Gene Ontology-Based Annotation Analysis and Categorization of Metabolic Pathways

Ali Cakmak
cakmak@case.edu

Mustafa Kirac
kirac@case.edu

Marc R. Reynolds
reynolds@case.edu

Zehra M. Ozsoyoglu
meral@case.edu

Gultekin Ozsoyoglu
tekin@case.edu

Department of Electrical Engineering and Computer Science
Case Western Reserve University
Cleveland, OH, USA

Abstract

Functional characterizations of pathways provide new opportunities in defining, understanding, and comparing existing biological pathways, and in helping discover new ones in different organisms. In this paper, we present and evaluate computational techniques for categorizing pathways, based upon the Gene Ontology (GO) annotations of enzymes within metabolic pathways.

Our approach is to use the notion of functionality templates, GO-functional graphs of pathways. Pathway categorization is then achieved through learning models built on different characteristics of functionality templates. We have experimentally evaluated the accuracy of automated pathway categorization with respect to different learning models and their parameters. Using KEGG metabolic pathways, the pathway categorization tool reaches to 90% and higher accuracy.

1. Introduction

Metabolic pathways are networks of biochemical reactions, concerned with generating energy for driving various cell processes, and degrading and synthesizing many different molecules. A metabolic pathway contains a set of reactions (processes), where a reaction is a biochemical step that (a) specifies the consumption of specific input (substrate) molecules and the production of specific output (product) molecules, (b) usually involves one enzyme (or a set of enzymes), a gene product, catalyzing the reaction, and combinations of molecules as cofactors, activators, inhibitors, and regulators. Metabolite denotes any molecule, except for the enzyme, in a reaction, which is sometimes referred as a "step".

Gene Ontology (GO) [7] describes the central attributes of genes/gene products, and contains about 20,000 concepts organized in a hierarchical manner through *is-a* and *part-of* relations. GO has three subontologies, namely, *biological process, cellular location,* and *molecular function.*

In this paper, we describe PW-ANN [23], a GO-based pathway annotation and categorization system. Intuitively, pathways inherit annotations of their building blocks, namely, enzymes as gene products; and, our hypothesis is that functional (i.e., GO-based) annotations of pathways will provide new opportunities in understanding, categorizing, and comparing pathways, and in helping discover new ones.

To perform pathway annotation analysis and pathway categorization, our approach is to model each pathway as a network of GO-based *enzyme functions*, which we call the (*pathway*) *functionality template* (PFT). Via the use of PFTs, we change our focus to the *function* carried out in each step of a pathway, rather than the performer of the step, i.e., the enzyme. In the rest of the paper, we use the term "functionality" to mean "GO-based" functionality (represented by GO concepts from the *molecular function* subontology).

The motivation in using functionality templates comes from the following reasoning: (a) essential cellular actions in different organisms involve similar sequences of functional units, (b) it is known that most of the cellular actions are common to a large set of organisms regardless of their complexity [16], and (c) the same function in different organisms does not have to be carried out by the same genomic agent; it can be performed by a different genomic agent with similar functionality (i.e., functional annotation). Hence, by employing pathway functionality templates as opposed to pathways, we compensate for the variances in genomes of different organisms in terms of functionality, and yet accommodate the commonness in blueprints of biological processes.

The first capability of PW-ANN is to provide "enrichment/deficiency-based" statistical analysis of

19th International Conference on Scientific and Statistical Database Management (SSDBM 2007)
0-7695-2868-6/07 $25.00 © 2007 IEEE

functional annotations of pathways. PW-ANN allows users to query for GO molecular function concepts that *significantly* "enrich" a pathway (via a statistical significance notion--see Section 2.2), or for those in which the pathway is "deficient". The same query can also be posed in the opposite direction to search for pathways which are enriched (or deficient in) a given GO functionality annotation. In order to evaluate how significant an annotation is, we perform statistical analysis based on a hypergeometric distribution, as described in Section 2.2.

The second capability of PW-ANN is *pathway categorization,* i.e., organizing pathways into a particular hierarchy. Pathway categorization helps researchers browse and get a better grasp of pathways based on their vicinities with other pathways. More importantly, it gives possibly different, previously unknown, perspectives about pathways in the same category in that they may perhaps have similar functionalities that were not known beforehand. Thus, there is a need for bioinformatics tools that perform automated pathway categorization in different ways. We evaluate (section 5) PW-ANN's accuracy in performing pathway categorization in different organisms.

PW-ANN [23] is developed as part of a long-term bioinformatics research project, PathCase [22], a web-based application that provides various tools for storing, browsing, querying, visualizing and analyzing genomic pathways. PW-ANN has been fully integrated into PathCase, and is available on the web.

