A novel insight into Gene Ontology semantic similarity

Yungang Xu a,b, Maozu Guo a,⁎, Wenli Shi a, Xiaoyan Liu a, Chunyu Wang a

a School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, PR China
b School of Life Science and Technology, Harbin Institute of Technology, Harbin 150001, PR China

A R T I C L E   I N F O

Article history:
Received 23 January 2013
Accepted 19 April 2013
Available online 26 April 2013

Keywords:
Gene Ontology
Semantic similarity
Directed acyclic graph

A B S T R A C T

Existing methods for computing the semantic similarity between Gene Ontology (GO) terms are often based on external datasets and, therefore, are not intrinsic to GO. Furthermore, they not only fail to handle identical annotations but also show a strong bias toward well-annotated proteins when being used for measuring similarity of proteins. Inspired by the concept of cellular differentiation and dedifferentiation in developmental biology, we propose a shortest semantic differentiation distance (SSDD) based on the concept of semantic topotropy to measure the semantic similarity of GO terms and further compare the functional similarity of proteins. Using human ratings and a benchmark dataset, SSDD was found to improve upon existing methods for computing the semantic similarity of GO terms. An in-depth analysis shows that SSDD is able to distinguish identical annotations and does not depend on annotation richness, thus producing more unbiased and reliable results. Online services can be accessed at the Gene Functional Similarity Analysis Tools website (GFSAT: http://nclab.hit.edu.cn/GFSAT).

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

Gene Ontology (GO) maintains a dynamic, structured, precisely defined and controlled vocabulary of terms to represent the roles and cellular localizations of proteins in a species-independent manner. It comprises three orthogonal ontologies: cellular component (CC), molecular function (MF) and biological process (BP). These ontologies are structured as three directed acyclic graphs (DAGs) in which the nodes correspond to the terms describing a certain biological semantic category and the edges represent the linkages between terms describing defined relationships [1]. The most common relationships are ‘is a’ and ‘part of’.

GO terms have been widely used to annotate proteins, such as those in the Gene Ontology Annotation (GOA) project [2]. This procedure enables bioinformaticians to benefit from using semantic similarity to compare proteins based on function instead of sequence similarity [3]. Ever since the first application of semantic similarity in biology by Lord et al. [4,5], a variety of measures have been developed for quantifying the semantic similarity of GO terms. These measures have been widely used in protein–protein interaction prediction [6,7], network prediction [8], cellular localization prediction [9], automatic annotation validation [10], semantic search integration of [11], pathway modeling [12], and improving analysis of microarray data quality [13]. The state-of-the-art methods for computing the semantic similarity of GO terms are traditionally divided into four categories: edge-based, node-based, hybrids of edge- and node-based, and graph-based [14]. A brief review of each category will be presented below; a detailed review is beyond the scope of this paper and has already been presented by Catia Pesquita et al. [15].

The edge-based methods are a natural and direct way of evaluating semantic similarity in taxonomy, originally used in lexical taxonomy by Rada et al. [16]. Edge-based methods simply count the number of edges on the shortest path traversed from one node to another. Similar methods were subsequently proposed and have been used in GO [17–22]. The shortcoming of these methods is that they implicitly assume that all edges represent uniform distances and that all nodes in the taxonomy are evenly distributed and have similar densities. Unfortunately, this assumption does not hold in GO [4]. Although several researchers [15,16,18] have attempted to address this issue by assigning different weights to edges at different levels and by taking into account network density, they still ignored two facts: GO terms at the same level did not necessarily share the same specificity, and spuriously high similarities appeared due to the shallow annotation problem, as shown by Richardson and Smytonik [18].

