BIOINFORMATICS ORIGINAL PAPER

Vol. 25 no. 14 2009, pages 1739–1745
doi:10.1093/bioinformatics/btp309

Sequence analysis

ESG: extended similarity group method for automated protein function prediction

Meghana Chitale[1], Troy Hawkins[2], Changsoon Park[3,*] and Daisuke Kihara[2,1,4,*]

[1]Department of Computer Science, [2]Department of Biological Sciences, Purdue University, IN 47907, USA, [3]Department of Statistics, Chung-Ang University, Seoul 156-756, Korea and [4]Markey Center for Structural Biology, Purdue University, IN 47907, USA

Received on January 8, 2009; revised on April 16, 2009; accepted on May 5, 2009

Advance Access publication May 12, 2009

Associate Editor: Limsoon Wong

ABSTRACT

Motivation: Importance of accurate automatic protein function prediction is ever increasing in the face of a large number of newly sequenced genomes and proteomics data that are awaiting biological interpretation. Conventional methods have focused on high sequence similarity-based annotation transfer which relies on the concept of homology. However, many cases have been reported that simple transfer of function from top hits of a homology search causes erroneous annotation. New methods are required to handle the sequence similarity in a more robust way to combine together signals from strongly and weakly similar proteins for effectively predicting function for unknown proteins with high reliability.

Results: We present the extended similarity group (ESG) method, which performs iterative sequence database searches and annotates a query sequence with Gene Ontology terms. Each annotation is assigned with probability based on its relative similarity score with the multiple-level neighbors in the protein similarity graph. We will depict how the statistical framework of ESG improves the prediction accuracy by iteratively taking into account the neighborhood of query protein in the sequence similarity space. ESG outperforms conventional PSI-BLAST and the protein function prediction (PFP) algorithm. It is found that the iterative search is effective in capturing multiple-domains in a query protein, enabling accurately predicting several functions which originate from different domains.

Availability: ESG web server is available for automated protein function prediction at http://dragon.bio.purdue.edu/ESG/

Contact: cspark@cau.ac.kr; dkihara@purdue.edu

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

1 INTRODUCTION

Developments in genomics and proteomics areas over the last decade have upshot into growing amount of newly sequenced genomes and large-scale experimental data that require computational assistance for predicting functions (Hawkins and Kihara, 2007; Hawkins *et al*., 2008). Conventionally, sequence homology has been used as the key information source for transferring annotations by comparing the new sequences with a database of annotated genes. Although considering homology is a genuine way of inferring function in the light of evolution, practically, it is not always trivial to extract correct function information from a sequence database search result.

Tian and Skolnick (2003) have observed that enzyme function starts diverging quickly when the sequence identity falls below 70% and E-values from PSI-BLAST are not always strongly correlated to enzyme function conservation. Studies (Devos and Valencia, 2001; Friedberg, 2006) have shown that the traditional annotation transfer methods may be reliable for very high sequence similarity but are likely to be erroneous in many situations. Even though sequence similarity strongly indicates functional similarity in many cases (Duan *et al*., 2006), methods for interpreting results of homology search needs to be expanded to consider weakly similar hits. Concepts of phylogenetic similarity, structural similarity and functional similarity are not necessarily interchangeable with the sequence homology, and these terms map imperfectly with each other (Fitch, 2000). It was demonstrated that non-critical transfer of annotations from similar sequences while ignoring the multi-domain architecture can result in serious prediction inaccuracies (Galperin and Koonin, 1998).

To capture the complex relation between sequence similarity and function, many methods have been developed. John and Sali (2004) and Park *et al*. (1997) have explored the concept of using an intermediate sequence to relate two weakly similar proteins that can be potential homologs. Song *et al*. (2008) have developed a method based on sequence homology network that compares local neighborhood network of proteins for deciding the functional similarity.

Recently, there have been efforts on developing automatic function prediction methods which are built on BLAST (Altschul *et al*., 1990) and PSI-BLAST. OntoBlast (Zehetner, 2003) uses the E-values of sequence hits in a BLAST search to score Gene Ontology (GO) terms. GOFigure (Khan *et al*., 2003) and GOtcha (Martin *et al*., 2004) incorporate the hierarchical structure of GO vocabulary (Harris *et al*., 2004) for scoring parental GO terms in the hierarchy. GOAnno (Plewniak *et al*., 2003) extracts GO terms from subfamilies of a multiple sequence alignment. GOPET (Vinayagam *et al*., 2006) and ProtFun (Jensen *et al*., 2003) apply support vector machine to sequence similarity features obtained from a BLAST search. We have previously developed protein function prediction (PFP) (Hawkins *et al*., 2006, 2009) that incorporates E-value-based scoring of GO terms along with the functional association between GO terms and parental GO term scoring. PFP is very successful,

*To whom correspondence should be addressed.

which is evidenced by the top rank in the function prediction category in AFP-SIG 05 (Friedberg *et al*., 2006) and CASP7 (Lopez *et al*., 2007) competitions.

Here, we report our new method, extended similarity group (ESG), which is shown to significantly outperform PFP and conventional PSI-BLAST search in a large benchmark dataset of 2400 genes. In addition, we find that the iterative search is able to capture functions of a protein, which originate from multiple functional domains in the sequence.

2 METHODS

ESG uses PSI-BLAST (Altschul *et al*., 1997) (version 2.2.18) as a sequence database search tool. The maximum number of passes is set to three (−j 3). The default inclusion E-value is used (−h 0.005). As described below, ESG assigns a probability to GO terms based on the sum of relative significance of E-value of sequences annotated with the GO terms.