Contributions of this paper are:

- Statistical analysis of GO annotation significances within*functionality templates.*
- Solving the pathway categorization problem (as an example functional pathway analysis problem). The use of Support Vector Machines (SVM) [26], Decision Trees [24] and Naïve Bayes classification [20] for automated pathway categorization based on functional annotations of pathways, namely, functionality templates.
- Implementation and description of the pathway annotation and functionality visualization tool, PW-ANN. On the fly-creation of pathway functionality templates, visualization of templates at different levels of specificity based on the hierarchical organization of the GO levels. New querying schemes for pathways databases in terms of functionality associations to pathways.
- Experimental evaluation of pathway categorization accuracy. Experimental results show that PW-ANN provides an accuracy of over 90%.

This paper considers only metabolic pathways. But, the methods described here can also be applied to other biological networks, e.g., signaling pathways.

Section 2 describes functional annotation of pathways, namely, PFTs (Section 2.1) and statistical enrichment-deficiency analysis of functional annotations (Section 2.2). In section 3, we present automated pathway categorization based on features that include frequent patterns in PFTs. Section 3.1 defines the problem of locating frequent patterns in PFTs. Section 4 presents an overview of the features of PW-ANN. Section 5 discusses an automated mechanism to estimate pathway categories, and presents its experimental evaluation. In section 6, we briefly discuss related work.

2. Functional Annotation of Pathways

2.1. Pathway Functionality Templates

A pathway can be viewed as a graph of enzymes, where enzymes are nodes, and an edge between two enzymes indicates that the reactions catalyzed by the two enzymes directly share products and substrates. Figure 2.1 illustrates a pathway with five reactions where the reactions are represented by rectangles with the (abbreviated) name of their catalyzing enzymes inside rectangles, and circles labeled with the letter "m" represent metabolites (not explicitly named for simplicity) being consumed and/or produced. Figure 2.2 depicts the *enzyme graph* for the same pathway where metabolites of Fig. 2.1 are eliminated.

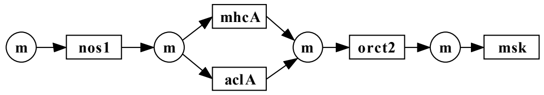

Figure 2.1. A sample pathway

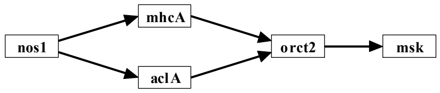

Figure 2.2. Enzyme graph for the pathway in Fig. 2.1

Usually, each enzyme is annotated with a set of concepts from GO. *Functionality template for a pathway* (PFT) is constructed by replacing the catalyzing enzyme of each process with its annotating GO concept(s). Figure 2.3 shows the pathway of Fig. 2.1 with its enzyme annotations. Note that, due to the hierarchical organization of GO concepts (Figure 2.5), annotations can be applied at *different levels of specificity*. In Figure 2.3, for instance, the last two steps can be considered as a single functional unit which is responsible for transporter activity. Therefore, a pathway can have multiple functionality templates depending on the levels of GO hierarchy from which the enzyme annotations are selected. Figure 2.4 depicts several functionality templates for the pathway of Fig. 2.1 in decreasing order of specificity.

In linking GO concepts with pathways, we used the mapping from EC (Enzyme Commission) Numbers to

19th International Conference on Scientific and Statistical Database Management (SSDBM 2007)
0-7695-2868-6/07 $25.00 © 2007 IEEE

GO concepts provided by the Gene Ontology Consortium [19] as well as the GenBank [4] records of genes that produce the enzymes in metabolic pathways. There are a total of 2,205 enzymes taking reaction-catalyzing roles in KEGG pathways (i.e., our experimental set of pathways). 1,086 of them have direct annotations available in the GO database. Among the1,119 enzymes which have no direct GO annotations, 978 of them are indirectly assigned at least one GO annotation through the "EC-to-GO" mapping given at the GO Consortium's web site. Thus, at the present time, only 141 out of 2,205 enzymes (about 6% of the total number) enzymes in our experimental pathways data set are not annotated (neither have direct annotations, nor can be indirectly annotated through EC-to-GO mapping).