The most prevalent node-based approaches are Resnik’s [23], Lin’s [24] and Jiang and Conrath’s methods [25]. They were originally developed for WordNet [26] and then applied to GO [4,5]. They all use information content (IC) to represent the specificity of GO terms. The IC value for a term t is defined as

$$IC(t) = - \log p(t).$$
where $p(t)$ is the probability of the occurrence of the term $t$ in a certain corpus (e.g., the GOA database). Later, Couto et al. [27], Schlicker et al. [28], Bodenreider et al. [29], and Rensche et al. [30] improved upon these three methods. Nevertheless, they still shared a drawback in that IC was calculated based on the occurrence of GO terms in an external corpus, which produced a constantly changing semantic similarity between the same term pair with updates of the external data (known as not intrinsic to GO). In addition, they had a preference for well-annotated proteins, and this preference led to biased results, which will be discussed later. There are other node-based methods, with or without the use of IC, based on groupwise approaches [31–38]. In these methods, the semantic similarity of proteins is found by representing proteins as sets or vectors of GO terms and calculating the overlap between sets or the inner product of vectors rather than their combined pairwise similarity. All of these groupwise methods lose the information contained in the hierarchical structure of GO and ignore the relationships between GO terms.

Alvarez et al. [14] proposed a purely graph-based approach. A protein can be described as a shortest-path graph, a sub-graph of GO, which consists of all of the terms annotating the protein and their ancestors in the ontology. The functional similarity of proteins can then be calculated using a shortest-path graph kernel, a graph matching technique proposed by Borgwardt et al. [39]. This method avoids the shortcomings of IC-based methods and explicitly explores the structure of GO. Nevertheless, graph matching is computationally expensive, being an NP-complete problem on general graphs.

The first hybrid method was Jiang & Conrath’s (J&C) method [25], which is known as a node-based method because its simplified version was widely used in GO (see formula (4)). The original J&C’s method considered the local network density, node depth, link type and strength and was applied to GO by Othman et al. [40], who first defined the overall edge weight $(wt)$ for a term $c$ and its parent term $p$ as

$$wt(c, p) = \left( \beta + (1-\beta) \frac{E(p)}{E(p)} \right) \left( \frac{d(p) + 1}{d(p)} \right)^\alpha |IC(c) - IC(p)| T(c, p),$$

(2)

where $d(p)$ denotes the depth of the term $p$ in the hierarchy, $E(p)$ the number of the edges in the child links (i.e., the local density), $E$ the average density in the whole hierarchy, $T(c, p)$ the link type factor, and $|IC(c) - IC(p)|$ the link strength. The parameters $\alpha (\geq 0)$ and $\beta (0 \leq \beta \leq 1)$ control the degrees of term depth and density contributing to the edge weight computation. The overall distance between two terms $t_1$ and $t_2$ would thus be the summation of edge weights along the shortest path linking the two terms, as shown in formula (3),

$$Dist(t_1, t_2) = \sum_{c \in \{path(t_1, t_2)\}} wt(c, parent(c)).$$

(3)

where $\{path(t_1, t_2)\}$ is the set that contains all of the terms in the shortest path from $t_1$ to $t_2$, and $\text{LCA}(t_1, t_2)$ denotes the lowest common ancestors of $t_1$ and $t_2$. In the special case where only link strength is considered in the weighting scheme of formula (2), i.e., $\alpha = 0$, $\beta = 1$, the distance simplification can be simplified to formula (4), i.e., the aforementioned simplified version of J&C’s formula.

$$Dist(t_1, t_2) = |IC(t_1) - IC(t_2)| - 2 \times IC(\text{LCA}(t_1, t_2)).$$

(4)

Another hybrid method, namely, the shortest path (SP) proposed by Shen et al. [41], uses the sum of the weights on the shortest path to compute the semantic distance for two GO terms $t_a$ and $t_b$, which is described as

$$\text{dist}_{sp}(t_a, t_b) = \frac{\arctan \left( \frac{\sum_{t \in \text{pathb}} IC(t_1) - \sum_{t \in \text{patha}} IC(t_2)}{\pi/2} \right)}{\sum_{t \in \text{pathb} \cap \text{patha}} IC(t_2)},$$

(5)

where $\text{path}_a(\text{path}_b)$ is the shortest path that connects the term $t_a(t_b)$ with the most informative common ancestor (MICA). $t_1$ and $t_2$ are the terms located on $\text{path}_a$ and $\text{path}_b$. The semantic similarity of the two terms is defined as $1 - \text{dist}_{sp}(t_a, t_b)$.