2.1 Computing the annotation probability

We begin with N sequences retrieved by performing PSI-BLAST search with the query sequence Q (Fig. 1A). This is called the first level of extended similarity group, from which searches are further iterated. The sequences obtained by PSI-BLAST search from the first-level ESG sequences are called the second-level ESG sequences and so on for the next levels. Thus, ESG covers multiple-level neighborhoods around the query protein. For each query protein Q, we define the probability that Q has a particular GO annotation f_a, and this direct probability is given by $P_Q^d(f_a)$. Here, we use two level neighbors but computations can be extended for multiple levels with an exponential increase of computation time with respect to N.

2.2 ESG level one computation

Let $S_1, S_2, \dots, S_N$ be the PSI-BLAST hit sequences for Q each with E-values $E_1, E_2, \dots, E_N$, respectively (Fig. 1A). Every sequence gets a weight equal to its relative E-value with respect to the E-values of all N sequences [Equation (1)]. In the computation of weights, we use sequences with an E-value of up to 1000. The score is shifted by a constant, b, which is given by log(1000) to make the score a non-negative value. Using the binary function $I_{S_i}(f_a)$ [Equation (2)], the direct probability $P_Q^d(f_a)$ for each annotation f_a assigned to S_i is the sum of weights of sequences with the annotation f_a [Equation (3)].

$$W_i = \frac{-\log(E_i)+b}{\sum_{j=1}^{N}\{-\log(E_j)+b\}} \tag{1}$$

$$I_{S_i}(f_a) = \begin{cases} 1 & \text{if } S_i \text{ has } f_a \text{ annotation} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$P_Q^d(f_a) = \sum_{i=1}^{N} W_i \cdot I_{S_i}(f_a) \tag{3}$$

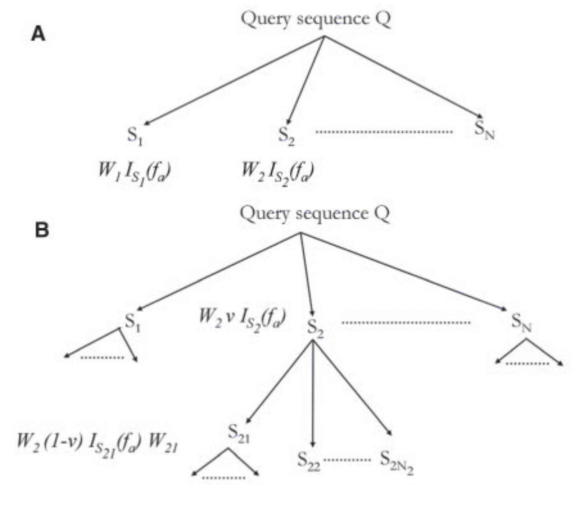

Fig. 1. (**A**) ESG computation with one level. Sequences S_1 to S_N are retrieved by PSI-BLAST search. (**B**) ESG with two levels. The second round of PSI-BLAST searches are performed from each of the sequences, S_1 to S_N.

2.3 ESG multilevel computation

For probability computation using two level neighbors of query sequence Q, we perform the second iteration of PSI-BLAST search from each of the sequences S_i obtained at the first level (Fig. 1B). Thus, from each sequence S_i we obtain N_i sequences, S_{ij} ($1 \le j \le N_i$). Each of these sequences has a weight W_{ij}, which is computed by applying Equation (1) to the PSI-BLAST search from S_i. Weight W_i of sequence S_i at first level is distributed between sequence S_i and all its child nodes S_{ij} using a stage weight parameter v ($0 \le v \le 1$). This parameter decides the proportion of W_i that is allocated to S_i itself and its children together [Equation (4)].

$$P_{S_i}^d(f_a) = v \cdot I_{S_{ij}}(f_a) + (1-v) \cdot \sum_{j=1}^{Ni} W_{ij} \cdot I_{S_{ij}}(f_a) \tag{4}$$

$$P_Q(f_a) = \sum_{i=1}^{N} W_i \cdot P_{S_i}^d(f_a) \tag{5}$$

To compute the final $P_Q^d(f_a)$ the weights of sequences at two ESG levels that have annotation f_a are summed up in proportion of step weights as shown in Figure 1B and Equation (5). Equations described here can be easily extended for more number of iterative levels.

2.4 ESG with function association matrix

Function association matrix (FAM) is used by PFP to capture observed correlation between GO terms in the annotation database. GO annotations have a hierarchical structure with three distinct annotation categories, i.e. molecular function, biological process and cellular component. Some of these terms even pairs across different categories frequently occur together in the database to annotate the same protein. FAM captures association of two GO terms, f_a and f_j, as a conditional probability:

$$P(f_a|f_j) = \frac{c(f_a,f_j)+\varepsilon}{c(f_j)+\varepsilon \cdot \mu} \tag{6}$$

where $c(f_a,f_j)$ is the number of times f_a and f_j are assigned simultaneously to a sequence and $c(f_j)$ is the total number of times f_j appeared in UniProt. μ is the size of one dimension of the FAM (i.e. the total number of unique GO terms), and ε is the pseudo-count (we used 0.05 for ε).

$$P_Q^{\mathrm{FAM}}(f_a) = \sum_{i=1}^{N} W_i \cdot P_{S_i}^{\mathrm{FAM}}(f_a) \tag{7}$$

