IMPORTANCE OF SECONDARY STRUCTURE ELEMENTS FOR PREDICTION OF GO ANNOTATIONS

Aslı Filiz[1], *Eser Aygün*[2], *Özlem Keskin*[3] *and Zehra Cataltepe*[2]

[1]Informatics Institute and [2]Computer Engineering Department, Istanbul Technical University
Maslak, 34469, Istanbul, Turkey
[3]Chemical and Biological Engineering, Koc University
Sarıyer, 34450, Istanbul, Turkey
phone: + (90) 212-285-3551, email: filizas@itu.edu.tr
web: bioinfo.ce.itu.edu.tr

ABSTRACT

Predicted or actual protein secondary structure, in addition to amino acid sequence, is often used for fold recognition and function prediction. Different kinds of secondary structure elements could be predicted with different accuracy by different prediction methods and this could affect the fold or function prediction performance. In this study, contribution of amino acid sequence residues belonging to different types of secondary structure elements (H: alpha helix, E: beta sheet, L: loop) for protein function prediction is investigated. Smith-Waterman alignment similarity scores between amino acid sequences belonging to 6 different sets of secondary structure elements, namely, HEL, HE, HL, H, E and L, are computed. Using these alignment scores, protein function prediction is performed. On a function prediction data set, consisting of 27 Gene Ontology (GO) classes and 4498 sequences, it is found out that using the whole amino acid sequence results in the best performance. Using H and L regions together results almost as well performance as HEL. E regions alone are the least significant in function prediction.

1. INTRODUCTION

Predicted secondary structure has been found to be useful for prediction of some protein functions (for example [13]) and fold recognition (for example [21]). Although actual secondary structure helps more than predicted secondary structure does, experimental determination of the secondary structure is a costly process. A lot of different secondary structure prediction methods have been used (see [20] for a review) in the literature.

In this study, importance of different secondary structure elements for protein function prediction is evaluated. In order to do that, first amino acid sequences are partitioned according to the actual secondary structure information. Instead of the whole sequence, portions of the amino acid sequence that belong to different secondary structure regions are used.

The rest of the paper is organized as follows: Section 2 explains the dataset used in this study. Section 3 includes the alignment, classification and evaluation techniques used. Section 4 contains the experimental results. The conclusions drawn from the experiments are in Section 5.

2. DATA SET

To obtain a list of annotated proteins, the Gene Ontology Annotation (GOA) project is used [8]. GOA provides GO assignments for the proteins of human, mouse, rat, arabidopsis, zebra fish, chicken and cow. It also provides a Protein Data Bank (PDB) [22] association file, which contains only the assignments for the proteins present in the PDB database. To be able to fetch sequence and structure information from PDB, we used the PDB association file.

The ontology structure is obtained from Gene Ontology (GO) [5 and 7] database. The three top level GO classes, molecular function, cellular component and biological process, are included in the data set used in this study. In GO hierarchy, a protein may be associated with more than one term if it is known that it has multiple functions. All terms are captured during the labelling process and we generated multi-labelled data with 27 dimensional target vectors for all sequences in the data set. To remove sequence homologs, PDB's scheme is applied. PDB provides several clusterings of proteins generated with CD-HIT or BLASTClust algorithms for different sequence identities. According to the scheme, only the best representative of each cluster is kept for a given clustering. Thereby, potential homologs are removed and non-redundant datasets are obtained. In this study, clusterings generated by BLASTClust for 40% sequence identities are used since amino acid sequences with more than 40% homology tend to have same or very similar functions. To obtain a well-balanced class distribution, classes with less than 100 or more than 550 sequences are eliminated which resulted in a dataset with 27 classes. Table 1 shows the 27 GO classes used.

The amino acid sequences and secondary structures of every protein are downloaded via PDB web service.

Contribution of H, E and L (H: alpha helix, E: beta sheet, L: loop) regions to function could rely on their portion in the sequences, so the average ratio of H, E and L regions' length to the whole protein sequence length is calculated for each function class. Table 1 and Figure 1 show the H, E and L portions in the sequences.

Due to lack of data, function prediction gets harder at deeper levels of GO tree. In Figure 2 (biological process) and Figure 3 (molecular function and cellular component) the locations of the GO functions used in this study are shown in the GO hierarchy.

The dataset obtained from the GO database includes secondary structure in DSSP representation. The secondary structure sequences are converted to HEL representation according to [14] (Table 2).