All of the above hybrid methods suffer from the aforementioned issue of dependence on external resources because the terms they use are represented as their ICs. Wang et al. [42] developed an IC-independent method, in which each edge is assigned a weight, named the semantic contribution factor $(\omega_e)$, according to the type of relationship. They represent a GO term $A$ as $\text{DAG}_A = (A, T_A, E_A)$, a sub-graph of GO, where $T_A$ is the set of all ancestors of $A$ and itself, and $E_A$ is the set of corresponding links. The contribution of any term $t$ to the semantics of a term $A$ is defined as the $S$-value of the term $t$ related to term $A$, $SA(t)$, which can be calculated by

$$\begin{cases}
SA(t) = 1 & \text{if } t \neq A \\
SA(t) = \max \{\omega_e, SA(t')|t' \in \text{children of } t\} & \text{if } t = A
\end{cases}$$

where $\omega_e$ is the semantic contribution factor for edge $e$ linking term $t$ with its child term $t'$. The semantic value of term $A$, $SA(A)$, is the aggregate semantic contribution of all terms in the $\text{DAG}_A$, i.e.,

$$SV(A) = \sum_{t \in T_A} SA(t).$$

(7)

Given $\text{DAG}_A = (A, T_A, E_A)$ and $\text{DAG}_B = (B, T_B, E_B)$ for the two terms $A$ and $B$ respectively, the semantic similarity between them, $S_{GO}(A, B)$, is defined as

$$S_{GO}(A, B) = \frac{\sum_{t \in T_B} \left( SA(t) + SB(t) \right)}{SV(A) + SV(B)}.$$

(8)

There are still two disadvantages to using Wang’s method, even though it avoids the drawbacks of IC-based methods and takes the linking types of GO terms into account. First, the semantic contribution factor $(\omega_e)$ is suggested to be 0.8 and 0.6 for ‘is-a’ and ‘part-of’ relationships, respectively, an approach that is highly arbitrary. Although $\omega_e$ can be set within a range from 0 to 1 according to different situations, it is difficult for most users to determine the most appropriate value. Second, GO is structured as directed acyclic graphs, wherein a term can have more than one parent and/or child term. This property indicates that there may be multiple paths from a given term $t$ to its ancestor $t_a$. The authors defined the semantic contribution of $t_a$ to $t$ as the maximum product of all of the paths linking them. They ignored the contributions of other paths, which causes the semantic value of a term to be invariant even though the deletion, addition, or alteration of terms can occur in those ignored paths. Thus, Wang’s method was not sensitive to GO updates.

It is insignificant to compare only the semantic similarity of GO terms. All methods for calculating the similarity of GO terms are proposed to finally compare protein function based on GOA, rather than that of GO terms per se. We therefore need a method to integrate pairwise semantic similarities into a single functional similarity because a protein may be annotated by more than one term. Three distinct approaches have been proposed for this integration: Lord et al. [45] used an arithmetic average (abbreviated as Avg) of pairwise similarities between all terms of the first protein and the second one; Sevilla et al. [43] used only the maximum (Max) similarity between all term pairs; and Couto et al. [3], Schlicker et al. [28] and Azua et al. [44] developed the best-match average (BMA) method, in which each term of the first protein is paired only with the most similar term of the second one and vice versa. We take the BMA approach to compare protein similarity, as it was found to be the most effective [34].

When any of the existing semantic similarity measurements is used to compare protein similarity by integrating the approaches mentioned above, a new problem arises: the similarity of any pair...
of proteins annotated by the same GO terms (known as identical annotation) will always be 1. This does not match the human perception that the similarity between genes annotated with more specific terms should be greater than those annotated with more general ones (see the ‘Discussion’ section for details). This case is also known as shallow annotation because the similarity of two proteins annotated by top terms in GO could be found to be spuriously high.

For the above reasons, we proposed a novel measure for semantic similarity between GO terms and further use it to compare the functional similarity of proteins, aiming at resolving at least four issues: 1) the dependence on external resources in IC-based methods, 2) the loss of the structural information in GO in edge- or node-based methods, 3) the identical annotation problem for all existing methods, and 4) the preference for well-annotated proteins. We completed two major evaluations of our method using two independent biological settings backed by in-depth analysis and discussion.

