GOTrees: Predicting GO Associations from Protein Domain Composition Using Decision Trees

B. Hayete and J.R. Bienkowska

Pacific Symposium on Biocomputing 10:127-138(2005)

GOTREES: PREDICTING GO ASSOCIATIONS FROM PROTEIN DOMAIN COMPOSITION USING DECISION TREES

BORIS HAYETE[1,2] AND JADWIGA R. BIENKOWSKA[1,3]
[1]*Serono Reproductive Biology Institute,*
One Technology Pl, Rockland MA 02370
[2]*Bioinformatics Program and* [3]*BioMedical Engineering Department,*
Boston University
36 Cummington St. Boston MA 02215
E-mail: jadwiga.bienkowska@serono.com

The Gene Ontology (GO) offers a comprehensive and standardized way to describe a protein's biological role. Proteins are annotated with GO terms based on direct or indirect experimental evidence. Term assignments are also inferred from homology and literature mining. Regardless of the type of evidence used, GO assignments are manually curated or electronic. Unfortunately, manual curation cannot keep pace with the data, available from publications and various large experimental datasets. Automated literature-based annotation methods have been developed in order to speed up the annotation. However, they only apply to proteins that have been experimentally investigated or have close homologs with sufficient and consistent annotation. One of the homology-based electronic methods for GO annotation is provided by the InterPro database. The InterPro2GO/PFAM2GO associates individual protein domains with GO terms and thus can be used to annotate the less studied proteins. However, protein classification via a single functional domain demands stringency to avoid large number of false positives. This work broadens the basic approach. We model proteins via their entire functional domain content and train individual decision tree classifiers for each GO term using known protein assignments. We demonstrate that our approach is sensitive, specific and precise, as well as fairly robust to sparse data. We have found that our method is more sensitive when compared to the InterPro2GO performance and suffers only some precision decrease. In comparison to the InterPro2GO we have improved the sensitivity by 22%, 27% and 50% for Molecular Function, Biological Process and Cellular GO terms respectively.

1 Introduction

With genomic data available in large volume for many organisms, assigning a function to a sequence has become the new challenge for genomics. Various computational methods provide insights into properties of a novel or poorly studied protein. Relevant to this study are the homology-motivated methods that describe the protein function in terms of *functional domains.* Databases such as InterPro (Mulder, Apweiler et al. 2003), PFAM (Bateman, Birney et al. 2002) and others identify functional domains and describe the domain function. Knowledge of a protein's domain content is crucial to understanding the protein’s role. However, proteins often contain multiple domains, some of which may be shared with proteins playing a different role in the cell (Fig. 1). Thus, domain identification cannot become the final point of any annotation. This issue can be addressed by mapping functional domains to Gene Ontology (Ashburner, Ball et al. 2000) terms and by using the Gene Ontology for protein annotation.

The goal of the Gene Ontology Consortium (Ashburner, Ball et al. 2000) is to produce a controlled biological vocabulary that can be applied to all organisms even as the knowledge of gene and protein roles in cells is accumulating and changing. GO provides three structured networks of defined terms to describe gene product attributes: biological process, cellular component and molecular function. GO is one of the controlled vocabularies of the Open Biological Ontologies (OBO.) It is one that is most advanced in its development and will likely serve as a reference for other proposed biological ontologies in the OBO family. The curators of GO annotate a gene's associations with GO terms and the evidence of the association is recorded in the GO database. Contributions to GO come from various public databases and model organism consortia such as the FlyBase (The FlyBase Consortium 2003), Saccharomyces Genome Database (SGD) (Issel-Tarver, Christie et al. 2002), WormBase (Harris, Lee et al. 2003), SwissProt (Boeckmann, Bairoch et al. 2003). Information stored in the GO database is extremely valuable as it is created in a controlled manner and provides a compilation of knowledge about genes from various organisms. The hierarchical nature of the Gene Ontology allows for an elegant representation of both knowledge and uncertainty in understanding of the biological role of a protein. For example, let us imagine a novel protein known to be a kinase whose target remains to be determined. Gene Ontology can store 'kinase activity' as the current known molecular function without requiring further specification. Thus each GO term has parents that represent a less certain annotation than the child. The Gene Ontology offers two properties essential for protein annotation: completeness and breakdown by generalization.