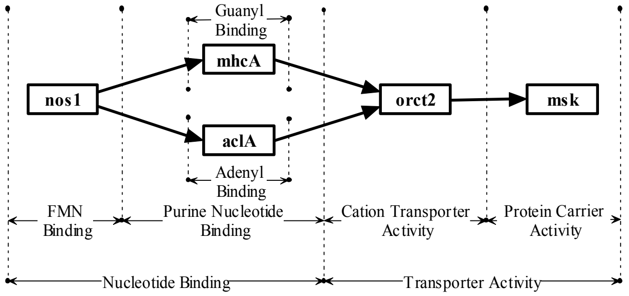

Figure 2.3. Pathway with Enzyme Annotations

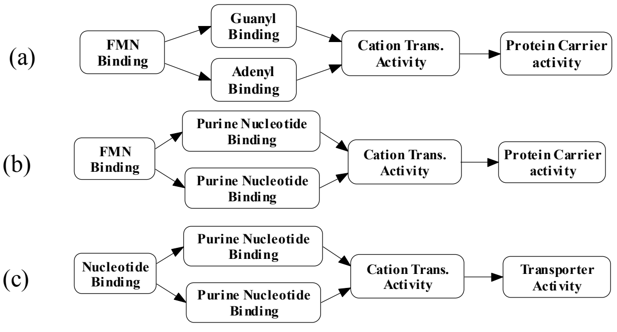

Figure 2.4. Alternative Functionality Templates

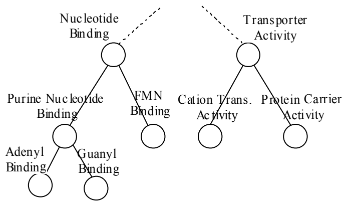

Figure 2.5. Hierarchical organization of GO concepts

2.2 Statistical Analysis of Annotations

Below we provide an "importance measure" of a GO concept annotation within a pathway, which allows the user to discriminate between an annotation which is common (among pathways of the same organism or over all organisms) and one which is not. We define importance by drawing an analogy from the field of Information Retrieval (Term Frequency – Inverse Document Frequency [25]), in which a word or a phrase (term) within a document is assigned a high weight (i.e., importance) if it has (i) a high frequency in a particular document, and (ii) a low frequency in all the documents in the reference set.

Given a reference set *S* of pathways, we use the hypergeometric distribution to determine the *statistical significance* [32] *of an annotation c in pathway p, p* ∈ *S*, which is defined as:

$$P(c,p,S)=\sum_{i=k(c,p)}^{n}\frac{\binom{N(S)-K(c,S)}{n(p)-i}\binom{K(c,S)}{i}}{\binom{N(S)}{n(p)}} \qquad (1)$$

where *k(c,p)* is the number of times the GO concept *c* annotates enzymes in pathway *p*, *n(p)* is the total number of enzymes in *p, N(S)* is the total number of enzymes in the set *S* of all pathways, and *K(c,S)* is the total number of times *c* annotates enzymes of pathways in *S*.

Def'n. *(GO Annotation Significance): Given a pathway set S, an annotation concept c, and a pathway p in S, c is significant in p if the statistical significance of c in p is less than the threshold γ, namely, P(c, p,S) < γ.*

The significance threshold *γ* that is used by PW-ANN is 0.01.

Def'n. *(GO Concept Enrichment/Deficiency): For a given GO concept c, assume K(c,S) out of N(S) processes in S are annotated by c. And, for a given pathway p with n processes, let k(c,p) be the number of processes annotated by c in p. We say that c enriches p if its annotation is significant in p, and the observed annotation count k(c,p) of c in p is greater than the expected annotation count n(p)*[K(c,S)/N(S)] of c in p, that is, k(c,p)>n(p)*[K(c,S)/N(S)]. Likewise, we say that c is deficient in p if its annotation is significant in p, and the observed annotation count k(c,p) of c in p is less than the expected annotation count n(p)*[K(c,S)/N(S)] of c in p, that is, k(c,p)< n(p)*[K(c,S)/N(S)]***.** *Furthermore, we say that c annotates p with the enrichment ratio R(c,p,S)= k(c,p)/ [n(p)*K(c,S)/N(S)]*

The pathways included in the reference pathway set *S* directly define the semantics of enrichment. If *S* contains all known pathways, and we find that *c* enriches *p*, then the indication is that the annotation of *p* by *c* is globally important. *Global Annotation Significance* is the GO annotation significance *P(c,p,S)* of GO concept *c* in pathway *p* with respect to all pathways in a database.

The reference pathway set *S* can also contain all pathways in a group, using the pathway groups defined by KEGG [15]. Considering the annotations of pathways in the same group as a reference set gives an indication of how significant an annotation is within a group (class) of pathways. *Group Annotation Significance* is the GO annotation significance *P(c,p,S)* of GO concept *c* in pathway *p* with respect to the pathways in the same group with pathway *p*.

In addition to displaying enrichments, PW-ANN discovers *annotation deficiencies*. A pathway *p* is deficient in an annotation *c* if *c* is significant in *p* (i.e., $P(c, p, S) < \gamma$) and the enrichment ratio *R* of *c* is less than 1; this means that *c* significantly under-annotates *p*. The 'missing' annotations are also included in the output data, defined as follows: a GO concept *c* annotates at least one pathway within *p*'s pathway group, but does not annotate *p*, and the expected number of annotations with the concept *c* is at least 1. We only output missing annotations from a given pathway group because the number of concepts annotating all pathways is much larger than the number of concepts annotating pathways within a group.