2. Methods

Inspired by developmental biology, we propose a new IC-independent hybrid method for computing the semantic similarity of GO terms from a novel perspective. In developmental biology, cellular differentiation is the process by which a less specialized cell becomes a more specialized cell type. The ability of a cell to differentiate into all possible cell types of an organism is known as totipotency. Dedifferentiation is a cellular process whereby a partially or terminally differentiated cell reverts to an earlier developmental stage. In GO, each term has one or more parents and children, excluding the three top (root) terms (e.g., GO:0008150, GO:0005575 and GO:0003674) and the bottom (terminal) terms, and each edge transmits the semantics between terms primarily through two types of relationships. The process of semantic transmission from a term to its ancestors resembles the process of cellular differentiation in that the three top terms are like three zygotes and the semantic specialization of the terms increases downward through the GO terms. Each specialized term in the ontology inherits and integrates partial semantics from its parental terms. Therefore, we introduce the concept of semantic differentiation, the transition of a term from one pattern of semantic integration to another, and the capacity of this differentiation is termed semantic totipotency.

2.1. The shortest semantic differentiation distance

Based on semantic differentiation and semantic totipotency, we proposed a method for measuring the semantic similarity of GO terms. Firstly, a sub-graph was extracted from GO for a given term, e.g., GO:0043229, shown in Fig. 1, which consists of a set of terms including all of its ancestors. The DAG was then viewed as a semantic genealogy wherein a term inherits the semantics of its ancestors and distributes them to its descendants. Incorporating all paths linking a term and its ancestors, the semantic totipotency of a given term t can be quantified as a T-value \( T(t) \) as follows:

\[
T(t) = \begin{cases} 
\frac{1}{t_{\text{parent of } t}} 
& \text{if } t = r \\
\text{mean} \left( \omega \cdot T(t') \right) 
& \text{if } t \neq r
\end{cases}
\]  

(9)

where \( r \) represents one of the root terms (e.g., GO:0008150, GO:0005575 or GO:0003674). The semantic totipotency values of the three root terms are assigned to be 1 because they are the most general terms and have the differentiation capacity to convert to any descendant terms in their own ontology, similar to the zygote in cellular differentiation. The variable \( \omega \) is the semantic differentiation factor for the edge linking term t with its parent \( t_p \). The T-value of any other terms can be derived as the average of all of its parents’ T-values multiplied by the semantic differentiation factor \( \omega \), as shown in formula (9). The differentiation capacity \( T(t) \) should decrease moving down the hierarchy and be positively proportional to the number of descendants, or local density. Thus, the \( \omega \) between a term t and its parent \( t_p \) should be greater than 0 and less than 1 and can be calculated as

\[
\omega = \frac{\text{Dist}(t)}{\text{Dist}(t_p)}
\]  

(10)

where \( \text{Dist}(t) \) is the number of descendants of the term t, including itself. It should be noted that the number of descendants is calculated using the DAG of the entire Gene Ontology rather than the sub-graphs mentioned above. In this way, each GO term will be mapped to a numerical property, the T-value. It is invariable, except in cases of the deletion of obsolete terms or the addition of new terms accompanying the update of the GO database. Thus it is sensitive to updates.

Based on T-values, we proposed the shortest semantic differentiation distance (SSDD) to measure the semantic similarity in the Gene Ontology. It is intuitive that if two terms \( t_A \) and \( t_B \) diverge at a higher level, i.e., their lowest common ancestors (LCAs) are nearer to the root, the difference between them should be larger, whereas if they diverge at a lower level, the difference should be smaller. This difference can be quantified by the difficulty of \( t_A \) transforming to \( t_B \), a process whereby \( t_A \) first dedifferentiates into their LCAs and then their LCAs differentiate into \( t_B \). Based on this principle, SSDD first finds the shortest paths from one term to another via their LCAs, then calculates the sum of the T-values for all of the terms on each path and defines the minimum sum as the semantic distance for the two terms. SSDD implicitly indicates that if the LCA is near the root, the sum on the shortest path will increase, and otherwise the opposite will occur. Therefore, the sum of the T-values on the shortest path is consistent with the expected distance and can be exploited for its estimation.