A number of methods have been developed to annotate gene products with GO terms electronically[1]. The electronic methods fall into the categories of:

- Text mining, such as literature mining (Chiang and Yu 2003) and (Raychaudhuri, Chang et al. 2002), and pattern of annotation mining (King, Foulger et al. 2003).
- Analysis of experimental and sequence data:
  - Sequence similarity-motivated methods such as The Institute Of Genomic Research (TIGR) annotations for *T. brucei* (El-Sayed, Ghedin et al. 2003) and *Arabidopsis* (Buell, Joardar et al. 2003; Haas, Delcher et al. 2003; Wortman, Haas et al. 2003), annotations by (Hennig, Groth et al. 2003), the BLAST-based approach of (Khan, Situ et al. 2003), and others.
  - Methods based on protein domains, such as (Schug, Diskin et al. 2002).
  - Methods using gene-expression datasets as in (Lagreid, Hvidsten et al. 2003), (Hvidsten, Komorowski et al. 2001), (Hvidsten, Laegreid et al. 2003).
  - Methods using protein-protein interaction data, such as (Letovsky and Kasif 2003).
- Multi-source and multi-approach (integrative methods):
  - Database-driven EBI GOA (Camon, Magrane et al. 2003) and Mouse Genome Informatics annotations (Hill, Davis et al. 2001).

[1] See http://bio-mirror.asti.dost.gov.ph/biomirror/geneontology/docs/GO.annotation.html for the most current available computational annotations.

- Other multi-source methods, such as (Xie, Wasserman et al. 2002), which use text mining, domain information, sequence homology and other approaches.

Electronic annotation methods can take advantage of mappings between various existing databases and GO[2]. The following types of mappings to GO from several datasets currently exist:

- Keyword/concept: spkw2GO, which maps UniProt (Apweiler 2004) keywords; genprotec2GO, which maps GenProtEC (Riley 1998; Serres and Riley 2000) function names to GO; tigr2GO, mapping TIGR roles (Haft, Selengut et al. 2003) and others.
- Protein family: tigrfam2GO maps TIGR protein families (Haft, Selengut et al. 2003).
- Pathway: metacyc2go maps MetaCyc (Karp, Riley et al. 2002) metabolic processes and functions .
- Domain: InterPro2GO, which maps InterPro (Mulder, Apweiler et al. 2003) entries to GO; PFAM2GO, derived from InterPro2GO, which maps PFAMs, and others.

In some cases, such as in the case of InterPro2GO/PFAM2GO, the map between the PFAM domains and GO terms can be used to associate proteins with GO terms directly.

A number of methods combine various sources of information to predict the GO assignments. For example, a Bayesian method developed by (Troyanskaya, Dolinski et al. 2003) integrates multiple data sources such as protein-protein interactions and gene expression data and creates functional groupings of *S. Cerevisiae* genes. Probabilistic decision trees have been used by (Syed and Yona 2003) to predict protein function based on biochemical properties of proteins coupled with sequence database analysis. (Syed and Yona 2003) noted the possibility of using domain content for function prediction. (Hvidsten, Komorowski et al. 2001) have combined gene expression and ontology data to predict protein function using rough set theory. (Raychaudhuri, Chang et al. 2002) have implemented a text-mining algorithm for GO annotation.

There are about 100,000 genes currently annotated by the GO consortium and only a small fraction of those, about 10,000, are human and mouse genes. Most of the proteins for which there exists an experimentally validated annotation come from simple uni-cellular organisms like bacteria or yeast. Thus for many human and mammalian genes the association with the GO terms has to be predicted by comparison with simpler proteins. Given the complexity of the mammalian protein domain architectures, a comparison with the simpler proteins is not straightforward. In order to address this challenge we propose a method for inferring the GO annotation of a protein from the protein domain composition.