When using the two level system as described previously, we follow Equation (7) together with Equation (4) except for replacing the probability scores $P_{S_{ij}}^d(f_a)$ coming from each sequence hit S_i by FAM probability $P_{S_i}^{\mathrm{FAM}}(f_a)$. In Equation (8), we compute the FAM probability for sequence S_i. $I_{s_i}(f_a)$ is the binary function [Equation (2)], f_j are annotations of S_i, $P(f_a\,|\,f_j)$ is the FAM [Equation (6)], N_{Si} is number of GO annotations for S_i, v is the stage weight parameter. For the FAM probability given to the sequences found in the second level, i.e. $P_{S_{ij}}^{\mathrm{FAM}}(f_a)$, Equation (9) is used. Equations (8) and (9) show that FAM is used only when the sequence S_i is not annotated with f_a (i.e. $I_{S_i}(f_a)=0$).

$$P^{\mathrm{FAM}}_{S_i}(f_a)=v\left\{I_{s_i}(f_a)+(1-I_{s_i}(f_a))\cdot\max\left(\sum_{j=1}^{N_{si}}P(f_a|f_j),\,1\right)\right\}$$
$$+(1-v)\left\{\sum_{j=1}^{N_{ij}}W_{ij}\cdot P^{\mathrm{FAM}}_{S_{ij}}(f_a)\right\} \qquad (8)$$

$$P^{\mathrm{FAM}}_{S_{ij}}(f_a)=I_{s_{ij}}(f_a)+(1-I_{s_{ij}}(f_a))\cdot\max\left(\sum_{j=1}^{N_{ij}}P(f_a|f_j),\,1\right) \qquad (9)$$

In all the FAM computations, only the strong associations with a conditional probability, $P(f_a|f_j)$, of >0.7 are considered.

2.5 Top PSI-BLAST method

In the Top PSI-BLAST method, PSI-BLAST search is performed with a default setting with maximum of three iterations. Then the top hit with an E-value score better than 0.01 that has annotations is used for transferring annotations to the query sequence.

2.6 PFP method

PFP uses PSI-BLAST to retrieve sequences similar to query sequence and assigns scores to GO terms associated with these hits based on E-values. It also incorporates FAM that captures association between GO terms.

$$s(f_a)=\sum_{i=1}^{N}\sum_{j=1}^{N\mathrm{func}(i)}\big((-\log(E\text{-value}(i))+b\big)P(f_a|f_j)\big) \qquad (10)$$

Each GO term is assigned a score as shown in Equation (10), where $s(f_a)$ is the final raw score assigned to the GO term, f_a, N is the number of sequences retrieved by PSI-BLAST, Nfunc(i) is the number of GO terms assigned to sequence j, E-value(i) is the E-value for sequence i and $P(f_a|f_j)$ is the FAM score [Equation (6)]. Using the raw score distributions for each GO term in a benchmarking dataset, PFP computes a P-value and the expected accuracy from the P-value for the predicted GO terms. Here, we use predicted GO terms with an expected accuracy of 0.8 or higher.

2.7 Sequence database with GO annotations

PSI-BLAST searches needed for ESG, PFP and Top-PSIBLAST method are run against UniProt sequence database (Bairoch *et al.*, 2005). Then, GO terms associated to gene IDs are taken from the GO Consortium database (Ashburner *et al.*, 2000) version go_200804. Thus, the same sequence database and the same setting are used for genes of any organisms. Along with the annotation, we also use the evidence codes in the analysis shown in Figure 4. In the evidence code, Inferred from Electronic Annotations (IEAs) are those obtained from some computational method without any experimental evidences.

2.8 Prediction accuracy

We evaluate prediction accuracy by the modified funsim semantic similarity measure (Hawkins *et al.*, 2009), and also by precision and recall. The funsim score previously proposed (Schlicker *et al.*, 2006) is modified to include evaluation of GO terms in the cellular component category. The funsim score is explained below.

2.9 The funsim semantic similarity score

The funsim score uses the frequency of GO terms, freq(c), in the database as its basis [Equation (11)]. annot(c) is the number of times GO term c occurs in the database to annotate different sequences and children(c) is the list of child nodes of term c in the GO hierarchy. Probability for a GO term c is given by Equation (12).

$$\mathrm{freq}(c)=\mathrm{annot}(c)+\sum_{h\in\mathrm{children}(c)}\mathrm{freq}(h) \qquad (11)$$

$$p(c)=\frac{\mathrm{freq}(c)}{\mathrm{freq(root)}} \qquad (12)$$

The similarity of two GO terms is computed by Equation (13), where $S(c_1,c_2)$ is the set of common ancestors of terms c_1 and c_2 in GO hierarchy. The multiplier $(1-p(c))$ is to give less importance to a frequently occurring term.

$$\mathrm{sim}(c_1,c_2)=\max_{c\in S(c_1,c_2)}\left(\frac{2\log p(c)\cdot(1-p(c))}{\log p(c_1)+\log p(c_2)}\right) \qquad (13)$$

Now consider two sets of GO terms, GO_A and GO_B, to be compared (e.g. GO_A is predicted GO terms and GO_B is the correct terms). For each category we compute $\mathrm{sim}(c_1,c_2)$ separately, comparing pairs of terms from GO_A and GO_B, to form the matrix (S_{ij}) [Equation (14)]. A_{km} and B_{kn} is the number of terms in the set GO_A and GO_B in the category k, respectively. Then, the maximum of the average score of row max or column max is selected as the *categoryScore* for the category, k [Equation (15)].

$$(S_{ij})=\mathrm{sim}(\mathrm{GO}_{Aki},\mathrm{GO}_{Bkj}), \qquad (14)$$

for category $k, i=1$ to A_{km} and $j=1$ to B_{kn}

$$categoryScore_k=\max\begin{pmatrix}\frac{1}{\mathrm{A}_{km}}\sum_{\mathrm{i}=1}^{\mathrm{A}_{km}}\max\limits_{j=1\ldots\mathrm{B}_{kn}}(S_{ij}), \\ \frac{1}{\mathrm{B}_{kn}}\sum_{i=1}^{\mathrm{B}_{kn}}\max\limits_{j=1\ldots\mathrm{A}_{km}}(S_{ji})\end{pmatrix} \qquad (15)$$