DSSP	HEL
G, H, I	H
B,E	E
C, S, T	L

Table 2: Conversion from DSSP to HEL

3. EXPERIMENTAL METHODOLOGY

In order to find out the importance of each secondary structure element (H, E, L) for each function, portions of amino acid sequence that has corresponding secondary structure of H or E or L are isolated. Then the amino acid sequences that belong to 6 different secondary structure elements, namely HEL, HE, HL, H, E and L are produced. Figure 4 shows the original amino acid sequence, secondary structure and each of the six amino acid sequences produced for HEL, HE, HL, H, E and L regions. When a secondary structure element is not used, in the amino acid sequence the actual residue is replaced by the "+" symbol. BLOSUM50 substitution matrix is modified to incorporate the "+" symbol.

Pairwise alignment scores between two proteins could be used as input to the pattern recognition algorithm to be used for function prediction and whether the two proteins are in the same class or not are the outputs, as in [9]. Another approach is to use the alignment scores to all available training sequences as input. This is the approach taken in [17] and also in this study. "SVM-pairwise" [17] takes all sequence pairs in the database and aligns them to each other using the Smith-Waterman local alignment algorithm. This is based on the idea that two proteins belonging to the same class can be aligned similarly to a set of proteins containing both positive and negative instances. Alignment scores are then used as the constant-sized feature vector for a protein. For a training set of N sequences, every protein is aligned to all N sequences, including itself, and it has N features. These features are the input to the classification algorithm. Liao and Noble used this method with SVMs and they indicate that this method is not only easy to use, but also superior to similar algorithms (SVM-Fisher [11 and 12], PSI-BLAST [2], SAM [16] and FPS [10]) due to its low complexity and outputs with higher accuracy because of learning from negative examples. Liao and Noble [17] found that SVM-pairwise performs especially well when working with large numbers of protein sequences.

The local alignment algorithm Smith-Waterman is used for computing the pairwise alignment scores. The balign tool [6] developed by Eser Aygün for Bioinformatics Project at ITU is used for computing the alignment scores. balign produces two types of alignment scores, the percent identity and the bit score, which is the sum of the substitution matrix entries for matches minus gap penalties, normalized with respect to the statistical parameters of the scoring system and is therefore comparable between different alignments [18]. In this study, the conservation score is used which is calculated by normalizing the bit score as follows:

$$cons(x,y) = \frac{bitscore(x,y)}{\max(bitscore(x,x), bitscore(y,y))} \qquad (1)$$

where cons(x,y) is the conservation score of sequences x and y and bitscore(x,y) is the bitscore of sequences x and y.

Two different classification algorithms which are variants of nearest neighbor (NN) classification algorithm [1] is used. 1NN classification is preferred due to its effectiveness and low time-complexity [15]. The first algorithm is 1NN. 1NN is a special case of kNN where k=1. According to a certain distance measure, the distance between a test instance and each train instance (including both positive and negative instances) is computed and the test instance is classified to be in the class of the closest train instance. In this study, the Euclidean distance measure is used. The second classification algorithm is called thresholded nearest neighbor (tNN), which works similar to 1NN. Considering that the negative examples are not proven negatives in all cases (it is possible that these do show a specific function, but it is experimentally not shown yet), this algorithm deals with positive examples only. Let s_i be the ith sequence in the database and C_j be the jth class. D_{ij} is defined as the maximum pairwise Smith-Waterman score of s_i and all the sequences in class C_j. If $D_{ij} \geq$ threshold, then s_i is predicted to be in class C_j. Support Vector Machine (SVM) classifier is not used, because its performance on a smaller data set was found to be very close to that of 1NN and running SVM on our data set takes a lot of time.

Because of the multi-labeled character of the dataset, one-against-all classification is used. Each class is to be predicted independently from other classes, so for each of the 27 classes, data is partitioned into two classes with sequences in class C_i and sequences not in C_i.

Evaluation of each method is done via ten-fold cross validation where each partition inherits the class distribution in the original (not partitioned) data set. For evaluating the results, the ROC curves [1] for each class are drawn and area under ROC curve (AUC) is calculated for all ten folds. For testing the tNN, the threshold is initially set to the minimum D_{ij} in the test set and moved to the maximum in equally-sized steps. Since AUC is used to determine the classification performance, no certain threshold is needed for tNN. However, if accuracy would be used for performance, a certain threshold is necessary. The best accuracies obtained by computing the break-even point, which is point where recall and precision values are equal to each other [19], are shown in Table 3.

4. RESULTS

Function prediction results for 1NN and tNN classifiers are shown in Table 3.