We chose proteins' domain content as a model, and use PFAM domain annotations as our principal data source. Representation of proteins in terms of

[2] The complete, up-to-date listing of mappings to GO can be found at the Gene Ontology Consortium's web site at http://www.geneontology.org/GO.indices.html

their functional domain content creates a multi-dimensional attribute space, where each domain is a dimension. From here on, we will refer to a protein's domain composition as to its domain content or domain vector (in the space of all known domains). We will use these terms interchangeably. The task of GO term annotation becomes that of mapping a multi-dimensional attribute vector to a set of labels (GO terms). Broadly speaking, we restate protein annotation as a classification problem. As documented in the literature, for example, in (Krishnan and Westhead 2003), the decision tree paradigm can work well for this type of problems.

We compare two different representations of proteins by their domain composition, integer and binary one. We evaluate their performance in associating proteins and GO terms. We also compare the performance of our method to GO term association using the InterPro2GO mapping. We demonstrate that using a protein's entire domain vector over a single domain significantly enhances the sensitivity of annotation for all three GO networks at the expense of a relatively small decrease in precision.

2 Systems and Methods

We need to tackle several obstacles that confound the classification problem as formulated in the decision tree context:

- A protein can be associated with many GO terms. Owing to the complex structure of Gene Ontology, the number of "true" labels describing a protein is much larger than the number of GO terms. This is because, in principle, almost any subset of GO terms should be treated as a unique label. The number of such labels is combinatorially large and precludes any classification attempt. Thus GO terms must have individual classifiers.
- The space of attributes for our classification problem is quite large. Currently there are about 7,000 PFAMs identified. Generally, classifiers don't perform well in high-dimensional attribute spaces. Fortunately, most GO terms describe relatively small numbers of proteins. Thus it is feasible to construct for each GO term a training set that is represented in a subspace with much lower dimensionality.
- Our classifier needs to take into account not only the known associations between proteins and GO terms but also, cautiously, the absence of such associations. We can make a generally true assumption that proteins not already annotated with a term ought to stay dissociated from that term for a good reason. In other words, some proteins not associated in GO with a particular GO term may be treated as negative examples in that term's training set. However, due to the incompleteness of data in GO we cannot always take this to be true and therefore we should not accept blindly all such proteins as negative examples.

2.1 Details of Protein Representation

We have chosen PFAM domains as units of a domain content. PFAM domains become the single dimensions of the attribute space, with each protein represented as a sparse vector in that space. We distinguish between binary and integer attribute vectors, as illustrated in Figure 1 and Table 1. In addition to PFAM domain composition, we analyze all proteins from the training and testing sets for the presence of the signal peptide, transmembrane regions (Krogh, Larsson et al. 2001) and coiled coil (Berger, Wilson et al. 1995) regions. These three indicators are treated as extra dimensions in the attribute space.

Naturally, all proteins associated with a GO term constitute a ***true positives*** set. The definition of the negatives is more elaborate. Let us define all proteins associated with a particular GO term *t* as $\{P_t\}$. Let us denote as $\{D_t\}$ the set of all domains belonging to proteins described by that GO term. Let us further denote as $\{D_P\}$ the domains of some protein *P*. If $\{D_t\}$ and $\{D_P\}$ have non-empty intersection and *P* is not assigned to a parent of *t*, then we call *P* a ***true negative***. This definition reflects the fact that *P*, although similar in domain composition to proteins in $\{P_t\}$, was nonetheless annotated by a human expert to fall outside of this GO term, presumably for a biological reason. This definition of true negatives substantially reduces the dimensionality of the space of attributes for each GO term.

For each GO term we have also created a set of 'synthetic' negative protein examples. These 'synthetic' proteins are single domain proteins made of all domains from the true negative examples but only those domains that are not present in the positive set's parent terms. These ***synthetic negatives*** are always assigned binary values for their domain counts.

In addition to synthetic proteins we have also designed one artificial '***supernegative***' protein per GO term which is composed of all domains that are not present in the protein set representing parents of this term. The supernegative served as the reduced representation for the entire set of proteins eliminated from the training set by the selection of true positives and negatives. Dimensionality of supernegatives is much higher than that of regular proteins since $\{D_{supernegative}\} \approx \{D_{data\_set} - D_t\}$.