Finally, the funsim score between two sets of GO terms is defined as follows. Unlike Schlicker's method all the three GO categories are compared. The maximum score in each category is set to 1.

$$\mathrm{funsim}(\mathrm{GO_A},\mathrm{GO_B})=\frac{1}{3}\begin{pmatrix}\left(\frac{BPScore}{\max(BPScore)}\right)^2+ \\ \left(\frac{MFScore}{\max(MFScore)}\right)^2+\left(\frac{CCScore}{\max(CCScore)}\right)^2\end{pmatrix} \qquad (16)$$

Since funsim score matches each term from one set to any one term from other set and takes average scores from such matches, it penalizes over prediction.

2.10 Precision and recall computations

Along with funsim score we also use precision and recall for evaluation. Precision is defined as TP/(TP + FP) and recall is defined as TP/(TP + FN), where TP and FP denote true and false positive, respectively, and FN denotes false negative. For computing precision and recall, we consider exact match between predicted terms and the actual annotations of the sequence.

2.11 Benchmark dataset

We compare performance of function prediction accuracy on a benchmark dataset of 200 protein sequences each from the following 12 different species in the GO database: *Arabidopsis thaliana*, *Bos taurus*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Danio rerio*, *Escherichia coli*, *Gallus gallus*, *Homo sapiens*, *Mus musculus*, *Pseudomonas aeruginosa*, *Rattus norvegicus* and *Saccharomyces cerevisiae*. Thus, there are in total of 2400 sequences. In each genome, 200 proteins are randomly selected from those which have at least one sequence hit with GO annotations within the E-value of 0.01 by PSI-BLAST against the GO database, in order for the Top PSI-BLAST method to be able to provide function prediction to all the sequences. This dataset is available at our web site (http://dragon.bio.purdue.edu/ESG/testdata/).

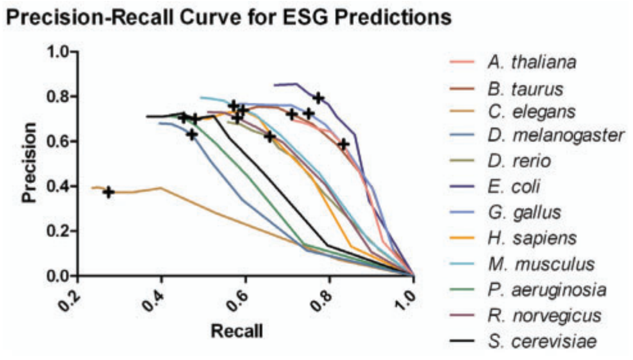

Fig. 2. Precision–recall curve of ESG predictions. Crosses show the data points with the probability cutoff of 0.35.

3 RESULTS

3.1 Prediction accuracy of ESG

We have run ESG with two levels and at each level we have selected the top $N = 50$ hits obtained by PSI-BLAST search. The stage weight v is set to 0.5. Unless specified, ESG only uses direct probability, but without FAM [i.e. Equation (5)]. Figure 2 shows the precision–recall curve for ESG predictions. With the probability cutoff of 0.35, fairly optimum precision of 68% and recall of 58% are obtained. For higher probability cutoffs above 0.35, there was no strong increase in the precision but the recall was falling sharply. At higher cutoffs, the number of terms predicted was less as compared with the false negatives in the prediction.

Figure 3A compares performance of ESG with PFP and the Top PSI-BLAST method. We used two probability cutoff values, 0.35 and 0.15 for ESG. Among the methods compared, ESG shows the best funsim score of 0.697 with the cutoff of 0.15. With the probability cutoff of 0.35, the average funsim score decreased to 0.683, but it is still better than the performance of PFP and the Top PSI-BLAST. The average funsim score of PFP and the Top PSI-BLAST is 0.574 and 0.452, respectively. According to Schlicker *et al*. (2006), the funsim scores in the range of 0.5–0.7 indicate that GO terms are functionally related. It is very interesting that ESG achieves better performance than PFP without using FAM (PFP is using FAM). We also compared the three methods by the funsim score with only using MF and biological process (BP) categories, since GO terms in cellular component (CC) category is relatively less developed (the total number of terms in MF, BP and CC is 8827, 15131 and 2182, respectively). ESG again performs the best but the probability cutoff of 0.35 for ESG now shows a marginally better performance than 0.15: the average funsim score of ESG (with cutoff 0.35), ESG (0.15), PFP and the Top PSI-BLAST is 0.718, 0.717, 0.601 and 0.465, respectively. The graph is available in the Supplementary Material. This result clearly shows that extensive use of information retrieved from PSI-BLAST sequence search result (by considering very weak hits: ESG and PFP; by extending search space by iterative database search: ESG; considering GO term association by FAM: PFP) significantly contributes in improving function prediction accuracy.

In Figure 3B, we further compare the three methods with GOPET, a SVM-based GO prediction method. As its available program only computes GO terms in the MF category, this comparison is made in MF. ESG with the cutoff of 0.15 shows the best performance (0.706), and GOPET and ESG with cutoff of 0.35 follow with the score of 0.705 and 0.690, respectively. Moreover, note that we compared PFP with GOtcha and InterProScan (Mulder and Apweiler, 2007) in our previous work (Hawkins *et al*., 2009) and showed that PFP performs significantly better than them (Figs 7 and 8 in the paper).