For 1NN classification, the HEL dataset has the best performance (mean AUC: 0.90) for all molecular functions except for class 14 (hormone activity) where all classifiers have very close and high performances. The mean AUC of HL is 0.86 which is very close to HEL and it is followed by H (mean AUC: 0.79) and L (mean AUC: 0.77). E (mean AUC: 0.74) and HE (mean AUC: 0.64) performed generally worse than other secondary structure regions. The higher performance of H and L regions is to be expected since their ratio in the data set is higher than E regions (see Figure 1). The fact that HL performs better than both H and L alone, shows that adding L regions to H regions results in a better prediction. But the facts that the AUC values of E and HE are very close to each other and that both are lower than H alone indicate that using E regions introduce noise and reduces classification performance. This also explains the peak at class 14 since the portions of E regions in this class is only 4,75%, ca. one third of the closest E regions' portion of other classes, which makes the sequences in this class far less vulnerable to the noise introduced by E regions.

tNN has the best performance for HEL classification with mean AUC 0.81, followed by HL classification (mean AUC: 0.77) as by 1NN. Class 14 is again an outstanding point with best performance for each classifier. Different from 1NN, the AUC values for H, E and L are very close to each other, mean AUCs 0.66, 0.67 and 0.67 respectively; but HE performed better than these three classifiers with mean AUC 0.73. Another distinguishing point is the very low AUC values for class 27 (catalytic activity) for all classifiers which is not the case by 1NN except for HE classification. The noise effect of E regions stated by 1NN classification is not seen by tNN. The performance of HL classification being better than HE classification is explained by the L portion in the data set which is greater than the E portion. Generally, the AUC values obtained for tNN classification is lower than for 1NN. Since tNN does not use the negative examples, its lower prediction performance is not surprising as learning from negative examples enhances the prediction performance [17].

5. CONCLUSIONS

In this study, it is found out that using the whole amino acid sequences, as opposed to portions belonging to different secondary structure elements, results in the best function prediction performance. Using HL regions together results in almost as good performance as the whole sequence. On the other hand, E regions are the least significant in function prediction. When learning only from positive examples (tNN), HE follows the performance of HL and the distribution of H, E and L does not play a significant role. However, using kNN algorithm which takes into account both positive and negative examples produces better prediction results.

ACKNOWLEDGEMENTS

Filiz was supported by a program 2228 Tubitak scholarship. Cataltepe, Keskin and Aygun were supported by Tubitak EEEAG research project 10E164.

REFERENCES

[1] E. Alpaydın, *Introduction to Machine Learning*, The MIT Press, 2004.

[2] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller and D. J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," *Nucleic Acids Research*, vol. 25, pp. 3389-3402, 1997.

[3] M. Ashburner, "On the representation of gene function in genetic databases," *Proceedings of the Intelligent Systems for Molecular Biology*, vol. 6, 1998.

[4] M. Ashburner, C. Ball, J. Blake, D. Botstein, H. Butler., J. Cherry, A. Davis, K. Dolinski, S. Dwight and J. Eppig, "Gene Ontology: tool for the unification of biology," *Nat. Genet.*, vol. 25, pp. 25–29, 2000.

[5] Z. Aydin, H. Erdogan and Y. Altunbasak, "Protein Fold Recognition using Residue-Based Alignments of Sequence and Secondary Structure," *Acoustic, Speech and Signal Processing 2007*, Vol.1, 15-20 Apr. 2007, pp. I-349-I-352.

[6] E. Aygün and Z. Cataltepe, "balign," in preparation, 2008.

[7] C. Bystroff, V. Thorsson, and D. Baker "HMMSTR: a hidden markov model for local sequence structure correlations in proteins," *Journal of Molecular Biology*, Vol. 301, issue 1, pp. 173–190, Aug. 2000.

[8] E. Camon, M. Magrane, D. Barrell, V. Lee, E. Dimmer, J. Maslen, D. Binns, N. Harte, R. Lopez and R. Apweiler, "The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology," *Nucleic Acids Research*, vol. 32, no. 1, pp. D262-D266, 2004.

[9] J. Cheng and P. Baldi, "A machine learning information retrieval approach to protein fold recognition," *Bioinformatics*, vol. 22, no. 12, pp. 1456–1463, 2006.

[10] N. Grundy, "Family-based homology detection via pairwise sequence comparison," *Proc. 2nd Ann. Int. Conf. Computational Molecular Biology*, 1998, pp. 94–100.

[11] T. Jaakkola, M. Diekhans and D. Haussler, "Using the Fisher kernel method to detect remote protein homologies," *Proc. 7th Int. Conf. Intelligent Systems for Molecular Biology*, 1999, pp. 149–158.