2.2 Algorithm

Several well-known decision-tree-learning algorithms are available (e.g. CART (Breiman, Friedman et al. 1984)). For this work we have chosen the OC1 decision tree package (Murthy, Kasif et al. 1994). OC1 allows for oblique as well as axis-parallel splits[3] in the space of attributes, which gave us more flexibility in our data representation.

The OC1 classifier was configured to test both axis-parallel and oblique splits in the integer attribute space, and axis-parallel splits only in the binary space. Following some initial experimentation with different goodness criteria supported by OC1, we have selected the twoing rule (Breiman, Friedman et al.

[3] The OC1 is comparable to C4.5 when run in axis-parallel mode (see the OC1 paper and user manual)

1984), although similar (but slightly inferior) results were obtained using information gain during this initial testing. In the oblique mode, the maximal number of iterations was set to 50 and minimal to 20. We have always used the axis-parallel mode for the GO terms that had more than 1000 positive examples in the training set due to the runtime constraints.

2.3 Training and Testing Sets

We have used two different sets of proteins from SwissProt database for training and testing the validity of the decision tree approach.

- Training set: SwissProt proteins from human, mouse and yeast annotated by GO – 6367 proteins, 1756 domains and 3375 GO terms.
- Testing set: SwissProt proteins from fly (*D. Melanogaster)* and worm (*C.Elegans)* annotated by GO – 1024 proteins, 640 domains and 1138 GO terms.

The training set was selected to encompass the eukaryotic evolutionary tree. The testing set was selected to represent organisms that are evolutionarily distant from the training set organisms.

2.4 Benchmark

As a benchmark we used a reference list generated by the InterPro database. InterPro2GO associates the protein domains with GO terms (Camon, Magrane et al. 2003; Mulder, Apweiler et al. 2003). InterPro2GO is the only available approach that uses domain information to predict GO terms. InterPro2GO uses a simple association rule: a protein domain is associated with a GO term if all proteins associated with the term have that domain. We have used the InterPro2GO list to assess the performance of our method.

2.5 Measures of Performance

Each GO term has a unique decision tree trained using the set of examples from the training set. For each term we have a number of true positive associations (TP) predicted by the method, a number of all true associations in the test set (T), a number of positive associations (P) predicted by the method, a number true negative associations predicted (TN), and a total number of negative (non-existing) associations possible (N). We measure the overall performance of a method by Sensitivity=TP/T, Specificity=TN/N and Precision=TP/P where the numbers TP, T, P, TN, N are summed over all predicted GO term associations.

We also can measure the average per term performance of the classifier. Given the limited size of training and testing sets, not all GO terms were trained or tested and in consequence not all tested GO terms have positive examples. Thus to estimate the average term performance over the complete set of GO proteins we also calculate the performance averaged over the GO terms that do have positive examples. We split the set of GO terms into parents, those terms that do have as children more specific terms linked with them and leaves, the terms that are the last nodes in the GO diagram and represent the most specific protein descriptors in GO. We calculate the averages for 'parent' and 'leaf'[4] terms separately to check whether the method performs differently on generic and specific term types and whether it is affected by a small number of positive examples.

3 Results

In the initial testing of the decision tree approach we limited our predictions to the molecular function of the GO ontology. We have tested 12 possible settings of the method that split into two categories:

1. using synthetic negatives and synthetic supernegative examples
2. using binary versus integer domain representation and axis-parallel or oblique attribute space partition.

Results from those tests are presented in Table 2. We completed only one of four tests with the oblique mode in the time limit that we have set for the algorithm runtime (4 days). Performance of this test convinced us that the oblique search mode is not improving the performance. The standard deviations for Precision and Sensitivity are about 1-1.5%. The standard deviations for specificity are less then 0.5%. Results presented in Table 2 show that the use of synthetic negatives improves the precision of the method by about 15%. It is also clear that the binary representation of the protein is sufficient for the best resolution of the GO terms and is also computationally less demanding. The differences in performance of axis-parallel, oblique and binary domain methods are not statistically significant. The use of supernegative alone is detrimental to the performance of the classifier. However, the supernegative in combination with synthetic negatives marginally improves the precision and sensitivity. The time runs for the algorithm are only approximate since the processors were not uniquely reserved for our application.