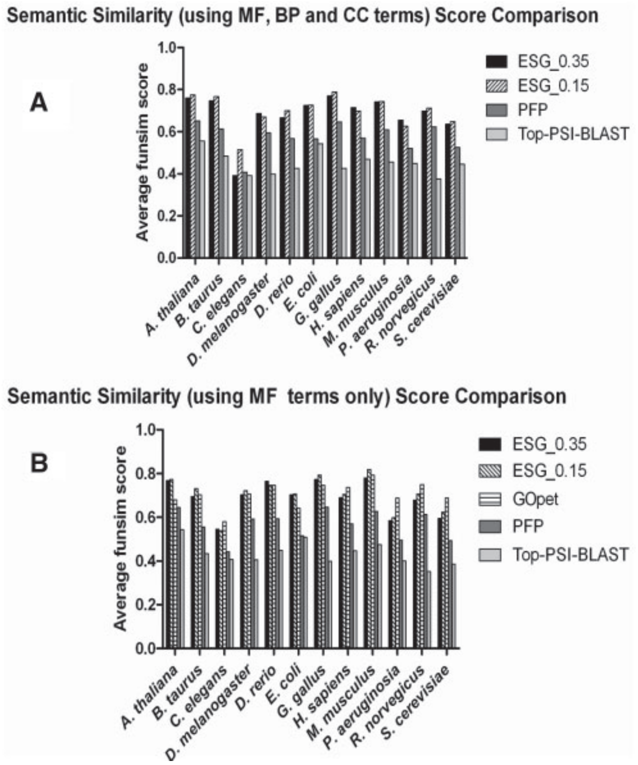

Fig. 3. Prediction accuracy measured by the funsim score. (**A**) Average funsim scores for ESG, PFP and Top PSI-BLAST for benchmarking dataset across 12 species. (**B**), Comparison with GOPET with funsim score using MF terms only.

Next, we examine ESG's performance on two sets of sequences, one set with IEA and one without IEA(Fig. 4). The 200 sequences for each organism are randomly selected for both with/without IEA. This test is performed because predicting electronic annotation (i.e. IEA) by another electronic method, ESG, may seem tautological and thus seem trivial. It is shown that the funsim score of sequences with and without IEA are almost the same, around 0.68. Therefore, good performance of ESG (Fig. 4) is not biased by sequences with IEA.

At last in this section, we examine the effect of different levels of iterations. We ran iterations up to 3 with $N = 10$ on the 200 *E.coli* genes without IEA. The resulting funsim score was 0.693, 0.696 and 0.693, for 1, 2 and 3 levels, respectively. This result indicates that ESG with two levels is quite optimal and further iterations probably may not improve the accuracy.

3.2 Comparison of precision and recall values

In addition to the funsim score, we also computed precision and recall using exact matches with the actual database annotations (Fig. 5). ESG shows significantly higher average precision of 0.67 as compared with those of PFP (0.10) and Top PSI-BLAST (0.10). In

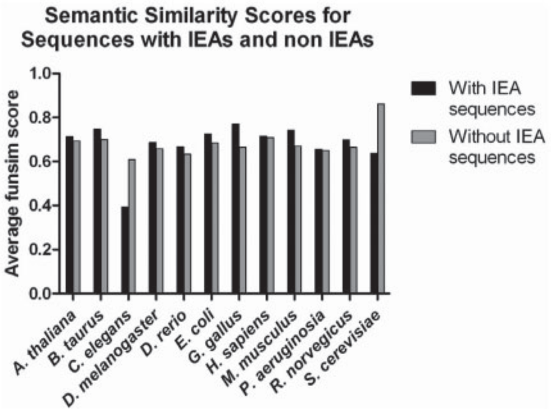

Fig. 4. The funsim score for benchmarking set with and without IEAs across the 12 different organisms. Probability cutoff of 0.35 is used.

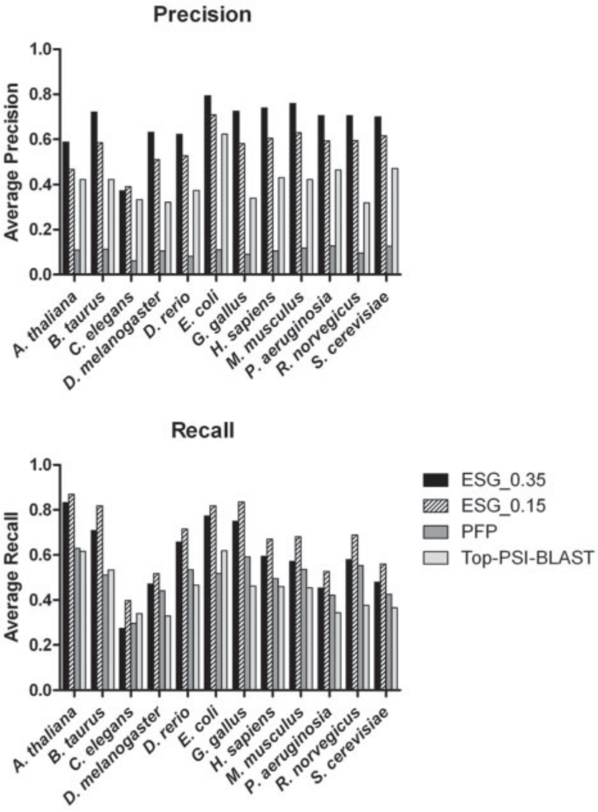

Fig. 5. Precision and recall values for ESG, PFP and Top PSI-BLAST.