[12] T. Jaakkola, M. Diekhans and D. Haussler, "A discriminative framework for detecting remote protein homologies," *Journal of Computational Biology*, vol. 7, no. 1–2, pp. 95–114, 2000.

[13] L. J. Jensen, R. Gupta, H.-H. Starfeldt and S. Brunak, "Prediction of human protein function according to Gene

Ontology categories," *Bioinformatics,* Vol. 19 no. 5, pp 635–642, 2003.

[14] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, no. 12, pp. 2577-637, 1983.

[15] A. Kocsor, A. Kertész-Farkas, L. Kajan and S. Pongor, "Application of compression-based distance measures to protein sequence classification: a methodological study," *Bioinformatics*, vol. 22, no. 4, pp. 407-412, 2005.

[16] A. Krogh, M. Brown, I. Mian, K. Sjolander and D. Haussler, "Hidden Markov models in computational biology: Applications to protein modelling," *Journal of Molecular Biology,* vol. 235, pp. 1501–1531, 1994.

[17] L. Liao and W. S. Noble, "Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships," *Journal of Comp. Biology*, vol. 10, no. 6, pp. 857-868, 2003.

[18] NCBI Glossary, http://www.ncbi.nlm.nih.gov/ Education/BLASTinfo/glossary2.html , 2004.

[19] A. Passerini, M. Punta, A. Ceroni, B. Rost and P. Frasconi, "Identifying cysteines and histidines in transition-metal-binding sites using support vector machines and neural networks", *Proteins: Structure, Function, and Bioinformatics*, vol. 65, pp. 305–316, 2006.

[20] B. Rost, "Review: Protein Secondary Structure Prediction Continues to Rise," *Journal of Structural Biology,* vol. 134, issues 2-3, pp. 204-218, 2001.

[21] A. Wallqvist, Y. Fukunishi, L. R. Murphy, A. Fadel and R. M. Levy, "Iterative sequence/secondary structure search for protein homologs: comparison with amino acid sequence alignments and application to fold recognition in genome databases," *Bioinformatics*, vol. 16 no 11, pp. 988-1002, 2000.

[22] J. Westbrook , Z. Feng , L. Chen, H. Yang and H. M. Berman, "The Protein Data Bank and structural genomics," *Nucleic Acids Research,* 31: 489-491, 2003.

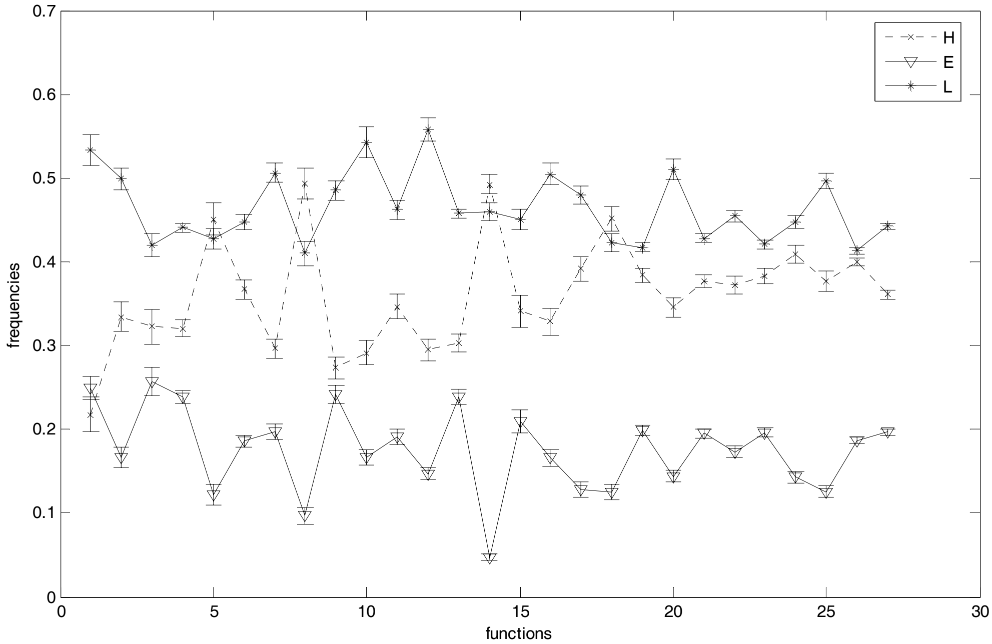

Figure 1: Frequencies of H, E and L regions in the dataset

Original sequence	QYKEVNETKWKMMDPILTTSVPVYSLKVDKEYEVRVRSKQRNSGN
Secondary structure	HHHHHEEEEELLEHLLEEEEEELLLLLLLLLLLLLLLHHHEEEEL
HEL	QYKEVNETKWKMMDPILTTSVPVYSLKVDKEYEVRVRSKQRNSGN
HE	QYKEVNETKW++MD++LTTSVP+++++++++++++++SKQRNSG+
HL	QYKEV+++++KM+DPI++++++VYSLKVDKEYEVRVRSKQ++++N
H	QYKEV++++++++D+++++++++++++++++++++++SKQ+++++
E	+++++NETKW++M+++LTTSVP++++++++++++++++++RNSG+
L	++++++++++KM++PI++++++VYSLKVDKEYEVRVR+++++++N