Given relatively small training set, some GO terms have quite a small number of positive examples. Table 3 shows the average numbers of the positives, negatives, and synthetic negatives (taken from the training set of proteins) for the GO terms that are in the testing set. We need to note that these distributions are not normal. As expected, the average number of positives is much smaller for the leaf terms than the parent terms. However, distributions of sensitivity, specificity and precision do not show strong dependency on the number of positives or the dimensionality of the attribute space (data not shown). To check whether the performance is different for leaf and parent terms we have calculated precision, sensitivity and specificity averaged over terms in each of those two categories. We discuss below results for the Molecular Function branch of the GO graph. For the binary domain representation and axis-parallel method the average performance for leaves (168 leaves with defined testing set positives) was: Precision = 93.8 ± 21.0,

[4] Here we mean the 'leaves' in the sense of 'leaves of the Directed Acyclic Graph', or those terms that are the most specific in their hierarchy and don't have any children. 'Parents' are 'all nodes which are not leaves'.

Sensitivity=88.6±25.3, Specificity=99.99 ± 0.07. The average performance for parents (321 parents with defined testing set positives) is: Precision = 88.0 ± 24.0, Sensitivity = 81.7 ± 25.3, Specificity = 99.6 ± 4.1. This results show that the performance of the classifier is not adversely affected by a small number of positive examples. In fact the leaf terms have on average better performance as we can expect from the more detailed level of description given by the leaf terms. Similar results are obtained for the other two branches of the GO graph.

The comparison of our results to InterPro2GO mappings is shown in Table 4. We evaluated both methods using the same test set of proteins. The decision trees were trained using the binary domain representation with synthetic negatives and supernegative examples. The InterPro2GO mapping used for comparison was derived from the complete set of GO proteins but GO terms absent from our training set of proteins were not considered. The results demonstrated that the Decision Tree approach was more sensitive than the InterPro2GO mapping. The greatest improvement in sensitivity, a 50% increase, was achieved for the cellular component network of the GO ontology. This good result is not surprising since, in addition to PFAM domain information, we have included as attributes the information about signal peptides, transmembrane helices and coiled coil regions which are good predictors of cellular localization. Sensitivity was also substantially improved for two other networks: for the Biological Process the sensitivity was improved by 26.5% and for Molecular Function the sensitivity was improved by 22.6%. The precision of our method is lower than that of InterPro2GO. This, too, is not unexpected since the InterPro2GO mapping uses a very conservative assignment of domains to GO terms. The values of precision and sensitivity averaged over leaves and parents (see above) suggest that the performance over a complete set of proteins characterized in GO should have a better precision than the ones listed in Table 4.

4 Discussion

The results of our initial investigation show that the Decision Tree classification approach is a valid and effective method for assigning GO ontology terms to proteins based on the domain composition. The method tested here favorably compares with the InterPro2GO approach, currently the only available method that analyses proteins by their domain composition. A prototype web server for assignment of GO ontology to proteins has been designed. Figure 2 shows the results of assignment to GO terms based on the small training set described above. We note that our definition of true negatives could introduce some false negative assignments in the training set. This is unavoidable due to incompleteness of curation and the sparseness of experimental evidence. The excessive assignment of training examples to the set of true negatives should increase the count of false negatives, decreasing both sensitivity and precision of the algorithm. This may account for some of classification mistakes (see Fig. 3), however the very good performance of the classifier justifies our definition of true negatives.

We plan to run the domain assignment for all GO terms with reviewed annotations. This set of proteins will be used for a final training of the GO decision trees. Ten–fold cross-validation procedure will be used to estimate the performance of each tree trained for individual GO terms. The performance data from those experiments will be used to estimate the performance of each decision tree as exemplified in Figure 2. The final trees will be trained on the whole set of GO proteins.

