Characterisation of semantic similarity on gene ontology based on a shortest path approach

Ying Shen

School of Software Engineering,
Tongji University,
Shanghai, P.R. China
E-mail: csyingshen@gmail.com

Shaohong Zhang

Department of Computer Science,
Guangzhou University,
Guangzhou, P.R. China
E-mail: zimzsh@gmail.com

Hau-San Wong*

Department of Computer Science,
City University of Hong Kong, Hong Kong
E-mail: cshswong@cityu.edu.hk
*Corresponding author

Lin Zhang

School of Software Engineering, Tongji University,
Shanghai, P.R. China
E-mail: cslinzhang@tongji.edu.cn

Abstract: Semantic similarity defined on Gene Ontology (GO) aims to provide the functional relationship between different GO terms. In this paper, a novel method, namely the Shortest Path (SP) algorithm, for measuring the semantic similarity on GO terms is proposed based on both GO structure information and the term's property. The proposed algorithm searches for the shortest path that connects two terms and uses the sum of weights on the path to estimate the semantic similarity between GO terms. A method for evaluating the nonlinear correlation between two variables is also introduced for validation. Extensive experiments conducted on the PPI dataset and two public gene expression datasets demonstrate the overall superiority of SP method over the other state-of-the-art methods evaluated.

Keywords: GO; gene ontology; shortest path; semantic similarity.

Reference to this paper should be made as follows: Shen, Y., Zhang, S., Wong, H.S. and Zhang, L. (2014) 'Characterisation of semantic similarity on gene ontology based on a shortest path approach', *Int. J. Data Mining and Bioinformatics*, Vol. 10, No. 1, pp.33–48.

Biographical notes: Ying Shen is an Assistant Professor in School of Software Engineering, Tongji University, Shanghai, PR China. She received her PhD degree from City University of Hong Kong in 2012. Her research interests include biological data mining, knowledge-based clustering, RNA motif recognition and RNA folding prediction.

Shaohong Zhang is an Associate Professor in the Department of Computer Science at Guangzhou University. He was a Postdoctoral Fellow in the Department of Computer Science, City University of Hong Kong. He received the PhD degree from Department of Computer Science, City University of Hong Kong. His research interests include pattern recognition, data mining and bioinformatics.

Hau-San Wong is an Associate Professor in the Department of Computer Science at City University of Hong Kong. Prior to joining the City University of Hong Kong, he was a research associate in the School of Electrical and Information Engineering, the University of Sydney and a post-doctoral teaching fellow in the Department of Computer Science, Hong Kong Baptist University. His research interests include bioinformatics and machine learning.

Lin Zhang received the BS and MS degrees from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, PR China, in 2003 and 2006, respectively. He received the PhD degree from the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, in 2011. From March 2011 to August 2011, he was a Research Assistant with the Department of Computing, the Hong Kong Polytechnic University. He is now an Assistant Professor in School of Software Engineering, Tongji University, Shanghai, PR China. His research interests include biometrics, pattern recognition, computer vision and perceptual image/video quality assessment, etc.

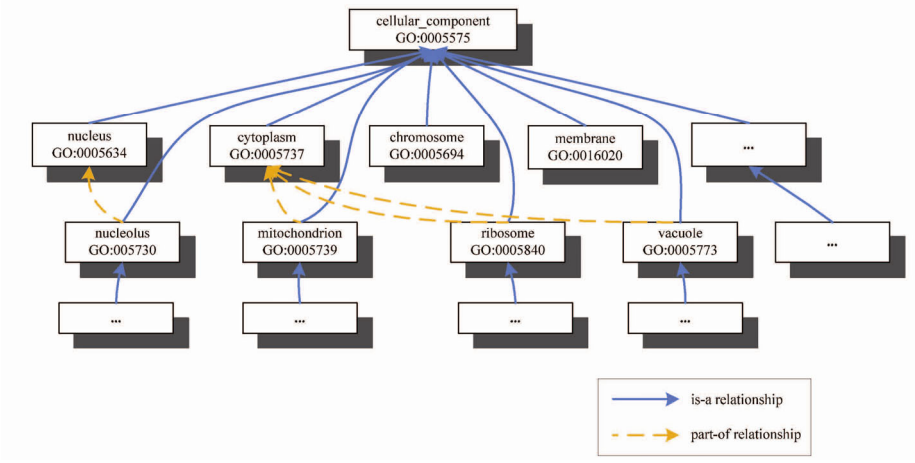
1 Introduction

Genome sequencing projects are now producing a large amount of biological data. For the sake of exploring and making use of these data, an efficient system to describe their biological properties becomes important. Toward this end, many functional descriptor systems have been established in the past years. For example, the MIPS Functional Catalogue (FunCat) (Ruepp et al., 2004) is a biological function annotation database aims at describing the functions of proteins of prokaryotic and eukaryotic origins. Currently, FunCat consists of 28 primary function classes, e.g., metabolism, transcription, protein synthesis, and 1362 more specific function categories. Another important annotation system is Gene Ontology (GO), which is more widely used in describing functional propertities of genes and proteins. GO not only focuses on describing the protein/gene functions, but also attempts to characterise other properties like the cellular localisation of the proteins, as well as the relationships between these properties. The complexity of protein properties leads to the intricacy of the GO structure. In the next two subsections, we shall introduce more details about GO and the functional similarity defined on GO.