Given two terms \( t_A \) and \( t_B \), the normalized distance between them is defined as

\[
\text{Dist}(t_A, t_B) = \frac{\arctan \left( \min \left\{ \sum_{t \in \text{path}(t_A, t_B)} T(t) \right\} \right)}{\pi/2}.
\]  

(11)

Fig. 1. The overlapped DAG for GO: 0043231 and GO: 0044444. A sub-graph generated from GO of the two seed terms (gray background). The dotted arrows represent the ‘part–of’ relationship and the solid arrows show the ‘is–a’ relationship.
where \( \text{path}(t_A, t_B) \) represents a set of terms on the shortest path connecting the terms \( t_A \) and \( t_B \) via their LCAs. The arctan function is used to normalize the distance to \((0, 1)\). Apparently, \( \text{Dist}(t_A, t_B) \) is symmetric because \( \text{Dist}(t_A, t_B) = \text{Dist}(t_B, t_A) \). After normalization, the semantic similarity can be defined as:

\[
\text{Sim}_{SSDD}(t_A, t_B) = 1 - \text{Dist}(t_A, t_B),
\]

(12)

2.2. Example

We take two terms as an example, computing their semantic similarity using SSDD. It should be noted that the DAG for this example was obtained from the GO database in November 2011. Due to the daily evolution of the GO database, a GO term’s DAG may change with the deletion of obsolete terms and the addition of new terms.

Given two terms GO: 0043231 (\( t_{11} \)) and GO: 0044444 (\( t_{10} \)), an overlapping DAG can be generated from GO (see Fig. 1). The T-value of each term in this DAG was calculated by formula (9), and the results are listed in Table 1.

The shortest path connecting \( t_{10} \) and \( t_{11} \) is easily found to be \( t_{10} \rightarrow t_6 \rightarrow t_9 \rightarrow t_{11} \), with \( t_6 \) as their unique LCA. Using formulas (11) and (12), the semantic similarity of terms ‘GO: 0043231’ and ‘GO: 0044444’ is calculated in two steps as

\[
\text{Dist}(t_{10}, t_{11}) = \frac{\text{arctan}(0.3258 + 0.6396 + 0.4795 + 0.3597)}{\pi/2} = 0.6779,
\]

\[
\text{Sim}_{SSDD}(t_{10}, t_{11}) = 1 - 0.6779 = 0.3221.
\]

2.3. Evaluations

How well a measure captures the similarity in function between two proteins is not a trivial assessment because there is no direct way to ascertain the true functional similarity between them [15]. However, the performance of existing semantic similarity measurements has been verified in terms of correlations with sequence similarity [4,28,33,34], gene expression profiling [43,45], protein–protein interactions [6], protein family similarity [27,28] and human ratings [38], or against subnuclear location [9], gene clustering of pathways [42], and human regulatory pathways [12]. First, we evaluate the performance of SSDD by comparing the calculated semantic similarities with human ratings as described in [38]; we then compare SSDD with other existing methods using the Collaborative Evaluation of GO-based Semantic Similarity Measures (CESSM), an online tool for the automated evaluation of GO-based semantic similarity measures. CESSM enables the comparison of new measures against previously published ones in terms of performance against sequence, Pfam, and EC similarities [46].

2.3.1. Comparison of SSDD with human ratings

Li et al. [38] selected 25 pairs of GO terms that cover term pairs with high, intermediate and low similarities and invited ten biologists to grade the selected term pairs from 0 (no similarity) to 10 (synonymy) individually. The mean of every term pair’s grades from the ten biologists was calculated as its value of artificial semantic similarity as shown in Additional file 1. Here, we directly quote the artificial similarity values and the semantic similarities of the four measures from the reference [38]. We then calculate each term pair’s semantic similarity by Wagner’s measure, simUI and SSDD using the GO database of September 2005, as used in [38]. The results are shown in Additional file 1. Finally, we calculate the Pearson’s correlation coefficients (PCCs) between the artificial grades and the similarity values obtained by each of these seven methods to compare their performance. A higher PCC represents better performance.