3. Pathway Categorization

We note that, at the present time, the number of "known" (i.e., curated) pathways is small (in low thousands) since most of the known pathways are curated manually from the literature and/or constructed in a wetlab environment (and categorized manually) by researchers (e.g., Reactome [21], KEGG [15], and MetaCyc [3]). However, there are also efforts to develop computational tools [9, 14] that allow construction, or more correctly, prediction, of pathways automatically. Therefore, in the very near future, once such computational tools become sufficiently mature in terms of accuracy, the number of available pathways will become considerably large.

In this section, we discuss alternative approaches for automated pathway categorization via the use of functionality templates. More specifically, we construct *pathway feature vectors,* containing as dimensions (a) information from Section 2.2 (e.g., existence of GO annotations in pathways, GO annotation counts, global annotation significances), (b) frequent sub-graphs (patterns) of PFTs (Section 3.1). Note that our pathway feature vectors have very high (more than 10,000) dimensions. We then apply binary or multi-class classifiers (section 3.3), namely, SVM (Support Vector Machines), Naïve Bayes classifier, and Decision Trees, to categorize pathways.

3.1 Frequent PF Pattern Discovery

GO annotation importance computation of section 2.2 considers pathways as enzyme lists with no structure. As a result, different occurrences of the same GO annotation in a pathway all have the same annotation significance. This is not always desirable: sometimes, it is more important to locate those GO concept occurrences where the annotated enzyme occurs in a portion of the pathway that is *functionally conserved* among many organisms (i.e., a sub-graph of the pathway is *frequent* among organism-specific versions of the pathway), in comparison to other annotations of the GO concept in non-conserved portions of the same pathway. Thus, discovering frequent sub-graphs of pathway functionality templates is an important task to identify and visualize functionally conserved portions of pathways. In this section, we formulate the frequent pattern location problem (from our recent work [9]), which is used in pathway categorization as part of the pathway feature vector.

Given a pathway, by eliminating its metabolites, we obtain an enzyme-only pathway graph (e.g., Figure 2.2) where a node represents a reaction in the pathway and is labeled with the catalyzing enzyme of the reaction, and an edge from enzyme e_1 to enzyme e_2 represents the information that a product of the reaction catalyzed by e_1 is a substrate to the reaction catalyzed by e_2.

In order to simplify the presentation, and decrease the level of the problem complexity, we transform all pathways and the GO into trees by node and edge replications through the following three actions (see [9] for details):

(a) Given a GO concept c with multiple parents in GO, a copy of the subDAG rooted at c is created for each parent of c. After this conversion, GO becomes a tree.

(b) Given an enzyme e with multiple GO annotations, for each distinct annotation of e, a distinct copy e' of e is created. After this conversion, each enzyme has a single annotation.

(c) Given a pathway P and its enzyme graph G, each enzyme e with multiple incoming edges is replicated such that each distinct copy e' of e is connected to a distinct incoming edge of e. After this conversion, enzyme graph of each pathway becomes a tree or a set of trees.

After the replication actions take place, to obtain *the most-specific PFT of a pathway*, we translate the enzyme-only pathway graph by replacing each enzyme with its GO annotation. Note that duplicate node occurrences of a PFT are kept separately by assigning each a distinct nodeid. The overall transformation results in *the most-specific* (i.e., the *most-detailed* in terms of its annotations) *PFT of a pathway* (e.g., Figure 2.4.a among the three PFTs of Figure 2.4). Each pathway has one most-specific PFT. However, a given pathway may have large numbers of PFTs due to (a) multiple GO annotations for an enzyme, and (b) GO functionality generalizations by using the true-path rule, as illustrated next.

Example. Figure 2.3 depicts the enzyme graph of the pathway of Figure 2.1 with GO annotations of the enzymes.

Next, we replace each enzyme with its most-specific annotation to obtain the *most-specific Pathway Functionality Template (PFT)* for the pathway. Note that, due to the true-path rule [7] on the hierarchical organization of GO concepts (Figure 2.5), a given PFT can be turned into a "more general" PFT by replacing any annotation with any of its ancestors. In the original PFT of Figure 2.4.a, the branching nodes that follow the first node, FMN Binding, can be replaced with their immediate parents to obtain the PFT in Figure 2.4.b. Similarly, in the PFT in Figure 2.4.b, the first and the last steps can be replaced with their ancestors to get the PFT in Figure 2.4.c. Therefore, a pathway can have multiple functionality templates depending on the levels in the GO hierarchy from which the annotations are selected. Also note that enzymes can have multiple annotations. In such cases, the original enzyme node can be replicated with all of its edges for each distinct annotation.

Our hypothesis is that comparisons of different pathways in terms of their functionalities may lead to new biological insights that are not possible by comparisons in terms of the involved enzymes.

As stated above, by a *PF pattern*, we refer to a subgraph of a PFT. Next, given a set S of PFTs for organism-specific versions of a pathway P_R, we formulate the problem of finding frequent PF patterns in S. First, we give some definitions.