Table 1. Prediction accuracy comparisons on the *E.coli* dataset

Method	Precision	Recall	funsim	Average no. of terms predicted
ESG with FAM	0.566	0.810	0.711	12.3
ESG without FAM	0.794	0.773	0.726	7.3
PFP	0.111	0.518	0.565	56.5
Top PSI-BLAST	0.112	0.104	0.204	8.2

terms of recall, ESG again shows overall the best performance over the other two methods (PFP shows a slightly better performance on the *C.elegans* dataset). However, the margin of ESG over PFP is shrunken. The average recall for ESG, PFP and Top PSI-BLAST is 0.60, 0.50 and 0.12, respectively. We find that the key advantage of ESG is the much smaller number of GO terms predicted by ESG (on average 7) as compared with PFP (on average 56), giving fewer false positives and maintaining high precision value. Top PSI-BLAST gives 11 predicted GO terms on average, while the database annotations have on an average of 10 terms for each protein. Narrowing to the minimum set of correct prediction by ESG eliminates over predictions and is a main reason for the high funsim scores shown in Figure 3.

3.3 Prediction accuracy for ESG with FAM

Next, we examine the effect of FAM on function prediction of ESG. Recall that FAM is not incorporated to ESG, but has been incorporated to PFP in the results so far shown. The testing ESG with FAM is performed on the 200 *E.coli* genes, because the computational time increases significantly when FAM is integrated in ESG. For example, in a typical case of an ESG run without FAM takes about 15–20 min, while it increases to 2 h or more when FAM is used. FAM is added in following Equations (7–9).

As shown in Table 1, when FAM is incorporated, precision of ESG falls from 0.794 to 0.566. But at the same time recall improves from 0.773 to 0.810. Reflecting the drop of precision, the funsim score is slightly deteriorated by using FAM from 0.726 to 0.711. FAM brings in additional information of GO term association mined from the sequence database. Therefore, generally using FAM results in predicting more GO terms. The average number of GO terms predicted with FAM by ESG is 12.3 as compared with 7.3 when predicting without FAM. This fact accounts for the decrease in precision and the increase in recall by incorporating FAM. The average number of predicted GO terms by ESG without FAM is same as that of Top PSI-BLAST. In contrast, PFP makes significantly larger number of predictions (i.e. 56.5 on average) using the threshold expected accuracy value of 0.8, which result in a low precision of 0.111.

3.4 Protein with multiple functional domains

In analyzing prediction results of ESG closely, we found that ESG's iterative search scheme is often effective in capturing multiple functional domains of a protein. When a query sequence has multiple domains, each sequence hit of the first level of PSI-BLAST search shares at least one common domain with the query. Then, the second-level search starting from each first-level hit can boost the score of a domain by identifying more sequences that share the domain. Such examples of proteins where conventional BLAST search assigns false positive annotations due to their multiple domains are shown in Figure 6 (Song *et al.*, 2008). PDGFRB and PRKG1 have a statistically significant alignment with an *E*-value of 2e-13. They have protein kinase domain in common, while Ig-like C2-type domains are unique to PDGFRB and cyclic nucleotide binding domains are unique to PRKG1. Now PDGFRB and NCAM2 have a significant alignment with an *E*-value of 1e-8, having Ig-like C2-type domains in common. Although the two proteins, PRKG1

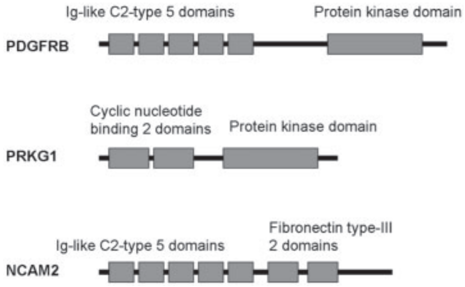
Ig-like C2-type 5 domains
Protein kinase domain
PDGFRB
Cyclic nucleotide binding 2 domains
Protein kinase domain
PRKG1
Ig-like C2-type 5 domains
Fibronectin type-III 2 domains
NCAM2

Fig. 6. Domain structure of PDGFRB, PRKG1 and NCAM2 (figure is not drawn to the exact scale of the proteins).

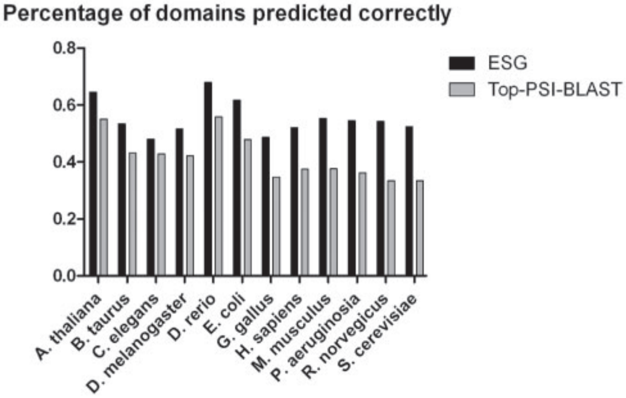

Fig. 7. Domain assignment accuracy.

and NCAM2, have a significant E-value with PDGFRB, their functionalities differ due to the presence of other unique domains to each protein: PDGFRB is β-type platelet-derived growth factor receptor and PRKG1 is cGMP-dependent protein kinase 1, both of which are involved in protein amino acid phosphorylation. NCAM2 with Fibronectin type III domains is unique in the sense that it involves in neural cell adhesion. In this example, the E-value, the alignment coverage, the number of shared domains, length of alignments, etc., are not sufficient for correctly establishing the fact that NCAM2 is involved in different biological process though it shares the common domain with PDGFRB (Song *et al*., 2008).