2.3.2. Comparison of SSDD with state-of-the-art methods

We used the CESSM to compare our methods with other existing methods. CESSM is an online tool made available by the XLDB research team at the University of Lisbon. It provides a standard dataset consisting of 13,430 pairs of proteins involving 1039 distinct proteins and implements 11 state-of-the-art semantic similarity measures: simGIC [47], simUI [48], and the average [3], maximum [43] and best-match average [2] combinations of the term similarities found by Resnik’s [23], Lin’s [24] and J&C’s methods [25]. As noted by Pesquita et al. [34], the maximum and average approaches have limitations from a biological point of view, and the best-match average performs better than the other two. Thus, we only consider the best-match average (BMA) versions of Resnik’s (RB), Lin’s (LB) and J&C’s (JB) methods, coupled with simGIC (GI) and simUI (UI). Finally, SSDD was compared with five other methods (i.e., GI, UI, RB, LB, and JB).

CESSM provides three standards of evaluation: EC similarity (ECC), Pfam similarity (Pfam) and sequence similarity (SeqSim). The performance of a given method is evaluated by measuring the Pearson’s correlation coefficient between it and other state-of-the-art methods by the aforementioned standards under different ontologies in GO. As noted by Pesquita et al. [34], the relationship between semantic similarity and sequence similarity is not linear. Thus, they recommended using a measure called resolution instead of the correlation coefficient to evaluate how well the semantic similarity matches the sequence similarity. Resolution is the relative intensity whereby variations in the sequence similarity scale are translated into the semantic similarity scale. The detailed measurement was described in [34]. Higher resolution values mean that the semantic similarity method has a higher capability to distinguish between different levels of protein functions. Therefore, a method with a higher resolution performs better than one with a lower resolution.

3. Results

3.1. The performance of SSDD against human ratings

The performance of SSDD against human ratings (expert evaluation) is measured by comparing the Pearson’s correlation coefficients (PCCs) of SSDD to those of the other six methods. PCCs were calculated between the semantic similarity given by each method and the human ratings. A higher PCC means that the method has a higher capability to achieve semantic similarities closer to human performance. As shown in Table 2, the SSDD method achieves the best results. In particular, it outperforms two intrinsic methods, Wang’s method and simUI, confirming that the semantic similarities obtained by SSDD most closely match human perception.

3.2. Evaluation by CESSM

The functional similarities of 13,430 protein pairs were computed by SSDD coupled with the best-match average (BMA) approach, using the annotations of all code evidence (i.e., including IEA). After uploading the results according to the requirements of CESSM (see

Table 1

<table>
<thead>
<tr>
<th>GO terms</th>
<th>T-value</th>
<th>GO terms</th>
<th>T-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_1 )</td>
<td>1.0000</td>
<td>( t_2 )</td>
<td>0.3665</td>
</tr>
<tr>
<td>( t_2 )</td>
<td>0.8614</td>
<td>( t_3 )</td>
<td>0.4171</td>
</tr>
<tr>
<td>( t_3 )</td>
<td>0.4877</td>
<td>( t_6 )</td>
<td>0.4795</td>
</tr>
<tr>
<td>( t_4 )</td>
<td>0.8011</td>
<td>( t_{10} )</td>
<td>0.3258</td>
</tr>
<tr>
<td>( t_5 )</td>
<td>0.6400</td>
<td>( t_{11} )</td>
<td>0.3597</td>
</tr>
<tr>
<td>( t_6 )</td>
<td>0.6396</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Additional file 2), we obtained the original results shown in Table 3 using PCcs and Table 4 using resolutions.

As shown in Table 3 and Fig. 2, the SSDD method performs better than the other five methods in MF, BP, and CC when they are compared to ECC, Pfam, and SeqSim, respectively. Table 4 shows the resolutions for different methods when sequence similarity is compared with the semantic similarity measured by them. It also shows that SSDD performs comparably to the other five methods. These results confirm that SSDD can serve as an alternative method to evaluate the functional similarity between proteins. Although it does not show very significant improvements compared to other methods, it can compare proteins with identical annotations and reduce the effects of the uneven number of annotations (known as annotation richness), providing a more authentic and unbiased result (see the ‘Discussion’ section for details).