Def'n *(Induced Pattern Set of a PF Pattern). Given a PF pattern F, the induced pattern set F* of F is the set of all PF patterns that can be obtained by (i) replacing any node in F with any of its ancestors in the GO ontology, and/or (ii) deleting any node and its incident edges from F.*

Example. Given the PFT in figure 2.4.a as a PF pattern F, the PFTs in figures 2.4.b and 2.4.c are both in F^*.

Note that $F_1 = F_2$ iff $F_1^* = F_2^*$.

Def'n. *Support of the PF pattern F, denoted as support(F), with respect to a set S of PFTs is the ratio of the number of PFTs that contain F to the total number of PFTs in S.*

It is easy to see that, given a set S of PFTs and a PF pattern F, for any $F_i \in F^*$, $support(F_i) \geq support(F)$ within S.

We also require the discovered frequent pattern set to be *minimal* and *complete*. For a set of patterns to be minimal, no pattern in the set should be included in the induced pattern set of another pattern in the set. And, completeness imposes a pattern set to include all possible PF patterns that satisfy the specified threshold requirements.

Def'n *(Minimality of a PF Pattern Set): A set R of PF patterns is minimal if, for any pair of patterns* F_i, F_j *in R,* $\{ F_k \mid F_k \in F_j^*$ *and* F_i *is a subgraph of* $F_k \} = \emptyset$.

Example. Consider a pattern set R that includes as patterns of both Fig. 2.4.a. and Fig. 2.4.b. R is not minimal as the induced pattern set of the pattern in fig. 2.4.a includes the pattern of fig. 2.4.b.

Def'n *(Completeness of a PF Pattern Set): Let S be a set of PFTs, and R(ε) be a set of patterns over the PFTs in S with support* $\geq \varepsilon$ *where* ε, $0<\varepsilon \leq 1$, *is the support threshold. Then a set of patterns R' with support threshold* ε *is complete with respect to S if R' contains R(ε).*

Frequent PF Pattern Location (FLP) Problem: *Given (a) a pathway* P_R, *(b) a set O of organisms* O_i, $1 \leq i \leq n$, *(c) a set S of PFTs* P_i, $1 \leq i \leq n$, *where* P_i *is the most-specific functionality template for the organism-specific version of* P_R *in organism* O_i, *(c) a threshold* ε, $0<\varepsilon \leq 1$, *the frequent PF pattern location problem is to find the PF pattern set* $F(P_R, O, S, \varepsilon)$ *such that* $F(P_R, O, S, \varepsilon)$ *is minimal and complete with respect to* ε.

In [9], we defined *Generalized Suffix Graphs* and gave algorithms to solve a variant of the FLP problem. For the experimental results of this paper, we have implemented these algorithms to solve the FLP problem.

3.2 Constructing Pathway Feature Vectors

In our pathway categorization system, our approach is to create a functionality template from each individual organism-specific pathway, and to collect features from the functionality templates to form *feature vectors* representing each pathway class. We employ the combination of four techniques to construct the elements (i.e., dimensions) of pathway feature vectors:

- Existence/nonexistence (E/N) of GO annotations (i.e., boolean values),
- GO annotation counts,
- Global annotation significances, and,
- Existence/nonexistence (E/N) of frequent PF patterns in pathways.

3.3 Binary and Multi-Class Classification

Next we perform automated pathway categorization by employing *binary or multi-class classifiers based on pathway features* as defined by feature vectors. There are a number of data mining tools that one can use as a classifier such as Bayesian networks and decision trees [11]. In this paper, we employ

- Support Vector Machines, or SVMs, a machine learning method commonly used to classify complex objects. The objective of SVM is to find a hyperplane which separates the negative and positive examples by the widest margin. A major advantage of using SVM is that the performance is independent of dimensionality [13]. Since in our case the number of dimensions is very large (more than 10,000), this is a significant advantage.

- Decision Trees [24], which have a higher time complexity than SVMs, but are simple to understand and interpret.
- Naïve Bayes Classifiers [20] which have a relatively simpler model as compared to SVMs.

In section 5, we compare the classification (i.e., pathway categorization) accuracy of SVM, Decision Trees, and Naïve Bayes Classifiers.

4. PW-ANN and New Queries

The PW-ANN system, implemented in .NET with C# on an MS SQLServer relational database, is accessible online, via PathCase [22]. In PW-ANN, users can view pathways annotated by a GO concept, and navigate from a pathway to GO concepts that annotate that pathway. To use PW-ANN, click to (1) *Browse Pathways* in PathCase, (2) any pathway class (e.g., *Amino Acids and Derivatives*), (3) any pathway (e.g., *Homocysteine pathway*), and (4) *GO Pathway Annotations (PW-ANN)*; then you can start using PW-ANN. For more details, please see [18].