5 Tables and Figures

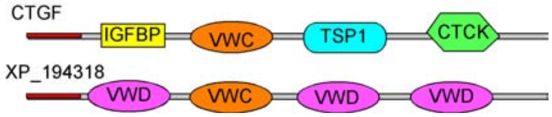
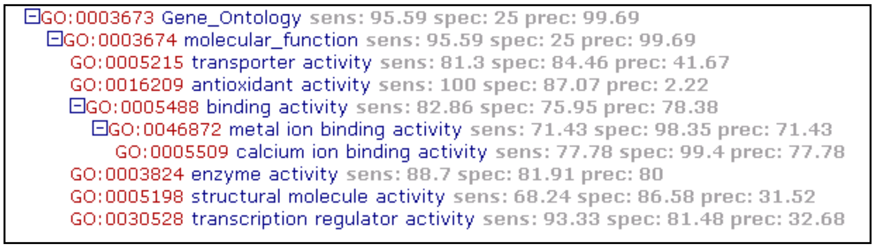


Figure 1. An example of two proteins sharing common domains: Connective Tissue Growth Factor (CTGF) and XP_194318 (similar to kielin). Domains are: Insulin-like Growth Factor Binding Protein (IGFBP), Von Wlillebrand Factor C domain (VWC), Von Willebrand factor D domain (VWD), Trombospondin-domain 1, c-terminal cys-knot domain (CTCK).



```
⊟GO:0003673 Gene_Ontology sens: 95.59 spec: 25 prec: 99.69
  ⊟GO:0003674 molecular_function sens: 95.59 spec: 25 prec: 99.69
     GO:0005215 transporter activity sens: 81.3 spec: 84.46 prec: 41.67
     GO:0016209 antioxidant activity sens: 100 spec: 87.07 prec: 2.22
    ⊟GO:0005488 binding activity sens: 82.86 spec: 75.95 prec: 78.38
      ⊟GO:0046872 metal ion binding activity sens: 71.43 spec: 98.35 prec: 71.43
         GO:0005509 calcium ion binding activity sens: 77.78 spec: 99.4 prec: 77.78
     GO:0003824 enzyme activity sens: 88.7 spec: 81.91 prec: 80
     GO:0005198 structural molecule activity sens: 68.24 spec: 86.58 prec: 31.52
     GO:0030528 transcription regulator activity sens: 93.33 spec: 81.48 prec: 32.68
```

Figure 2. The prototype GO ontology assignment server.

	Binary model	Integer model
CTGF	{1, 1, 1, 1, 0}	{1, 1, 1, 1, 0}
XP_919318	{0, 1, 0, 0, 1}	{0, 2, 0, 0, 2}

Table 1. Example of two different models of domain composition. The proteins show are the same as in Figure 1. The attribute space for these two proteins is D={IGFBP,VWC,TSP1, CTCK, VWD}

	Integer domains Axis-parallel	Integer domains Oblique	Binary domains Axis-parallel

No synthetics no supernegative	prec = 67.3 ± 1.0
	sens = 79.3 ± 1.3
	spec = 99.8 ± 0.1
		Longer than the runtime limit	prec = 65.9 ± 1.0
			sens = 82.7 ± 1.3
			spec = 99.8 ± 0.1

Using synthetics no supernegative	prec = 80.9 ± 1.3
	sens = 78.5 ± 1.3
	spec = 99.9 ± 0.1
		Longer than the runtime limit	prec = 82.2 ± 1.3
			sens = 80.8 ± 1.3
			spec = 99.9 ± 0.1

No synthetics using supernegative	prec = 26.8 ± 0.4
	sens = 80.5 ± 1.3
	spec = 99.1 ± 0.1
		Longer than the runtime limit	prec = 28.0 ± 0.4
			sens = 82.3 ± 1.3
			spec = 99.1 ± 0.1

Using synthetics and supernegative (timed[5])	prec = 81.1 ± 1.3
	sens = 79.9 ± 1.3
	spec = 99.9 ± 0.1
	time : 27 hours
		prec = 81.4 ± 1.3
		sens = 78.8 ± 1.3
		spec = 99.9 ± 0.1
		time : 66 hours
			prec = 82.6 ± 1.3
			sens = 81.0 ± 1.3
			spec = 99.9 ± 0.1
			time : 20 hours

Table 2. Data models' effect on GO term prediction

	all	Leaves	Parents
Positives	28	4	67
Negatives	792	601	1113
synthetics	205	164	275

Table 3. The average numbers of positive, negative and synthetic negative examples for different types of GO terms.