4. Discussion

We have proposed a new measure for GO semantic similarity based on concepts from developmental biology. The well-structured Gene Ontology that contains terms of different semantic specificities was viewed as an organism system that contains different cell types. Semantic transfer from a parent to its daughters and the reverse process were defined as semantic differentiation and semantic dedifferentiation, respectively. These processes are similar to cellular differentiation and cellular dedifferentiation. Semantic totipotency was proposed to measure the degree of semantic transmission from a term to all its daughters, which is similar to measuring the ability of a cell to differentiate into other cell types using cellular totipotency. In view of this characteristic, we calculated the shortest semantic differential distance between two terms to measure their semantic similarity, a novel insight into GO semantic similarity. The two major evaluations mentioned above were not able to fully reveal the advantages of the present method. Next, we will present an in-depth discussion and analysis of its main advantages.

4.1. Exploring the structural information of GO

A good semantic similarity measurement should take into account the irregular nature of the Gene Ontology (also known as structural information), which contains variable edge lengths (edges at the same level convey different semantic distances), variable depths (terms at the same level have different specificities), and variable node densities (some areas of the ontology have a greater density of terms than others). In general, edge-based methods rely on the calculation of the length of the edges between two terms and ignore these irregularities. Othman et al. [40] fully acknowledged these irregularities and presented their algorithm as formula (2). However, the formula is complex, and it is difficult to handle the two parameters $\alpha$ and $\beta$. For this reason, only the simplified version was widely used in GO. Wang et al. [42] explored the different links between terms using the contribution factor ($\omega$). Unfortunately, they did not compute this parameter using GO or other data. They suggested setting it arbitrarily within a range of 0 to 1 according to different situations. We also use a parameter in SSDD called the semantic differentiation factor ($\omega$). In contrast, its assignment method is given by the ratio of the number of descendants of two given terms (formula (11)). $\omega$ reflects the strength of the semantic transfer between parental and daughter terms, which can be understood as edge distance, and the descendant number, which can be understood as the node density (local density). Therefore, although SSDD is a simple measurement, it fully incorporates the irregularities in GO.

4.2. Comparing identical annotations

Identical annotation occurs when two proteins are annotated with the same terms. It is an issue shared by existing methods as mentioned in the “Introduction” section. For example, in the simplest case in Fig. 1, a given protein pair $g_1$ and $g_2$ both annotated with the single term GO: 0005623 and another protein pair $g_3$ and $g_4$ both annotated with the single term GO: 0043229 will both have similarities of 1 using any of the existing measurements. This condition is not true in reality. We would expect there to be more similarity in the latter case than in the former case because the specificity of terms increases downward through the DAG. SSDD can distinguish proteins with identical annotations from each other because the semantic similarity between any term and itself is not constantly 1, but varies with its location in the ontology. Computed by SSDD, the similarity between $g_1$ and $g_2$ is 0.5474 and the similarity between $g_3$ and $g_4$ is 0.7154.

4.3. Annotation richness

The reason that SSDD performs only slightly better than other methods if evaluated by CESSM can be found by analyzing the standard dataset provided by CESSM. A statistical analysis of the
The annotations of the 1039 distinct proteins in the 13,430 protein pairs provided by CESSM clearly show that all of these proteins are heavily researched and annotated. The average numbers of annotation entries of each protein (known as annotation richness) are all greater than that of the entire UniprotKB-GOA dataset in BP, MF, CC, and All (all three ontologies) (see Fig. 3). Furthermore, the proteins are primarily distributed in well-studied species and richly annotated, especially in the first three (Human, Baker’s yeast and mouse), in which up to 60% of the proteins are distributed (see Fig. 4). As described before, all of the methods provided by CESSM are IC-based except for simUI, and IC was calculated by the occurrence of one term in a special corpus such as GOA. This situation will most likely lead to biased results in a highly active research domain because more research produces more annotations. This uneven distribution of the test sample facilitates the high performance of IC-based methods.