PW-ANN provides new querying schemes for genomic pathway databases such as

- Given a pathway P (or a pathway fragment, or a super-pathway), find the relative "strengths" of top-k GO concepts, in sorted order, with respect to P. That is, *find the mapping from P to top-k GO concepts.*
- Given a GO concept G and a set of pathways, sort and rank the pathways with respect to their G "participations". That is, *find the mapping from GO concepts to top-k pathways*.
- *Given a GO annotation g within a pathway p, find the statistical significance of g with respect to GO annotations in a reference set of pathways S.*

5. Experiments

In this section, we present experimental evaluations to evaluate the accuracy of the automated pathway categorization tool of section 3. We discuss alternatives to represent pathways as multidimensional feature vectors and categorize the pathways using an unsupervised classifier model (i.e., Support Vector Machines (SVM) [26], Decision Trees (DT) [24] and Multinominal Naïve Bayes classifier (MNB) [20]).

5.1 Experimental Settings

In our experiments, we have categorized metabolic pathways from KEGG database [15] using the second-level categories defined in [28]. For the construction of classifiers, we have used YALE machine learning tool [29] and libsvm [5]. YALE provides a very flexible environment where users can construct experiments by inter-connecting the building blocks provided in YALE. In our experiments, for correlated attribute removal (Section 5.4) and learning model comparison (Section 5.7), we have used the YALE software. In all other classification experiments (Sections 5.1-5.6), we have used the stand-alone libsvm tool. The classification of the stand-alone libsvm tool is exactly the same with the libsvm component of YALE. However, the stand-alone libsvm works much faster than the one in YALE.

We have performed k-fold cross validation to evaluate the average classification accuracy of the automated pathway categorization tool. k-fold cross validation consists of splitting the entire data set into k equally-sized parts. Using k-1 of those parts, a learning model is constructed (training), then, using the remaining part, the model is tested. The same process is repeated k times, setting aside a different 'testing' and training portion of the data set each time. We used k=5 for all of our testing, unless mentioned otherwise. Accuracy is computed as the ratio of the number of correctly classified pathways to the number of all pathways.

In all experiments, we have used the default classification parameters of libsvm and YALE.

5.2 Feature Vectors, GO concepts as Attributes

In this experiment, we compare SVM classification performances of different feature vector types. We create features from metabolic pathways that are 1) GO Existence/Nonexistence (E/N) (Boolean) values that represent whether a GO annotation is found in a pathway or not (*the GO E/N dataset*), 2) GO counts (Integer): the number of enzymes in a pathway that are annotated with a GO concept (*the GO count dataset*), 3) Global annotation significance values (reals) (see Section 2.2) of GO concepts (*the GO significance dataset*). We repeat the classification experiment with different GO "frequency levels" (see Figure 5.1). For instance, frequency level 20 shows that we only used GO concepts that annotate enzymes in more than 20% of all pathways in the database. The reason for using GO frequencies is to observe accuracy changes against different GO specificity levels. The number of annotation occurences of a GO concept is referred as the "*informativeness*" [30] of the GO concept.

In our results, we obtained the best overall accuracy with the GO count dataset through most of the frequency levels. The GO E/N dataset produces similar, but worse results than the GO count dataset at frequency levels between 0 and 10. At higher frequency levels, the GO E/N data set provided the worst results, possibly because frequent, or generic, GO concepts are annotated to many of the pathways and SVM was not able to distinguish between the feature vectors with existence/nonexistence values at higher frequency levels.

Accuracy of classification at different annotation frequency levels shows how the classification of a partially annotated pathway may behave. For instance, recent work on GO-based prediction on biological networks [17, 31] shows that GO annotations are best predicted at lower GO-levels (i.e., closer to the root of the GO hierarchy) and the accuracy is lost at higher GO-levels (i.e., closer to the leaves of the GO hierarchy) with the most-specific GO concepts. In this experiment, we obtained the best accuracy with the GO count dataset for complete annotations (i.e., frequency levels between 0-10). Feature vectors with the GO significance dataset provide better results for GO concepts between the frequency levels 15 and 40. Finally, we obtained the best accuracy at the highest frequency level with the GO count dataset.

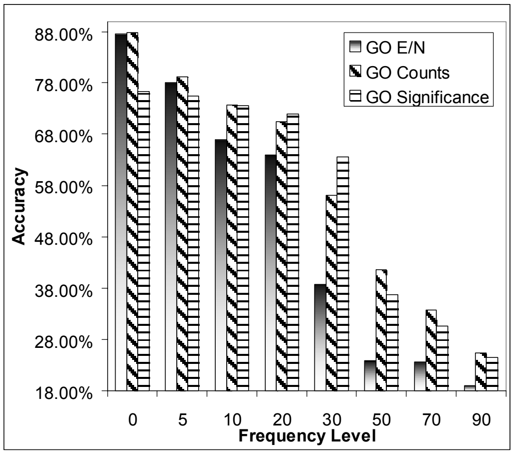

Figure 5.1: Comparing accuracies of SVM classifications on different GO-annotation based pathway features.