1.1 Gene Ontology

Gene Ontology (The Gene Ontology Consortium, 2000) is a structured and controlled vocabulary, which characterises the functional properties of gene/proteins using standardised terms. GO is composed of three independent ontologies: Biological Process (BP), Molecular Function (MF), and Cellular Component (CC). Unlike the disjoint classes used in FunCat, GO adopts a more complex structure to describe protein properties and the relationships between them. The first concept in GO is the GO term. A GO term is a word used to describe a certain functional property. Every GO term has a corresponding GO ID in the form of 'GO:*******'. For example, the GO ID for the term 'cellular_component' is GO:0005575. The second concept is the relationships defined between GO terms. There are two kinds of relationships: 'is-a' relationship and 'part-of' relationship. If two terms *A* and *B* have a 'is-a' relationship, it means term *A* is an instance of term *B*. For example, nucleus is a cellular component. Therefore, the terms 'nucleus' and 'cellular_component' are connected by 'is-a' relationship. If *A* and *B* have a 'part-of' relationship, it means that *A* is a component of *B*. For example, nucleolus is a part of nucleus. They are connected by a 'part-of' relationship. One term may be linked with two or more terms. Terms, together with the relationships, are represented using Directed Acyclic Graphs (DAG). Figure 1 presents a subgraph extracted from the CC ontology. It can be seen that the terms are located at different layers according to their specificities. 'Cellular_component' is the root term containing the most general information. Other terms, like 'nucleus' and 'cytoplasm', are located at the lower layers because they have more specific meanings compared with the root term. Relationships between terms are represented by arrows. Terms at the end of the arrows are called 'ancestors', while terms at the start of the arrows are called 'descendents'. Ancestors and their descendents are connected by different relationships. The complexity of GO lies in the intricacy of the relationships of terms. From Figure 1, it can be seen that descendents may have more than one ancestor and *vice versa*.

Figure 1 A subgraph extracted from the Cellular Component ontology. Arrows with solid lines represent the 'is-a' relationships, and arrows with dash lines represent the 'part-of' relationships (see online version for colours)



GO structure provides abundant information which can facilitate the existing clustering and classification algorithms (Dotan-Cohen et al., 2009; Pandey et al., 2009; Wang et al., 2005; Denaxas and Tjortjis, 2008; Chen and Tang, 2010) in bioinformatics.

1.2 Semantic similarity on GO

Semantic similarity is an important type of information derived from GO, the concept of which is originally used in the field of linguistics. In linguistics, two words are considered to be similar if they have similar meanings. When GO was first constructed, its biological terms and graphical structures allow the comparisons between two GO terms based on their semantic contents. For example, in Figure 1, 'mitochondrion' is more similar to 'ribosome' than to 'nucleolus', because mitochondrion and ribosome are both in cytoplasm while nucleolus is in the nucleus.

To estimate the semantic similarity between two GO terms in the early years, the previously defined methods in linguistics were used directly to measure the semantic similarity over the terms in GO (e.g., Resnik's method (Resnik, 1999) and Lin's method (Lin, 1998)). In 2003, Lord et al. (2003) discovered that the semantic similarity calculated from annotations correlates well with the sequence similarity. After that, many new approaches have been proposed specifically for measuring the semantic similarity on GO (Schlicker et al., 2006; Wang et al., 2007). Although new methods are proposed from time to time, they all have their own advantages and limitations and there is still a large scope for improvement.

Semantic similarity can be defined for both the GO terms and gene products. The state-of-the-art methods for specifying semantic similarity over the GO terms can be divided into three groups: edge-based, node-based, and a hybrid of the above two. For the edge-based approaches, they mainly consider the lengths of the paths connecting the terms (Cheng et al., 2004; Pekar and Staab, 2002) as the distance between the terms. For the node-based methods, they rely on the properties of the terms derived from information theory (Jiang and Conrath, 1997; Lin, 1998; Resnik, 1999; Schlicker et al., 2006). There are also hybrid methods that consider both the substructure of GO and the properties of terms involved (Wang et al., 2007).

Based on the semantic similarity over terms, the semantic similarity for gene products can be defined as the maximum ('Max') (Wu et al., 2005), or average ('Ave') (Wang et al., 2005) value of the semantic similarity between their annotations. In addition to the 'Max' and 'Ave' methods, there are some more complicated methods proposed in (Schlicker et al., 2006; Wang et al., 2007). The details of these methods will be described in Section 2.

In this work, we propose a new method for measuring the semantic similarity over GO terms. The new method is proposed based on the observation that, if two terms diverge at the higher levels of GO, the discrepancy between the functions represented by the terms should be larger, and vice versa. The new method aims to find a path connecting the terms and uses a metric defined on the path to characterise the semantic similarity of two terms. If the path is long, which means that the two terms diverge at a higher level, the terms are different. On the contrary, if the path is short, the terms are similar to each other.

To evaluate the accuracy of the existing methods for measuring semantic similarity, manually curated information and experimental data (e.g., Protein-Protein Interaction (PPI) (Xu et al., 2008), pathway information (Wang et al., 2007), and gene expression

data (Xu et al., 2008)) have been used. In the previous work, researchers often use the Pearson correlation score between the similarity based on gene expression data and the computed semantic similarity to assess the accuracy of the proposed methods. However, in this paper, we introduce another validation approach to replace the previously used Pearson correlation coefficient.

The rest of the paper is organised as follows. Section 2 introduces a number of representative methods for semantic similarity computation over terms and genes. Section 3 presents the algorithm proposed by us for measuring the semantic similarity over terms. Section 4 reports the experimental results obtained from five different methods. Finally, Section 5 concludes the paper with a summary.

2 Measuring semantic similarity on GO

Existing methods for semantic similarity computation over terms generally fall into three categories: edge-based, node-based, and a hybrid of the former two.

Edge-based methods are intuitive, among which (Pekar and Staab, 2002) and (Cheng et al., 2004) are two representative ones. Suppose t_1 and t_2 are two terms, and t is their lowest common ancestor. The *distance* method in (Pekar and Staab, 2002) counts the number of edges connecting the root with t, and the number of edges connecting t with t_1 and t_2. The distance between t_1 and t_2 is calculated using equation (1) below and can be easily converted to a similarity value:

$$\text{dist}(t_1, t_2) = \frac{\text{len}(\text{root}, t)}{\text{len}(\text{root}, t) + \text{len}(t, t_1) + \text{len}(t, t_2)} \tag{1}$$

where $len(x, y)$ is the length of the path connecting the nodes x and y, represented by the number of edges on the path. The *distance* method assumes that the weight of each edge is always 1. Another edge-based algorithm (Cheng et al., 2004) uses the average length of all paths that go through the longest partial path shared by two nodes. The edges are weighted using the depth information. The disadvantage of the edge-based methods is that the weights of the edges at the same level are assumed to be the same. However, the terms at the same level of GO do not necessarily correspond to the same specificity, and accordingly the edges connecting two terms do not necessarily have the same weights.