	InterPro2GO	Decision Tree

Biological Process	prec = 91.5 ± 1.8
	sens = 42.6 ± 0.7
	spec = 99.9 ± 0.2
		prec = 82.9 ± 1.3
		sens = 69.1 ± 1.1
		spec = 99.9 ± 0.1

Cellular Component	prec = 99.8 ± 3.4
	sens = 34.8 ± 1.0
	spec = 99.9 ± 0.4
		prec = 85.0 ± 1.8
		sens = 84.9 ± 1.8
		spec = 99.8 ± 0.2

Molecular Function	prec = 98.9 ± 2.0
	sens = 58.4 ± 1.0
	sepc = 99.9 ± 0.2
		prec = 82.6 ± 1.3
		sens = 81.0 ± 1.3
		spec = 99.9 ± 0.1

Table 4. Comparison of performance of the Decision Tree classifier (binary domain representation and synthetic negative examples) to that of InterPro2GO PFAM-based assignments

6 Acknowledgements

We thank S. Kasif for making available to us the OC1 software and S.K Murthy and S. Salzberg for developing the method.

[5] Time required to finish testing and training of Decision Trees for the sets using synthetic negative and supernegative examples using 20 600MHz SGI R3800 CPUs during training stage. The computational times for tests not using synthetic examples were shorter.

7 References

3. Apweiler (2004). "UniProt: the Universal Protein Knowledgebase." Nucleic Acids Res. **32**(in press.).
4. Ashburner, M., C. A. Ball, et al. (2000). "Gene ontology: tool for the unification of biology. The Gene Ontology Consortium." Nat Genet **25**(1): 25-9.
5. Bateman, A., E. Birney, et al. (2002). "The Pfam protein families database." Nucleic Acids Res **30**(1): 276-80.
6. Berger, B., D. B. Wilson, et al. (1995). "Predicting coiled coils by use of pairwise residue correlations." Proc Natl Acad Sci U S A **92**(18): 8259-63.
7. Boeckmann, B., A. Bairoch, et al. (2003). "The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003." Nucleic Acids Res **31**(1): 365-70.
8. Breiman, L., J. Friedman, et al. (1984). Classification and Regression Trees, Wadsworth International Group.
9. Buell, C. R., V. Joardar, et al. (2003). "The complete genome sequence of the Arabidopsis and tomato pathogen Pseudomonas syringae pv. tomato DC3000." Proc Natl Acad Sci U S A **100**(18): 10181-6.
10. Camon, E., M. Magrane, et al. (2003). "The Gene Ontology Annotation (GOA) project: implementation of GO in SWISS-PROT, TrEMBL, and InterPro." Genome Res **13**(4): 662-72.
11. Chiang, J. H. and H. C. Yu (2003). "MeKE: discovering the functions of gene products from biomedical literature via sentence alignment." Bioinformatics **19**(11): 1417-22.
12. El-Sayed, N. M., E. Ghedin, et al. (2003). "The sequence and analysis of Trypanosoma brucei chromosome II." Nucleic Acids Res **31**(16): 4856-63.
13. Haas, B. J., A. L. Delcher, et al. (2003). "Improving the Arabidopsis genome annotation using maximal transcript alignment assemblies." Nucleic Acids Res **31**(19): 5654-66.
14. Haft, D. H., J. D. Selengut, et al. (2003). "The TIGRFAMs database of protein families." Nucleic Acids Res **31**(1): 371-3.
15. Harris, T. W., R. Lee, et al. (2003). "WormBase: a cross-species database for comparative genomics." Nucleic Acids Res **31**(1): 133-7.
16. Hennig, S., D. Groth, et al. (2003). "Automated Gene Ontology annotation for anonymous sequence data." Nucleic Acids Res **31**(13): 3712-5.
17. Hill, D. P., A. P. Davis, et al. (2001). "Program description: Strategies for biological annotation of mammalian systems: " Genomics **74**(1): 121-8.
18. Hvidsten, T. R., J. Komorowski, et al. (2001). "Predicting gene function from gene expressions and ontologies." Pac Symp Biocomput: 299-310.
19. Hvidsten, T. R., A. Laegreid, et al. (2003). "Learning rule-based models of biological process from gene expression time profiles using gene ontology." Bioinformatics **19**(9): 1116-23.