Interestingly, ESG correctly assigns function to these proteins. ESG predicts correct function for NCAM2 as involved in 'cell adhesion' with probability 0.9. The 'protein amino acid phosphorylation' is predicted with a low probability of 0.25, which is below the cutoff used. ESG predicted nine terms, while PFP predicts 33 terms that contain the term 'protein kinase activity' with an expected accuracy of 0.979, even though kinase domain is not present in NCAM2. For protein PRKG1 both ESG and PFP predict the correct terms indicating that the protein is involved in cyclic nucleotide-dependent kinase activity, with ESG predicting total seven terms and PFP predicting 48 terms.

In Figure 7, we further investigated domain prediction accuracy by ESG compared with Top PSI-BLAST on the entire benchmark dataset. Pfam domains are assigned to sequences in the benchmark dataset by referring to the Uniprot database, which gives a list of Pfam domains for a given gene sequence. Then, a set of predicted GO terms for a sequence are mapped to corresponding Pfam domains using a GO to Pfam correspondence table available at the GO database. Assuming that the set of Pfam domains are predicted by ESG (or Top PSI-BLAST), these predicted domains are compared with the domains assigned to each sequence to compute the recall. Figure 7 shows that ESG captures more correct domains as compared with Top PSI-BLAST. Note that it is very difficult to compute precision of the methods in terms of Pfam domains because Pfam domains and GO terms have many-to-many relationship.

4 DISCUSSION

In this article, we described a novel function prediction algorithm, ESG, which extracts function annotation from the sequence similarity space that is extended by the iterative database search. To clarify characteristics of ESG, we compared performance of ESG with two other methods, the Top PSI-BLAST method and PFP. Top PSI-BLAST represents the typical homology search used in large-scale genome annotation where annotation is transferred to the query protein from the most significant hit in a search. PFP is a previously developed method by our group that was proved successful in automated function prediction. On the benchmarking dataset of 2400 sequences taken from 12 organisms, ESG consistently showed the best funsim score among the three methods. Top PSI-BLAST performed significantly worse than ESG and PFP.

At this juncture, we briefly discuss differences in design and concept of PFP and ESG. PFP extracts relevant GO annotation even from sequences with insignificant E-values by summing up scores reflecting the E-value of sequences. FAM further expands PFP's sensitivity to capture related GO terms. ESG, in contrast, limits GO terms to predict, by examining consistency of appearance of the GO terms in a number of searches in the vicinity of the query sequence on the sequence similarity space. Thus, PFP is designed to enhance sensitivity, while ESG has better precision for GO term prediction. Another difference is that ESG assigns probability to predicted GO terms using a rigorous statistical framework as opposed to PFP, which assigns a custom PFP score to GO terms and computes the P-value from background distribution of the custom PFP score.

Biological implication by the success of ESG and PFP is that there exist functional commonalities among proteins which are not traditionally considered as homologous, and importantly, such common function can be captured by making use of very weakly similar sequences in a database search. To exemplify this statement, ESG search for a query protein, P76216, is examined. P76216 is involved in biological processes GO:0019544 arginine catabolic process to glutamate, GO:0006525 arginine metabolic process and GO:0006950 response to stress. And it has molecular functions GO:0009015 N-succinylarginine dihydrolase activity, GO:0005515 protein binding, and GO:0016787 hydrolase activity. ESG could predict all terms except for GO:0005515 with the probability cutoff of 0.35. Figure 8 illustrates that many sequences with an insignificant E-value hold common annotation to the query and some of them are recaptured by the second-level searches. Out of the 1615 sequences found, 268 have common annotations to the query. Note that this hit rate is far better than random: a randomly selected set of same size from the database contain 112 proteins that had at least one common annotation.

There is a strong need for accurate automatic function prediction methods as the number of sequenced genomes is rapidly increasing.

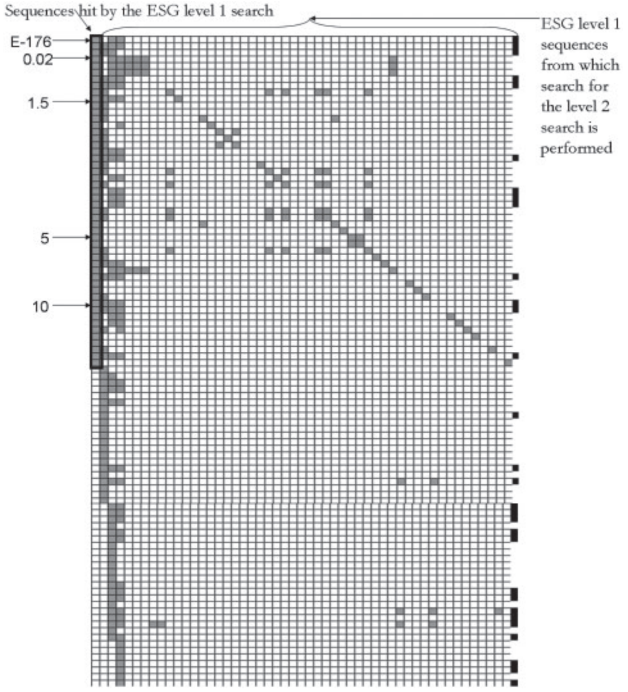

Fig. 8. Heatmap representation of sequence hits by ESG for a query sequence, P76216. The left most column shows sequence hits by the first-level ESG search sorted by the *E*-value. Each row represents one sequence. The top 50 sequences used as queries in the second-level search are surrounded by a thick rectangle. Sequences below the thick rectangle have an *E*-value between 10 and 13. The second-level search results from each of the 50 sequences are visualized in the next 50 columns. In columns of the second-level search, gray boxes indicate that the sequences found in the first-level search reappeared in the second-level search. Black boxes at the right side of each row indicate that the sequence representing the row has annotation common with the query. The figure represents the top part of sequences obtained in the ESG computation.