Node-based methods focus mainly on the specificity of the terms, which is expressed using the concept of *Information Content* (IC). The IC value for a term t is defined as

$$IC(t) = -\log p(t) \tag{2}$$

where $p(t)$ is the probability of occurrence of the term t in a certain corpus (e.g., SGD database). All the node-based methods are defined based on the IC values of the GO terms involved.

Resnik's method (Resnik, 1999) is one of the first methods using IC values to measure the semantic similarity for GO terms. In this method, the semantic similarity for terms t_1 and t_2 is defined as

$$\text{sim}_{\text{Resnik}}(t_1, t_2) = \max_{t \in \text{ancester}(t_1, t_2)} IC(t) \tag{3}$$

where t is the common ancestor of t_1 and t_2. Term t with the largest IC value is also called the *Most Informative Common Ancestor* (MICA). Because Resnik's method only

considers the specificities of the common ancestors when measuring the semantic similarity of two GO terms, Lin's (Lin, 1998) and Jiang's (Jiang and Conrath, 1997) methods made some improvements by adding the IC values of the terms t_1 and t_2. Their methods involve the specificities of t_1 and t_2, since with an increase of the two terms' specificities, the terms will become less similar. The model used in Lin's and Jiang's methods is represented as Eqs. (4) and (5) below.

$$\mathrm{sim}_{\mathrm{Lin}}(t_1, t_2) = \frac{2IC(t)}{IC(t_1) + IC(t_2)} \tag{4}$$

$$\mathrm{sim}_{\mathrm{Jiang}} = 1 - (IC(t_1) + IC(t_2) - 2IC(t)) \tag{5}$$

where t is the MICA of t_1 and t_2. From these equations, it can be seen that, if the IC values of t_1 and t_2 increase, which means their specificities increase, the similarity will decrease.

From equations (4) and (5), it can also be seen that, when t_1 is equal to t_2, the semantic similarity of the two terms will correspond to the value of 1, which indicates that, compared with itself, the similarity will stay at the largest value regardless of the specificity of the term. However, since the terms at the top levels of GO are less specific than the leaf terms, the similarities between these terms and themselves should be smaller accordingly. To address this issue, (Schlicker et al., 2006) revised Lin's method by incorporating a weight item as shown in equation (6) below, and referred to it as the Relevance method:

$$\mathrm{sim}_{\mathrm{rel}}(t_1, t_2) = \mathrm{sim}_{\mathrm{Lin}}(t_1, t_2)\,(1 - p(t)) \tag{6}$$

In addition to the edge-based and node-based methods, there are also a number of hybrid methods proposed, e.g., Wang's method (Wang et al., 2007). In this method, a term A can be represented as a DAG structure, where $DAG_A = (A, T_A, E_A)$. T_A is a set containing term A and all its ancestors in a GO graph. E_A is a set containing edges that connect terms in T_A. To compute the semantic similarity, Wang's method also defines an S-value for each term in the set T_A. The S-value of a term t in T_A, $S_A(t)$, can be calculated using equation (7) below.

$$\begin{cases} S_A(A) = 1 \\ S_A(t) = \max\{w_e \times S_A(t') \mid t' \in \text{descendents}(t)\} \quad \textit{if } t \neq A \end{cases} \tag{7}$$

w_e is the weight of the edge e in E_A connecting term t and its descendent t′For an 'is-a' relationship, the weight is 0.8, and for a 'part-of' relationship, the weight is 0.6.

The semantic similarity of two terms A and B is computed based on their DAG structures using equation (8) below.

$$\mathrm{sim}_{\mathrm{Wang}}(A, B) = \frac{\sum_{t \in T_A \cap T_B} (S_A(t) + S_B(t))}{\sum_{t \in T_A} S_A(t) + \sum_{t \in T_B} S_B(t)} \tag{8}$$

After calculating the semantic similarity over the terms, the next step is to define a measure for the semantic similarity over gene products. The often used methods are the 'Max' (Wu et al., 2005) and 'Ave' (Wang et al., 2005) methods. Given two gene products g_1 and g_2, the semantic similarities between their annotations form a semantic similarity

matrix. For the 'Max' method, the semantic similarity is the maximum value in the matrix. For 'Ave' method, it is the average value over the matrix. They can be computed using equations (9) and (10) below respectively.

$$\mathrm{sim}_{\mathrm{Max}}(g_1, g_2) = \max_{\substack{t_1 \in \mathrm{annotation}(g_1) \\ t_2 \in \mathrm{annotation}(g_2)}} \mathrm{sim}(t_1, t_2) \tag{9}$$

$$\mathrm{sim}_{\mathrm{Ave}}(g_1, g_2) = \mathrm{average}_{\substack{t_1 \in \mathrm{annotation}(g_1) \\ t_2 \in \mathrm{annotation}(g_2)}} \mathrm{sim}(t_1, t_2) \tag{10}$$

where annotation(g_1) and annotation(g_2) are annotations for g_1 and g_2. There are some more complicated methods defined for special structures, e.g., the method proposed by (Wang et al., 2007). The evaluation of these methods shows that the similarity obtained using the 'Max' method is best correlated with the gene expression data. However, the 'Max' method is more sensitive to outliers, while the 'Ave' method is relatively stable (Xu et al., 2008).

3 Method

3.1 The SP algorithm

As mentioned in Section 2, for edge-based methods, the weights of the edges conflict with the property of GO, and for node-based methods, only IC values of the two terms and their MICAs are considered regardless of their positions in GO. To address these drawbacks, we propose a new hybrid method, namely the Shortest Path (SP) algorithm, to measure the semantic similarity over terms in GO.