Figure 4: Secondary structure filtering

Class No	GO ID	Class name	Size	H %	E %	L %
1	0009405	pathogenesis	103	21.78 ± 0.2	24.88 ± 0.14	53.34 ± 0.19
2	0009055	electron carrier activity	105	33.44 ± 0.18	16.66 ± 0.12	49.89 ± 0.13
3	0006810	transport	107	32.27 ± 0.17	25.75 ± 0.18	41.98 ± 0.14
4	0016787	hydrolase activity	117	32.06 ± 0.11	23.87 ± 0.08	44.07 ± 0.06
5	0005506	iron ion binding	118	45.06 ± 0.21	12.18 ± 0.13	42.76 ± 0.13
6	0000166	nucleotide binding	132	36.69 ± 0.13	18.59 ± 0.08	44.72 ± 0.11
7	0003676	nucleic acid binding	137	29.64 ± 0.13	19.76 ± 0.11	50.60 ± 0.14
8	0003700	transcription factor activity	137	49.32 ± 0.22	9.68 ± 0.11	41.00 ± 0.16
9	0006508	proteolysis	148	27.34 ± 0.16	24.11 ± 0.13	48.55 ± 0.14
10	0006412	translation	150	29.15 ± 0.19	16.61 ± 0.11	54.25 ± 0.22
11	0003723	RNA binding	155	34.62 ± 0.18	19.16 ± 0.12	46.21 ± 0.14
12	0008270	zinc ion binding	170	29.50 ± 0.17	14.73 ± 0.10	55.78 ± 0.18
13	0005975	carbohydrate metabolic process	173	30.46 ± 0.15	23.88 ± 0.12	45.77 ± 0.08
14	0005179	hormone activity	177	49.25 ± 0.15	4.75 ± 0.06	46.00 ± 0.15
15	0016020	membrane	202	34.10 ± 0.28	20.92 ± 0.20	44.98 ± 0.17
16	0005515	protein binding	210	32.87 ± 0.24	16.64 ± 0.14	50.49 ± 0.18
17	0005634	nucleus	214	39.18 ± 0.22	12.79 ± 0.13	48.02 ± 0.16
18	0006355	regulation of transcription, DNA-dependent	221	45.14 ± 0.22	12.53 ± 0.13	42.33 ± 0.16
19	0005737	cytoplasm	232	38.41 ± 0.13	19.85 ± 0.10	41.74 ± 0.07
20	0005622	intracellular	278	34.59 ± 0.20	14.37 ± 0.11	51.05 ± 0.20
21	0005524	ATP binding	288	37.72 ± 0.13	19.50 ± 0.09	42.78 ± 0.09
22	0006118	electron transport	297	37.24 ± 0.18	17.30 ± 0.12	45.46 ± 0.12
23	0016491	oxidoreductase activity	300	38.30 ± 0.15	19.63 ± 0.10	42.07 ± 0.09
24	0003677	DNA binding	329	40.92 ± 0.19	14.30 ± 0.13	44.78 ± 0.14
25	0005576	extracellular region	354	37.74 ± 0.23	12.57 ± 0.14	49.69 ± 0.17
26	0008152	metabolic process	361	39.96 ± 0.09	18.73 ± 0.07	41.31 ± 0.06
27	0003824	catalytic activity	522	36.10 ± 0.13	19.67 ± 0.10	44.23 ± 0.10
Total			4498			

Table 1: Gene Ontology class distributions in the data set used and the H, E and L ratios.