Various efforts have been made to address this goal including classification of large protein sequence space (Kaplan *et al*., 2005; Loewenstein and Linial, 2008), considering protein structures (Yeats *et al*., 2008), and pathway data (Kanehisa *et al*., 2008). A recent trend is to consider heterogeneous experimental data sources such as microarray and protein–protein interaction. Although such new data can provide additional function information, obviously major sources of function information reside in sequence databases. Thus, sequence-based methods should remain at the center of gene function annotation and it needs to be re-examined with a fresh perspective to investigate the complex relationship between sequence and functional similarity. ESG together with PFP shows a promising future direction with strong evidences that there are still rich sources of functional information in weakly similar sequences, which are previously underestimated.

Funding: National Institutes of Health (GM075004 and GM077905 in parts); National Science Foundation (DMS0604776 and DMS800568 in parts). CP was supported by the Chung-Ang University Research Grants in 2008.

Conflict of Interest: none declared.

REFERENCES

Altschul,S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*., **25**, 3389–3402.

Bairoch,A. *et al.* (2005) The Universal Protein Resource (UniProt). *Nucleic Acids Res*., **33**, D154–D159.

Devos,D. and Valencia,A. (2001) Intrinsic errors in genome annotation. *Trends Genet.*, **17**, 429–431.

Duan,Z.H. *et al.* (2006) The relationship between protein sequences and their gene ontology functions. *BMC Bioinformatics*, **7** (Suppl. 4), S11.

Fitch,W.M. (2000) Homology a personal view on some of the problems. *Trends Genet*., **16**, 227–231.

Friedberg,I. (2006) Automated protein function prediction - the genomic challenge. *Brief Bioinform.*, **7**, 225–242.

Friedberg,I. *et al*. (2006) New avenues in protein function prediction. *Protein Sci.*, **15**, 1527–1529.

Galperin,M.Y. and Koonin,E.V. (1998) Sources of systematic error in functional annotation of genomes: domain rearrangement, non-orthologous gene displacement and operon disruption. *In Silico Biol.*, **1**, 55–67.

Harris,M.A. (2004) The gene ontology (GO) database and informatics resource. *Nucleic Acid Res.*, **32**, D258–D261.

Hawkins,T. and Kihara,D. (2007) Function prediction of uncharacterized proteins. *J. Bioinform. Comput. Biol.*, **5**, 1–30.

Hawkins,T. *et al.* (2006) Enhanced automated function prediction using distantly related sequences and contextual association by PFP. *Protein Sci*., **15**, 1550–1556.

Hawkins,T. *et al.* (2008) New paradigm in protein function prediction for large scale omics analysis. *Mol. Biosyst.*, **4**, 223–231.

Hawkins,T. *et al.* (2009) PFP: automated prediction of gene ontology functional annotations with confidence scores using protein sequence data. *Proteins*, **74**, 556–582.

Jensen,L.J. (2003) Functionality of system components: conservation of protein function in protein feature space. *Genome Res*., **13**, 2444–2449.

John,B. and Sali,A. (2004) Detection of homologous proteins by an intermediate sequence search. *Protein Sci*., **13**, 54–62.

Kanehisa,M. *et al.* (2008) KEGG for linking genomes to life and the environment. *Nucleic Acids Res*., **36**, D480–D484.

Kaplan,N. *et al.* (2005) ProtoNet 4.0: a hierarchical classification of one million protein sequences. *Nucleic Acids Res*., **33**, D216–D218.

Khan,S. *et al.* (2003) GoFigure: automated Gene Ontology annotation. *Bioinformatics*, **19**, 2484–2485.

Loewenstein,Y. and Linial,M. (2008) Connect the dots: exposing hidden protein family connections from the entire sequence tree. *Bioinformatics*, **24**, i193–i199.

Lopez,G. *et al*. (2007) Assessment of predictions submitted for the CASP7 function prediction category. *Proteins*, **69**, 165–174.

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

Mulder,N. and Apweiler,R. (2007) InterPro and InterProScan: tools for protein sequence classification and comparison. *Methods Mol. Biol.*, **396**, 59–70.

Park,J. *et al.* (1997) Intermediate sequences increase the detection of homology between sequences. *J. Mol. Biol.*, **273**, 349–354.

Plewniak,F. *et al.* (2003) PipeAlign: a new toolkit for protein family analysis. *Nucleic Acids Res*., **31**, 3829–3832.

Schlicker,A. *et al.* (2006) A new measure for functional similarity of gene products based on Gene Ontology. *BMC Bioinformatics*, **7**, 302.

Song,N. *et al.* (2008) Sequence similarity network reveals common ancestry of multidomain proteins. PLoS *Comput. Biol*., **4**, e1000063.

Tian,W. and Skolnick,J. (2003) How well is enzyme function conserved as a function of pairwise sequence identity? *J. Mol. Biol.*, **333**, 863–882.

Vinayagam,A. *et al.* (2006) GOPET: a tool for automated predictions of Gene Ontology terms. *BMC Bioinformatics*, **7**, 161.

Yeats,C. *et al.* (2008) Gene3D: comprehensive structural and functional annotation of genomes. *Nucleic Acids Res.*, **36**, D414–D418.

Zehetner,G. (2003) OntoBlast function: from sequence similarities directly to potential functional annotations by ontology terms. *Nucleic Acids Res*., **31**, 3799–3803.