It is intuitive that, given two terms t_A and t_B, if they diverge at a higher level (i.e., their MICA is nearer to the root), the difference between them should be larger; while if they diverge at a lower level, the difference should decrease. Under this assumption, the SP algorithm first assigns the weights to the GO terms using the reciprocal of their IC values. With an increase of the term's IC value, i.e., the increase of its specificity, its weight will decrease. Then the algorithm finds the path connecting the two terms and their MICA with the smallest sum of weights, and defines the sum of the weights on the path as the semantic distance for the terms. This path is referred to as the shortest path. The rationale behind the algorithm is that, if MICA is near the root, the weights on the shortest path will increase and *vice versa*. Therefore, the sum of the weights on the shortest path is consistent with the expected distance and can be used as its estimation.

The SP algorithm can be described as follows. Given two terms t_A and t_B, the normalised distance between them is defined as:

$$\mathrm{dist}_{\mathrm{SP}}(t_A, t_B) = \frac{\arctan\left(\sum_{t_1 \in \mathrm{path}_A} \frac{1}{IC[t_1]} + \sum_{t_2 \in \mathrm{path}_B} \frac{1}{IC[t_2]}\right)}{\pi / 2} \tag{11}$$

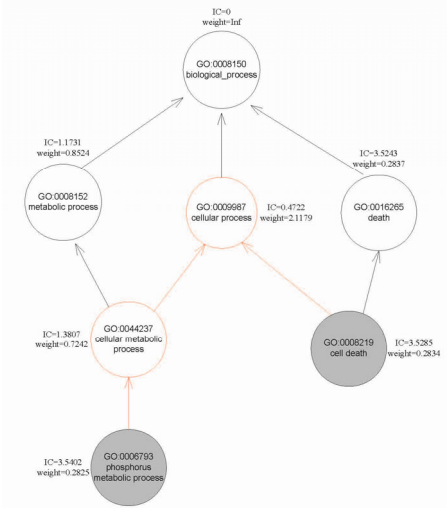
where path_A (path_B) is the shortest path that connects the term t_A (t_B) with MICA; t_1 and t_2 are the terms located on path_A and path_B. Because MICA appears in both path_A and path_B, it is considered only once in equation (11). The function of *arctan* is to normalise the distance obtained by summing the weights of the terms on the shortest path to [0, 1]. After the normalisation, the semantic similarity can be defined as

$$\text{sim}_{\text{SP}}(t_A, t_B) = 1 - \text{dist}_{\text{SP}}(t_A, t_B) \tag{12}$$

We propose to use Dijkstra algorithm to determine the shortest path connecting MICA and t_A (t_B), and we need to transform the weights associated with nodes to the corresponding edge weights. The weight of each edge is assigned as the weight of the more specific term between the two that it connects to. Dijkstra algorithm is then used to find the shortest path on the new edge-weighted graph. In equation (11), when the weights on path_A or path_B increase, i.e., MICA and its descendents on the shortest path become more general, the distance increases and the semantic similarity decreases.

We show an example of computing the semantic similarity for the terms GO: 0006793 and GO: 0008219 in Figure 2. In the first step, the SP algorithm calculates the IC values for the terms on the graph using equation (2). In practice, several R packages (e.g., (Froehlich et al., 2007)) provide IC information for GO terms. Therefore, the IC values can be retrieved from these packages when needed. In this example, $\text{IC}_{\text{GO: 0006793}} = 3.5402$ and $\text{IC}_{\text{GO: 0008219}} = 3.5285$. In the second step, SP algorithm weights each term using the value of 1/IC, i.e., $\text{weight}_{\text{GO:0006793}} = 1/3.5402 = 0.2825$, and $\text{weight}_{\text{GO:0008219}} = 1/3.5285 = 0.2834$. The weights of the other terms are computed in a similar way. In the third step, SP algorithm finds the MICA (GO: 0009987) and the shortest paths connecting MICA and the two terms. In this example, there is only one path starting from GO: 0006793 (GO: 0008219) to MICA. Therefore, it is marked as the shortest path and shown in Figure 2 in red. The distance between GO: 0006793 and GO: 0008219 is the normalised sum of weights on the shortest paths, i.e., dist = arctan $(0.2825 + 0.7242 + 2.1179 + 0.2834)/(\pi/2) = 0.8183$. The semantic similarity *sim* is calculated as $1 - 0.8183 = 0.1817$.

Figure 2 An example of semantic similarity computation for GO: 0008219 and GO: 0006793. Paths in red are the Shortest Paths connecting the two terms and their MICAs (see online version for colours)



SP algorithm integrates the information from two sources, which are the structure information contained in the paths connecting the terms, and the IC information of the terms represented by the weights on the graph. When searching for the shortest path, both

structure and IC information will be considered, unlike the existing edge-based/node-based methods that use only structure/IC information.

3.2 Validation method

Several types of data can be used to assess the accuracy of existing methods for measuring semantic similarity. In this paper, we use both PPI data and gene expression datasets to evaluate the correctness of computed semantic similarities.

3.2.1 Assessment of semantic similarity based on protein-protein interactions

The validation process using PPI information is as follows. First, for each pair of proteins (interacting pair or otherwise), their GO annotations (represented by GO terms) are retrieved from a suitable biological database (e.g., SGD, Uniprot). Then, semantic similarities over these GO terms are computed using the methods to be evaluated. After that, the semantic similarities over proteins can be calculated by existing methods for measuring semantic similarity over genes, e.g., the 'Max' or 'Avg' method. An accurate method means that, for proteins that have interactions, their semantic similarities should be large. On the contrary, for proteins that have no interactions, their semantic similarities should be small.