Fig. 2. The performance of SSDD compared with other methods evaluated by CESSM. The bar graph is drawn from the data in Table 3 and shows that SSDD outperforms the other five methods using the PCC of each method to ECC, Pfam and SeqSim in BP, MF, and CC, respectively. SSDD did not demonstrate a significant superiority to other methods when evaluated by CESSM because the proteins in CESSM are all well annotated, which facilitates the high performance of IC-based methods.

Fig. 3. The average annotation richness for each protein in CESSM, UniprotKB and our own samples. The 1039 proteins in CESSM are all well annotated; the average number of annotation entries of these proteins is all significantly greater than that of the entire UniprotKB-GOA datasets in BP, MF, CC, and All (all three ontologies) (denoted by “All”). This is precisely the reason why SSDD does not perform significantly better than other methods. We also re-sampled three sets of protein pairs. “Rich”, “Medium” and “Rare” represent these three sets of 100 protein pairs with average annotations per protein significantly higher than, approximately equal to and significantly less than that of UniprotKB-GOA, respectively.

Fig. 4. The distribution of the 1039 proteins in 63 distinct species. The 1039 proteins are primarily distributed in the first seven species (Human, Baker’s yeast, Mouse, Fission yeast, Fruit fly, Rat and Bovine) and include more than 85% of the total proteins. All of these species, especially the first three (approximately 60%), are all extensively studied model organisms whose proteins are richly annotated. This annotation bias favors the IC-based methods.
the average number of annotations per protein is approximately the same as that in UniprotKB-GOA; and "Rare" comprised 100 protein pairs involving 80 distinct proteins, for which the average number of annotations per protein are significantly less than that in UniprotKB-GOA (see Fig. 3). The similarities of these three sets of protein pairs were calculated by SSDD, GI, UI, LB, RB, and JB in BP, MF, and CC. The performances of different methods were evaluated by the PCCs between the semantic similarities and the sequence similarities. The sequence similarities were computed by the log reciprocal BLAST score (LRBS), as described in [34]. As shown in Fig. 5, SSDD does not depend on annotation richness because its PCCs do not significantly decrease from "Rich" to "Medium" to "Rare". Other methods show sharp decreases with the decline of annotation richness, especially in the CC ontology. These findings demonstrate that SSDD, unlike other IC-based or IC-independent methods, displays no bias for well-annotated proteins, thus providing a more accurate and reliable result.

As demonstrated above, SSDD has demonstrated at least four improvements over other methods: 1) it does not depend on external resources (i.e., it is intrinsic to GO), 2) it exploits the structural information contained in GO that can generate a more authentic similarity with GO’s updates, 3) it settles the matter of identical annotation shared by all existing methods, and 4) it reduces the effects of annotation richness, giving a more accurate and reliable result.

4.4. Prospects

In the light of development, no measure is perfect. SSDD still include certain limitations. For example, SSDD uses edge information, but does not distinguish between the edges of different semantic relationships, e.g., “part-of” and “is-a.” Thus, one direction for future development would be to assign different weights to different edges. Another limitation is that only the “best” lowest common ancestor is included, whereas the other disjunctive ancestors are overlooked. As with GraSM, a method proposed by F.M. Couto et al. [3,27], all of the disjunctive common ancestors should be selected and used in future development.

5. Conclusions

Inspired by cellular differentiation and dedifferentiation in developmental biology, we propose a method (SSDD) that calculates the functional similarity between proteins based on GO annotations from a novel perspective. In comprehensive evaluations using human ratings and a benchmark dataset (CESSM), SSDD compares favorably with other methods. In addition to its “intrinsic” property, SSDD is able to distinguish identical annotations, and its performance does not depend on annotation richness, thus producing more unbiased and reliable results. Online services based on SSDD can be accessed at the Gene Functional Similarity Analysis Tools website (GFSAT: http://nclab.hit.edu.cn/GFSAT).

Supplementary data to this article can be found online at http://dx.doi.org/10.1016/j.jygene.2013.04.010.

Acknowledgments

The work was supported by the Natural Science Foundation of China under Grant No. 60932008, 61172098 and 61271346, and the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant No. 20112302110040.

References