Class No	GO ID	HEL	HE	HL	H	E	L	Class No	GO ID	HEL	HE	HL	H	E	L
1	9405	0.96	0.98	0.97	0.95	0.95	0.96	**15**	16020	0.95	0.94	0.95	0.94	0.92	0.93
2	9055	0.98	0.97	0.98	0.92	0.96	0.97	**16**	5515	0.95	0.94	0.95	0.92	0.94	0.93
3	6810	0.98	0.97	0.97	0.95	0.94	0.96	**17**	5634	0.95	0.95	0.95	0.93	0.93	0.92
4	16787	0.98	0.97	0.97	0.96	0.96	0.96	**18**	6355	0.95	0.94	0.95	0.93	0.92	0.94
5	5506	0.98	0.97	0.98	0.96	0.95	0.96	**19**	5737	0.94	0.92	0.93	0.92	0.92	0.91
6	166	0.97	0.96	0.97	0.96	0.93	0.95	**20**	5622	0.94	0.91	0.92	0.91	0.90	0.93
7	3676	0.97	0.95	0.96	0.95	0.93	0.96	**21**	5524	0.94	0.91	0.93	0.91	0.88	0.91
8	3700	0.97	0.96	0.97	0.96	0.94	0.96	**22**	6118	0.94	0.91	0.93	0.91	0.90	0.91
9	6508	0.95	0.95	0.95	0.90	0.95	0.93	**23**	16491	0.95	0.94	0.95	0.91	0.91	0.92
10	6412	0.96	0.94	0.96	0.93	0.92	0.95	**24**	3677	0.92	0.92	0.92	0.91	0.89	0.91
11	3723	0.97	0.96	0.96	0.95	0.95	0.95	**25**	5576	0.95	0.93	0.95	0.93	0.92	0.93
12	8270	0.97	0.96	0.96	0.94	0.95	0.95	**26**	8152	0.93	0.92	0.92	0.90	0.90	0.90
13	5975	0.96	0.95	0.96	0.94	0.95	0.94	**27**	3824	0.88	0.85	0.87	0.84	0.86	0.86
14	5179	0.99	0.99	0.99	0.99	0.98	0.99								

Table 3: Accuracy values for tNN using the thresholds at the break-even point

		1NN						tNN					
Class No	**GO ID**	**HEL**	**HE**	**HL**	**H**	**E**	**L**	**HEL**	**HE**	**HL**	**H**	**E**	**L**
1	9405	0.86	0.68	0.83	0.62	0.78	0.72	0.82	0.76	0.79	0.74	0.65	0.72
2	9055	0.89	0.73	0.88	0.75	0.67	0.78	0.84	0.78	0.81	0.68	0.72	0.72
3	6810	0.90	0.64	0.86	0.74	0.78	0.76	0.78	0.68	0.73	0.66	0.61	0.62
4	16787	0.94	0.69	0.90	0.84	0.87	0.82	0.84	0.74	0.78	0.61	0.70	0.72
5	5506	0.93	0.66	0.89	0.86	0.64	0.75	0.85	0.81	0.82	0.72	0.74	0.70
6	166	0.92	0.52	0.89	0.83	0.79	0.77	0.79	0.70	0.74	0.61	0.64	0.58
7	3676	0.86	0.61	0.83	0.72	0.73	0.72	0.77	0.68	0.73	0.65	0.60	0.62
8	3700	0.88	0.70	0.85	0.81	0.52	0.65	0.88	0.81	0.86	0.75	0.72	0.75
9	6508	0.93	0.68	0.90	0.81	0.82	0.82	0.74	0.61	0.68	0.58	0.54	0.54
10	6412	0.88	0.58	0.83	0.74	0.67	0.75	0.84	0.72	0.78	0.65	0.67	0.65
11	3723	0.85	0.62	0.82	0.74	0.67	0.67	0.82	0.76	0.78	0.69	0.68	0.68
12	8270	0.91	0.65	0.89	0.73	0.67	0.82	0.81	0.74	0.79	0.67	0.69	0.72
13	5975	0.94	0.69	0.92	0.81	0.89	0.82	0.75	0.60	0.66	0.53	0.56	0.52
14	5179	1.00	0.97	0.99	0.97	0.95	0.99	1.00	0.99	0.99	0.99	0.97	0.99
15	16020	0.85	0.66	0.81	0.66	0.69	0.72	0.74	0.69	0.70	0.70	0.62	0.60
16	5515	0.87	0.55	0.86	0.71	0.67	0.73	0.80	0.72	0.78	0.68	0.67	0.67
17	5634	0.84	0.58	0.81	0.75	0.61	0.64	0.79	0.76	0.77	0.71	0.69	0.69
18	6355	0.88	0.66	0.85	0.77	0.59	0.67	0.87	0.81	0.84	0.74	0.72	0.75
19	5737	0.85	0.48	0.84	0.83	0.80	0.77	0.67	0.56	0.63	0.48	0.54	0.54
20	5622	0.87	0.59	0.84	0.76	0.64	0.70	0.83	0.77	0.79	0.68	0.71	0.66
21	5524	0.91	0.52	0.89	0.85	0.83	0.81	0.77	0.64	0.72	0.54	0.56	0.57
22	6118	0.93	0.66	0.90	0.80	0.71	0.83	0.81	0.71	0.75	0.60	0.63	0.63
23	16491	0.96	0.75	0.94	0.90	0.84	0.86	0.88	0.78	0.82	0.61	0.66	0.68
24	3677	0.88	0.56	0.83	0.76	0.63	0.72	0.77	0.69	0.73	0.62	0.64	0.64
25	5576	0.93	0.82	0.93	0.79	0.84	0.88	0.93	0.90	0.92	0.88	0.85	0.88
26	8152	0.95	0.58	0.93	0.92	0.89	0.85	0.84	0.73	0.79	0.61	0.72	0.69
27	3824	0.94	0.45	0.92	0.88	0.87	0.85	0.74	0.60	0.68	0.48	0.51	0.55
mean		0.90	0.64	0.86	0.79	0.74	0.77	0.81	0.73	0.77	0.66	0.67	0.67