To quantitatively assess the correctness of computed semantic similarity using PPI information, Receiver Operating Characteristic (ROC) curve analysis is adopted. ROC curves are plotted based on the True Positive Rate (TPR) and the False Positive Rate (FPR) (defined below).

$$\mathrm{TPR} = \frac{\mathrm{TruePositive}}{\mathrm{TruePositive} + \mathrm{FalseNegative}} \tag{13}$$

$$\mathrm{FPR} = \frac{\mathrm{FalsePositive}}{\mathrm{TrueNegative} + \mathrm{FalsePositive}} \tag{14}$$

Area Under a Curve (AUC) is computed for each ROC curve to measure the accuracy of the corresponding methods. A larger AUC value indicates a higher accuracy for a particular method.

3.2.2 Assessment of semantic similarity based on gene expression data

The validation process using gene expression data is similar to the process using PPI data. For genes in the expression profile, their semantic similarities are first computed in the same way as described in Section 3.2.1.

To assess the correctness of computed semantic similarity using gene expression data means that we need to find out whether the estimated semantic similarity is in line with the similarity based on the expression data. In general, a higher correlation indicates a better performance. The Pearson correlation coefficient is often used to evaluate the linear dependency between two variables. When dealing with the nonlinear dependency problem, evaluation using Pearson correlation will not be suitable. Here we introduce an approach for characterising nonlinear correlation (Sheikh et al., 2006).

Given a set of points $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, the first step of the algorithm is to apply regression analysis to find a fitting curve $f(x)$. The curve corresponds to a

nonlinear mapping between $\mathbf{x} = \{x_1, \ldots, x_n\}$ and $\mathbf{y} = \{y_1, \ldots, y_n\}$. In the next step, the Pearson correlation coefficient is calculated between $\mathbf{y}$ and $f(\mathbf{x})$ after nonlinear regression. In addition to the Pearson correlation coefficient, another metric referred to as Root Mean Squared Error (RMSE) is also calculated between $\mathbf{y}$ and $f(\mathbf{x})$ using equation (15) below. RMSE is used to measure the difference between values predicted by the model (i.e., $f(\mathbf{x})$) and the observed values (i.e., the value of $\mathbf{y}$). A smaller RMSE corresponds to a better prediction model.

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - f(x_i))^2}{n}}. \tag{15}$$

4 Experiments and results

In the experiments, we evaluated the performance of our SP method together with other four state-of-the-art methods for measuring the semantic similarity over the terms. In Section 4.1, we will present the experimental details, including the description of the datasets and the experimental setup. Then, in Section 4.2, the experimental results together with some explanations will be given.

4.1 Data description and experimental setup

We downloaded 4510 pairs of *S. cerevisiae* protein-protein interactions from the Database of Interacting Proteins (DIP) (Scere20100614, core version) (Xenarios et al., 2002). Interactions between the same proteins are removed from the list. The remaining protein pairs are used as the positive samples in the experiments. Besides, we randomly constructed 3377 pairs of proteins which are not in the DIP and used them as the negative samples.

In addition, we used two gene expression datasets in the experiments. The first one is the Eisen dataset (Eisen et al., 1998), which consists of 2467 genes. The second one is the Spellman dataset (Spellman et al., 1998), containing 6178 genes. The details of the datasets are described in Table 1. The missing values in the two datasets were filled in using the impute package from the bioconductor project (Gentleman et al., 2004).

The annotations for proteins and genes were retrieved from the Saccharomyces Genome Database (SGD). In our experiments, we used the annotations from BP ontology. According to (Xu et al., 2008), terms at the top levels will create noise. Therefore, in our experiments, annotations at the first three levels were removed. Proteins and genes that are annotated only by these general terms were removed afterwards. In the end, 3184 positive protein interactions, 3348 negative protein interactions, 2461 genes in the Eisen dataset, and 5545 genes in the Spellman dataset were used. These proteins/genes together with their BP annotations and expression data can be downloaded from http://www.cs.cityu.edu.hk/~yingshen/IJDMB/data/data.zip.

The similarity based on the gene expression data is calculated using the Pearson correlation and is referred to as the expression similarity.

We evaluated another four state-of-the-art methods including Resnik's (Resnik, 1999), Jiang's (Jiang and Conrath, 1997), the Relevance (Schlicker et al., 2006) method from the node-based category, and Wang's (Wang, 2007) method from the hybrid category, as a comparison to the SP algorithm. We used the GOSim package (Froehlich

et al., 2007) to calculate the semantic similarity for Resnik's, Jiang's and the Relevance methods, and the GOSemSim package (Yu et al., 2010) for Wang's method.

To compute the semantic similarity over gene products, we used the 'Max' operation for all the five methods, since it consistently results in the best correlation scores for all the methods measuring the semantic similarity over the terms (Xu et al., 2008).

Table 1 Gene expression datasets used in the experiments

	No. of genes	*No. of experiments*	*Species*
Eisen	2467	79	yeast
Spellman	6178	77	yeast

4.2 Experimental results

4.2.1 Results based on PPI data

We first sorted the semantic similarities between proteins in descending order. Then we computed TPR and FPR according to the labels. Resulting ROC curves for the five methods are shown in Figure 3. Corresponding AUC values are listed in Table 2. It can be seen that the SP method achieves the best AUC score among all the five methods. Although the AUC score of Wang's method is only slightly smaller to that of our method, it has the significant disadvantage of long computation time, on which we will provide further explanation in Section 4.2.2.