We explain the similar, but lower, accuracy of the GO E/N dataset in comparison with the GO count dataset as follows. GO counts provide richer information than existence/nonexistence values since the existence/nonexistence values are simply mappings of GO counts that are equal to or greater than 1 to the existence/nonexistence value true (i.e., existence), and mappings of GO counts equal to 0 to the existence/nonexistence value false (i.e., nonexistence). The results obtained from the GO significance dataset show that GO counts are not distinctive enough for GO annotations at intermediate levels of the GO hierarchy (i.e., frequency levels between 15 and 40) while the significance of an annotation is a better identifier of pathway classes.

5.3 Feature Vectors, PF Patterns as Attributes

In this experiment, we measure the accuracy of SVM classification using frequent PF patterns in metabolic pathways. As described in Section 3.1, we compute the frequent PF patterns among all pathways, and generate existence/nonexistence values as to whether a frequent PF pattern occurs in the PFT of a pathway or not (*the PF pattern dataset*). We selected 20% as our minimum support threshold while mining pathways for frequent PF patterns. In addition, we also created a third dataset by merging the GO count and PF pattern attributes of each pathway feature vector, called the *GO Count + PF pattern dataset*).

Table 5.2: Comparing SVM classifications based on structural and non-structural pathway features.

Data Set	SVM Accuracy
GO Count	87.85%
PF pattern	91.01%
GO Count + PF pattern	83.33%

As shown in Table 5.2, we obtained better classification results with PF patterns in comparison with GO counts which provided the best results among non-structural GO-based features (i.e., the GO E/N data set, the GO Count data set, and the GO Significance data set) in Section 5.2. In addition, merging GO count attributes with PF pattern attributes reduced the overall accuracy, possibly due to creating noise in the training data by mixing different attribute types.

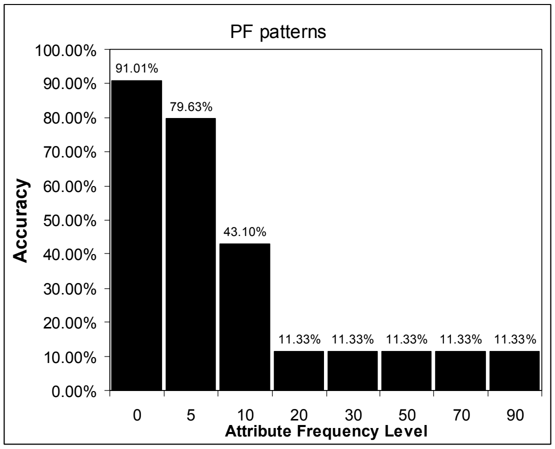

Figure 5.3: SVM classification accuracy values on the PF pattern dataset.

5.4 Feature

We repeated SVM experiments with PF patterns for different frequency levels. We observed that GO-based attributes (i.e., the GO E/N, the GO Count, and the GO significance data sets, as shown in Fig. 5.1) have more nonzero occurrences for the reference pathway set (i.e., KEGG pathways), in comparison with nonzero occurrences of PF patterns in the reference pathway set. Accuracy of the classification dropped to the minimum accuracy level (i.e., the random classification accuracy), 11.33%, at frequency levels greater than 20. This also shows that individual PF patterns are better indicators of pathway categories than individual GO annotations (i.e., lesser number of attributes, the same number of pathways, but better classification).

Selection with Attribute Correlations

In this experiment, we filter out the correlated attributes [1] in the datasets, and observe changes in the number of attributes and the SVM classification accuracy with respect to the allowed correlation amount between attributes. Correlated attributes are detected by using Pearson's correlation [1] and one of the correlated features is removed arbitrarily during the filtering process. As the correlation value, we use absolute values of Pearson correlation measurement (i.e., the correlation value is in the range [0,1]).

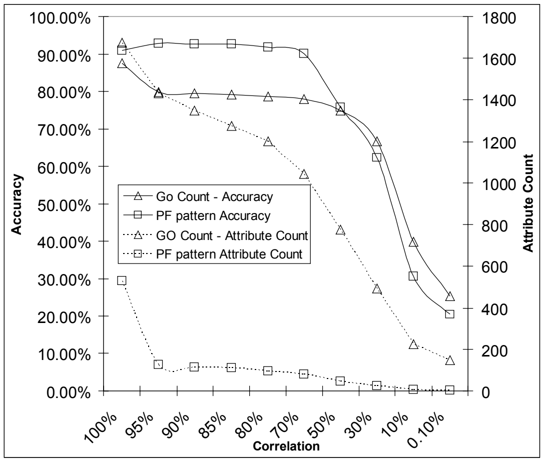

Figure 5.4: SVM classification using the PF pattern and the GO-count datasets.