Table 4: mean AUC values of 1NN and tNN classifier for HEL, HE, HL, H, E and L classifications

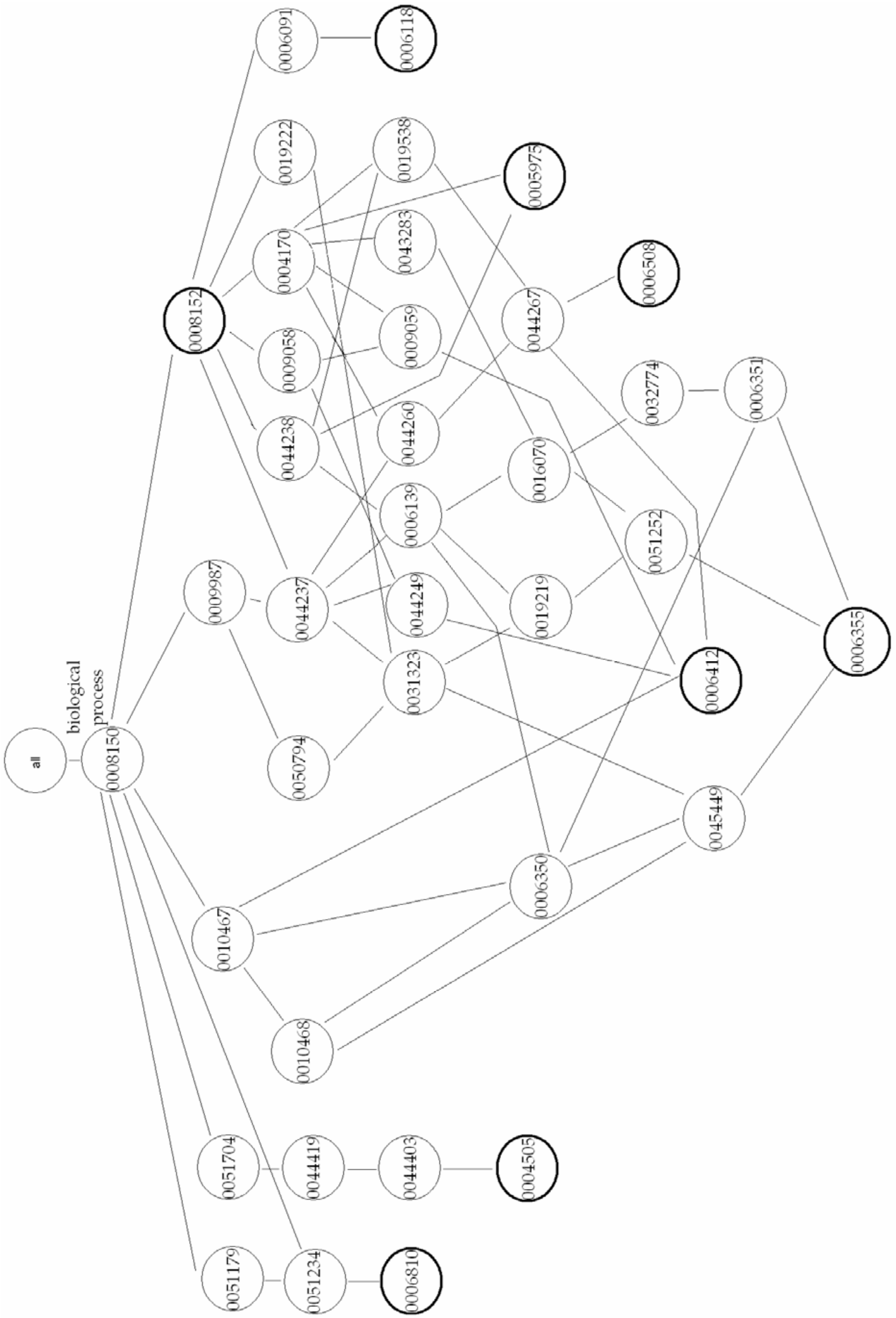

Figure 2: GO tree for biological process. Bold circles indicate the classes included in the dataset.