Figure 3 ROC curves for the five methods (see online version for colours)

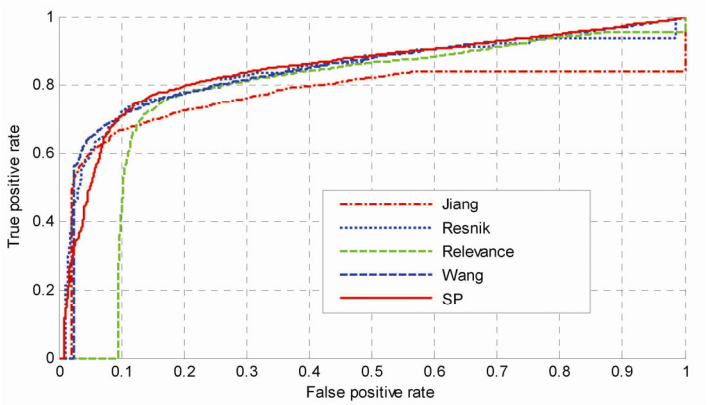


Table 2 AUC values for the ROC curves shown in Figure 3

	Jiang	*Resnik*	*Wang*	*Relevance*	*SP*
AUC	0.7713	0.8397	0.8426	0.7798	*0.8457*

4.2.2 Results based on gene expression data

In the previous works, Pearson correlation was used to evaluate the consistency between the semantic similarity and the expression similarity. We first equally divided the interval [0, 1] into 1000 sub-intervals. Then we calculated the semantic similarity and the expression similarity for each gene pair. All the gene pairs were assigned to the

sub-intervals according to their absolute gene expression similarity. After that, we calculated the average semantic similarity for each interval. Finally, the Pearson correlation coefficient value was calculated between the average semantic similarity and the expression similarity. The correlation coefficients for the different methods are shown in Table 3. From this table, it can be seen that the SP method proposed by us performs the best among the five methods. Specifically, our method achieves the correlation coefficient values of 82.8% and 86.9% on the Eisen and Spellman datasets respectively, which are about 3% higher than the second best methods (i.e., Wang's method on the Eisen dataset and Resnik's method on the Spellman dataset), and a more significant improvement over the others.

Next, we calculated the nonlinear correlation coefficients and RMSE values for the average semantic similarity and the expression similarity using the method introduced in Section 4. The following mapping is used for the nonlinear regression analysis (Sheikh et al., 2006; Zhang et al., 2010; Zhang et al., 2011):

$$f(x) = a_1\left(\frac{1}{2} - \frac{1}{1+\mathrm{e}^{a_2(x-a_3)}}\right) + a_4 x + a_5 \tag{16}$$

where a_i, $i = 1, 2, \ldots, 5$ are parameters to be determined.

Table 3 Pearson correlation coefficient

	Jiang	*Resnik*	*Wang*	*Relevance*	*SP*
Eisen	0.7901	0.8031	0.7955	0.7817	*0.8278*
Spellman	0.7436	0.8383	0.8368	0.7718	*0.8685*

Table 4 Nonlinear correlation and RMSE on Eisen dataset

	Jiang	*Resnik*	*Wang*	*Relevance*	*SP*
corr	0.9939	0.9602	*0.9950*	0.9581	0.9892
RMSE	0.0249	0.0299	0.0190	0.0321	*0.0166*

Table 5 Nonlinear correlation and RMSE on Spellman dataset

	Jiang	*Resnik*	*Wang*	*Relevance*	*SP*
corr	0.9133	0.9295	0.9636	0.8869	*0.9656*
RMSE	0.0738	0.0518	0.0435	0.0553	*0.0354*

Table 6 Time consumption (sec) for the example shown in Figure 1

	Jiang	*Resnik*	*Wang*	*Relevance*	*SP*
Time	0.02	0.01	0.42	0.01	0.15

The correlation coefficients and RMSE values on the Eisen dataset and the Spellman dataset are listed in Tables 4 and 5 respectively. Figure 4 shows the scatter plots of the min-max normalised gene semantic similarity vs. the expression similarity for the five methods on the two datasets. Min-max normalisation means that the minimum value is

mapped to 0, the maximum value is mapped to 1, and the other values are linearly rescaled accordingly. In this way, the various scatter plots, which correspond to methods with different dynamic ranges for the semantic similarity values, can be more easily compared with each other. The curves are obtained from a nonlinear fitting using the model in equation (16).

Figure 4 Scatter plots of the min-max normalised semantic similarity vs. the gene expression similarity on the Eisen and Spellman dataset. 1(a)–1(e) Eisen dataset; 2(a)–2(e) Spellman dataset. Five methods are compared: (a) Jiang's method; (b) Resnik's method; (c) Wang's method; (d) Relevance method and (e) the Shortest Path method (see online version for colours)

