesquita,C. *et al.* (2009a) Semantic similarity in biomedical ontologies. *PLoS Comput. Biol.*, **5**, e1000443.

Pesquita,C. *et al.* (2009b) CESSM: collaborative evaluation of semantic similarity measures. In: *Proceeding of JB 2009: Challenges in Bioinformatics*. Lisbon, Portugal.

Pozo,A.D. *et al.* (2008) Defining functional distances over gene ontology. *BMC Bioinformatics*, **9**, 50.

Qu,Y. and Xu,S. (2004) Supervised cluster analysis for microarray data based on multivariate Gaussian mixture. *Bioinformatics*, **20**, 1905–1913.

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.

Riensche,R.M. *et al.* (2007) XOA: Web-enabled cross-ontological analytics. In: *Proceeding of IEEE Congress on Services*. Salt Lake City, UT, pp. 99–105.

Schlicker,A. *et al.* (2007) Functional evaluation of domain-domain interactions and human protein interaction networks. *Bioinformatics*, **23**, 859–865.

Schlicker,A. *et al.* (2010) Improving disease gene prioritization using the semantic similarity of Gene Ontology terms. *Bioinformatics*, **26**, i561–i567.

Seco,N. *et al.* (2004) An intrinsic information content metric for semantic similarity in WordNet. In: *Proceedings of 16th European Conference on Artificial Intelligence*. Valencia, Spain, pp. 1089–1090.

Sevilla,J.L. *et al.* (2005) Correlation between gene expression and go semantic similarity. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **2**, 330–338.

Sheehan,B. *et al.* (2008) A relation based measure of semantic similarity for gene ontology annotations. *BMC Bioinformatics*, **9**, 468.

Tversky,A. (1977) Features of similarity. *Psychol. Rev.*, **84**, 327–351.

Wang,J.Z. *et al.* (2007) A new method to measure the semantic similarity of go terms. *Bioinformatics*, **23**, 1274–1281.

Wang,J. *et al.* (2010) Revealing and avoiding bias in semantic similarity scores for protein pairs. *BMC Bioinformatics*, **11**, 290.

Yang,D. *et al.* (2008) Gaining confidence in biological interpretation of the microarray data: the functional consistence of the significant GO categories. *Bioinformatics*, **24**, 26–271.

Yang,H. *et al.* (2012) Improving GO semantic similarity measures by exploring the ontology beneath the terms and modelling uncertainty. *Bioinformatics*, **28**, 1383–1389.

Ye,P. *et al.* (2005) Gene function prediction from congruent synthetic lethal interactions in yeast. *Mol. Syst. Biol.*, **1**, 0026.

Yilmaz,S. *et al.* (2009) Gene-disease relationship discovery based on model-driven data integration and database view definition. *Bioinformatics*, **25**, 230–236.

Yu,H. *et al.* (2007) Total ancestry measure: quantifying the similarity in tree-like classification, with genomic applications. *Bioinformatics*, **23**, 2163–2173.