20. Issel-Tarver, L., K. R. Christie, et al. (2002). "Saccharomyces Genome Database." Methods Enzymol **350**: 329-46.
21. Karp, P. D., M. Riley, et al. (2002). "The MetaCyc Database." Nucleic Acids Res **30**(1): 59-61.
22. Khan, S., G. Situ, et al. (2003). "GoFigure: Automated Gene Ontology(TM) annotation." Bioinformatics **19**(18): 2484-5.
23. King, O. D., R. E. Foulger, et al. (2003). "Predicting gene function from patterns of annotation." Genome Res **13**(5): 896-904.
24. Krishnan, V. G. and D. R. Westhead (2003). "A comparative study of machine-learning methods to predict the effects of single nucleotide polymorphisms on protein function." Bioinformatics **19**(17): 2199-209.
25. Krogh, A., B. Larsson, et al. (2001). "Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes." J Mol Biol **305**(3): 567-80.
26. Lagreid, A., T. R. Hvidsten, et al. (2003). "Predicting gene ontology biological process from temporal gene expression patterns." Genome Res **13**(5): 965-79.
27. Letovsky, S. and S. Kasif (2003). "Predicting protein function from protein/protein interaction data: a probabilistic approach." Bioinformatics **19 Suppl 1**: I197-I204.
28. Mulder, N. J., R. Apweiler, et al. (2003). "The InterPro Database" Nucleic Acids Res **31**(1): 315-8.
29. Murthy, S. K., S. Kasif, et al. (1994). "A System for Induction of Oblique Decision Trees." Journal of Artificial Intelligence Research **2**: 1-32.
30. Raychaudhuri, S., J. T. Chang, et al. (2002). "Associating genes with gene ontology codes using a maximum entropy analysis of biomedical literature." Genome Res **12**(1): 203-14.
31. Riley, M. (1998). "Genes and proteins of Escherichia coli K-12." Nucleic Acids Res **26**(1): 54.
32. Schug, J., S. Diskin, et al. (2002). "Predicting gene ontology functions from ProDom and CDD protein domains." Genome Res **12**(4): 648-55.
33. Serres, M. H. and M. Riley (2000). "MultiFun, a multifunctional classification scheme for Escherichia coli K-12 gene products." Microb Comp Genomics **5**(4): 205-22.
34. Syed, U. and G. Yona (2003). Using a mixture of probabilistic decision trees for direct prediction of protein function. Proceedings of the 7th annual international conference on CMB. Berlin, Germany, ACM Press**:** 289--300.
35. The FlyBase Consortium (2003). "The FlyBase database of the Drosophila genome projects" Nucleic Acids Res **31**(1): 172-5.
36. Troyanskaya, O. G., K. Dolinski, et al. (2003). "A Bayesian framework for combining heterogeneous data sources for gene function prediction (in Saccharomyces cerevisiae)." Proc Natl Acad Sci U S A **100**(14): 8348-53.
37. Wortman, J. R., B. J. Haas, et al. (2003). "Annotation of the Arabidopsis genome." Plant Physiol **132**(2): 461-8.
38. Xie, H., A. Wasserman, et al. (2002). "Large-scale protein annotation through gene ontology." Genome Res **12**(5): 785-94.