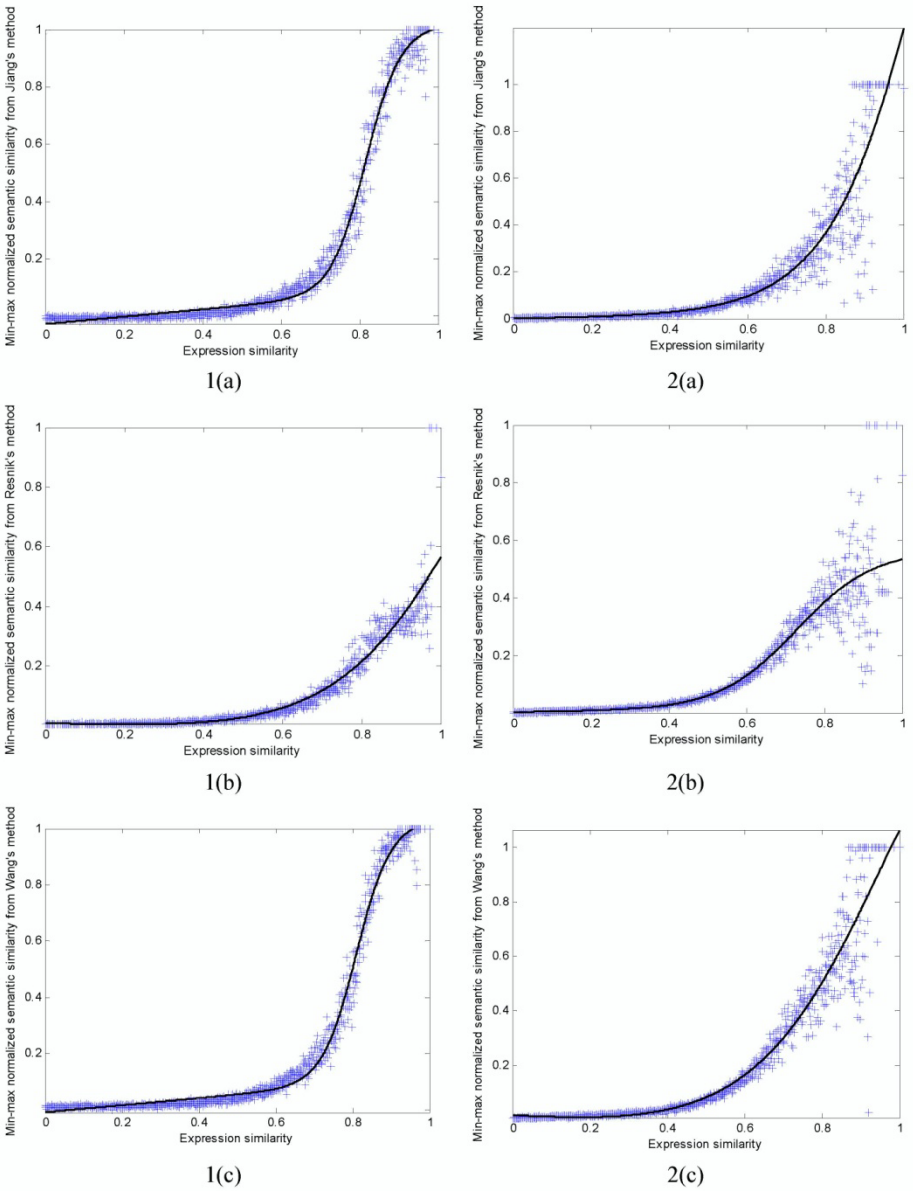
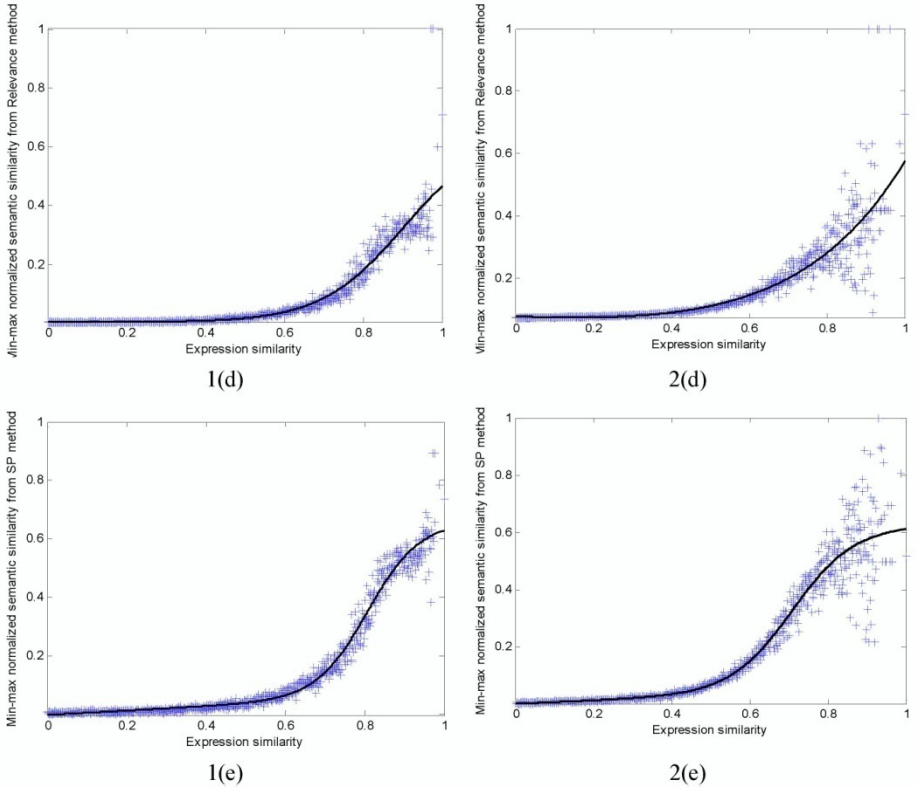


Figure 4 Scatter plots of the min-max normalised semantic similarity vs. the gene expression similarity on the Eisen and Spellman dataset. 1(a)–1(e) Eisen dataset; 2(a)–2(e) Spellman dataset. Five methods are compared: (a) Jiang's method; (b) Resnik's method; (c) Wang's method; (d) Relevance method and (e) the Shortest Path method (see online version for colours) (continued)



Wang's method and our SP method have similar performance with respect to the AUC score, the nonlinear correlation coefficient, and the RMSE scores, and they are much better than the other three methods. However, Wang's method requires a long computation time. First, comparing equation (8) with equation (11), it can be seen that, to compute semantic similarity using Wang's method, we need to compute the *S*-values of all ancestors of term *A* and term *B* in advance. In contrast, our SP method only requires the IC values of their common ancestors, the number of which is much smaller. Second, in equation (8), the *S*-values of term *t* to term *A* and term *B* are different. Therefore, the *S*-values of term *t* cannot be stored and reused, resulting in computational redundancy. On the other hand, the IC value of a term used in SP method is fixed, which can be stored and easily retrieved from a database. To support these claims, we report the time consumption (in sec) in Table 6 for the five methods on the example shown in Figure 2. It can be seen that the time consumption of Wang's method is about 3 times that of SP method.

5 Conclusion

In this paper, a new method for measuring the semantic similarity, namely the SP algorithm, is proposed. The SP algorithm depends on the substructure of GO associated with two terms and their MICA. The substructure contains more information than the IC values used in the node-based algorithm. In addition, the weights assigned to the substructure are more consistent than the previous edge-based methods. In general, the semantic similarity obtained by SP algorithm correlates better with PPI information and expression similarity than other node-based methods. Moreover, compared with another state-of-the art hybrid method, Wang's method, the SP algorithm has the advantage of less computation time due to fewer variables.

Acknowledgment

This work is supported by a grant from the City University of Hong Kong under grant no. 7008044 and the Natural Science Foundation of China under grant no. 61201394.