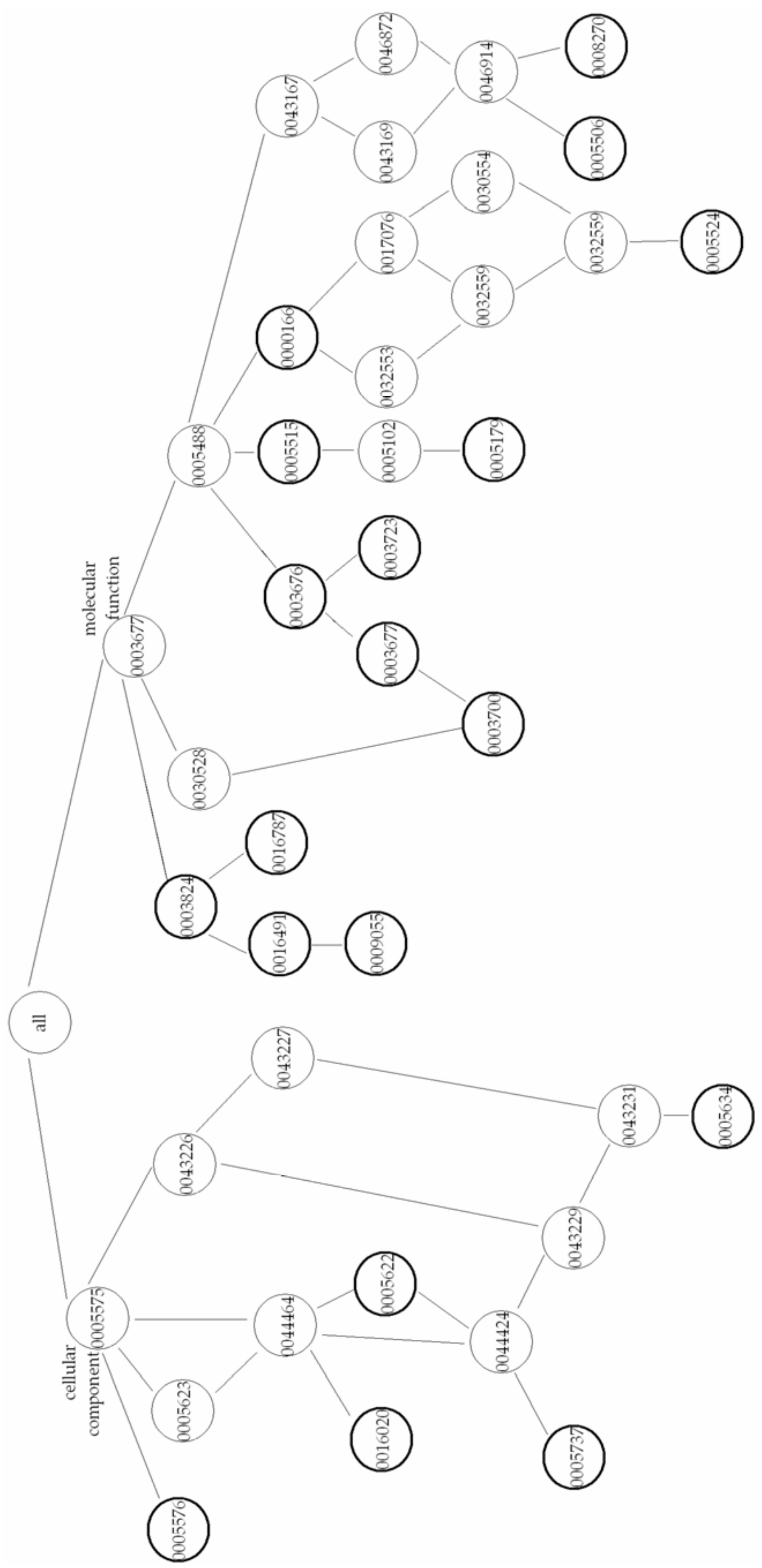

Figure 3: Go tree for cellular component and molecular function classes. Bold circles indicate the classes included in the dataset.