Reference

Chen, C. and Tang, J. (2010) 'Using gene ontology to enhance effectiveness of similarity measures for microarray data', *J. Data Mining and Bioinformatics*, Vol. 4, pp.520–534.

Cheng, J., Cline, M., Martin, J., Finkelstein, D., Awad, T., Kulp, D. and Siani-Rose, M. A. (2004) 'A knowledge–based clustering algorithm driven by gene ontology', *Journal of Biopharmaceutical Statistics*, Vol. 14, pp.687–700.

Denaxas, S.C. and Tjortjis, C. (2008) 'Scoring and summarising gene product clusters using the gene ontology', *J. Data Mining and Bioinformatics*, Vol. 2, pp.216–235.

Dotan–Cohen, D., Kasif, S. and Melkman, A. (2009) 'Seeing the forest for the trees: using the gene ontology to restructure hierarchical clustering', *Bioinformatics*, Vol. 25, pp.1789–1795.

Eisen, M., Spellman, P., Brown, P. and Botstein, D. (1998) 'Cluster analysis and display of genome-wide expression patterns', *PNAS*, Vol. 95, pp.14863–14868.

Froehlich, H., Speer, N., Poustka, A., Beissbarth, T. (2007) 'GOSim – an R-package for computation of information theoretic GO similarities between terms and gene products', *BMC Bioinformatics*, Vol. 8, p.166.

Gentleman, R., Carey, V., Bates, D., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry R., Leisch, F., Li, C., Maechler, M., Rossini, A., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J. and Zhang, J. (2004) 'Bioconductor: Open software development for computational biology and bioinformatics', *Genome Biology*, Vol. 5, p.R80.

Jiang, J. and Conrath, D. (1997) 'Semantic similarity based on corpus statistics and lexical taxonomy', *Proc. Int. Conf. Research in Computational Linguistics*, Taiwan, pp.19–33.

Lin, D. (1998) 'An information-theoretic definition of similarity', *Proc. Int. Conf. Machine Learning*, Madison, Wisconsin, USA, pp. 296–304.

Lord, P.W., Stevens, R.D., Brass, A. and Goble, C. (2003) 'Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation', *Bioinformatics*, Vol. 19, pp.1275–1283.

Pandey, G., Myers, C. and Kumar, V. (2009) 'Incorporating functional inter-relationships into protein function prediction algorithms', *BMC Bioinformatics*, Vol. 10, pp.1471–2105.

Pekar, V. and Staab, S. (2002) 'Taxonomy learning: factoring the structure of a taxonomy into a semantic classification decision', *Proc. Int. Conf. Computational Linguistics*, Taiwan, pp.786–792.

Resnik, P. (1999) 'Semantic similarity in taxonomy: An information-based measure and its application to problems of ambiguity innatural language', *Journal of Artificial Intelligence Research*, Vol. 11, pp.95–130.

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., Tetko, I., Güldener, U., Mannhaupt, G., Münsterkötter, M. and Mewes, H.W. (2004) 'The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes', *Nucl. Acids Res.*, Vol. 32, pp.5539–5545.

Schlicker, A., Domingues, F., Rahnenführer, J. and Lengauer, T. (2006) 'A new measure for functional similarity of gene products based on gene ontology', *BMC Bioinformatics*, Vol. 7, pp.302.

SGD project, http://www.yeastgenome.org/.

Sheikh, H., Sabir, M. and Bovik, A. (2006) 'A statistical evaluation of recent full reference image quality assessment algorithms', *IEEE Trans. Image Processing*, Vol. 15, pp.3440–3451.

Spellman, P., Sherlock, G., Zhang, M., Iyer, V., anders, K., Eisen, M., Brown, P., Botstein, D. and Futcher, B. (1998) 'Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization', *Molecular Biology of the Cell*, Vol. 9, pp.3273–3297.

The Gene Ontology Consortium (2000) 'Gene ontology: tool for the unification of biology', *Nature Genetics*, Vol. 25, pp.25–29.

Wang, H., Azuaje, F. and Bodenreider, O. (2005) 'An ontology-driven clustering method for supporting gene expression analysis, computer-based medical systems', *Proc. IEEE Symposium on Computer–based Medical Systems*, Ireland, pp.23–24.

Wang, J., Du, Z., Payattakool, R., Yu, P.S. and Chen, C. (2007) 'A new method to measure the semantic similarity of GO terms', *Bioinformatics*, Vol. 23, pp.1274–1281.

Wu, H., Su, Z., Mao, F., Olman, V. and Xu, Y. (2005) 'Prediction of functional modules based on comparative genome analysis and gene ontology application', *Nucl. Acids Res.*, Vol. 33, pp.2822–2837.

Xenarios, I., Salwínski, L., Duan, X.J., Higney, P., Kim, S.M. and Eisenberg, D. (2002) 'DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions', *Nucl. Acids Res.*, Vol. 30, pp.303–305.

Xu, T., Du, L. and Zhou, Y. (2008) 'Evaluation of GO-based functional similarity measures using S. cerevisiae protein interaction and expression profile data', *BMC Bioinformatics*, Vol. 9, pp.472.

Yu, G., Li, F., Qin, Y., Bo, X., Wu, Y. and Wang, S. (2010) 'GOSemSim: an R package for measuring semantic similarity among GO terms and gene products', *Bioinformatics*, Vol. 26, pp.976–978.

Zhang, L., Zhang, L. and Mou, X. (2010) 'RFSIM: a feature based image quality assessment metric using Riesz transforms', *Proc. Int. Conf. Image Processing*, pp.321–324.

Zhang, L., Zhang, L., Mou, X. and Zhang, D. (2011) 'FSIM: a feature similarity index for image quality assessment', *IEEE Trans. Image Processing*, Vol. 20, pp.2378–2386.

Zhu, S., Zeng, J. and Mamitsuka, H. (2009) 'Enhancing MEDLINE document clustering by incorporating MeSH semantic similarity', *Bioinformatics*, Vol. 25, pp.1944–1951.