In Figure 5.4, we compare the PF pattern and the GO Count datasets. X-axis shows the correlation filtering level, Y-axis on the left defines the accuracy values, and data points on solid lines plot the accuracy change w.r.t. the correlation filtering level. Y-axis on the right defines the number of attributes in datasets, and the data points on the dotted lines plot the change in the number of attributes in the datasets w.r.t. the correlation filtering level. The results show that the filtering of correlated attributes constantly reduces the accuracy in the GO count dataset. In comparison, from Figure 5.4, the accuracy of the PF pattern dataset first increases to 92.8% from 91.01% at 95% maximum correlation allowance (i.e., filtering correlated attributes with correlation above 95%), then steadily decreases. We observe in this experiment that PF patterns with only 125 attributes (at %95 max. correlation) provide better results (92.8%) than the GO count dataset with 1,673 attributes. This shows that frequent pattern information in pathways forms better attributes to represent and classify pathways.

5.5 Effect of k in k-fold Cross Validation

In this experiment, we changed the number of cross validation partitions (i.e., k) to see if the SVM learning model stabilizes for each k value, and how the accuracy changes by increasing the training set size while reducing the prediction set size. From Table 5.5., we observe little increase in the accuracy, which shows that the trained SVM model is precise enough even when half of the dataset is used as the training set, and the other half becomes the prediction set (i.e., k=2).

Table 5.5: k-fold cross validation. The SVM is trained on the PF pattern dataset after filtering out attributes with correlation above %95.

k	PF Pattern dataset accuracy
2	92.52%
3	92.68%
5	92.80%
10	92.84%
20	92.87%
50	92.88%

5.6 Binary Classification

In this experiment, we created binary classifiers for each pathway category separately, using the GO count dataset. The overall accuracy of binary classification is much higher (97.58%) than the accuracy of multi-class classification (87.85%) since binary classification simply classifies all pathways that are not in the target category into the same class. As a result, binary classification does not reflect the accuracy of a real-world pathway categorization task which is in fact a multi-class classification problem. However, as shown in Table 5.6, binary classification results are useful to observe the changes in the overall prediction accuracy for different pathway categories.

Table 5.6: Binary classification with pathway groups

Category Name	Overall prediction Accuracy	#path.	#ref.	#shared	#enz.
Amino Acid Mtbl.	96.71%	5688	16	332	522
Biodeg. of Xenobiotics	95.00%	2680	21	163	224
Biosynt. of Polyketides	99.36%	921	7	23	18
Biosynt. of Sec. Mtbl.	94.98%	2433	15	115	218
Carbohydrate Mtbl.	96.56%	5809	17	263	585
Energy Metabolism	98.55%	2109	6	227	254
Glycan Biosynt& Mtbl.	99.03%	1073	11	38	127
Lipid Metabolism	96.95%	2882	12	125	274
Mtbl. of Cofactors…	98.46%	3361	11	114	248
Mtbl of other A.A.	97.84%	2661	9	181	158
Nucleotide Mtbl.	99.93%	732	2	93	179

As shown in Table 5.6, to observe the relationship between the accuracy of predictions and pathway group properties such as the number of reference pathways (i.e., organism independent) in a group (**#ref.**), the number of all organism-specific pathways in a group (**#path.**), the number of enzymes that are shared with another pathway group (**#shared**), and the total number of enzymes in a group (**#enz.**), we computed the correlation (i.e., Pearson's correlation) between the binary classification accuracy and group properties.

There is a direct correlation between group properties and the prediction accuracy. Highest correlation (at -87.9%) is observed between the accuracy and the number of reference pathways. Negative correlation means that the accuracy increases while the number of reference pathways reduces. Other correlation amounts with the prediction accuracy are -53.3% with the number of pathways, -42.7% with the number of shared enzymes and -44.3% with the number of enzymes in a pathway group.

5.7 Comparison of Learning Schemes

In this experiment, we trained different learning models to evaluate the time performance of SVM. First, we used Multinominal Naïve Bayes (MNB) classification [20] to categorize metabolic pathways. As shown in Table 5.7, among the training models we applied, MNB is the fastest one requiring 49 seconds for training and making predictions 5 times (i.e., 5-fold cross-validation) on the PF patterns dataset and 60 seconds for GO Count dataset. The accuracy of MNB classification is around 88% for both datasets, which is better than the accuracy of SVM classification in GO Count dataset. We explain the difference between the accuracies of MNB and SVM in terms of different utilizations of attributes in different techniques. SVM finds the maximum-margin hyperplane that separates pathway categories in the attribute space. On the other hand, MNB measures attribute-class relationship for each attribute individually. As a result, the large number of GO annotations which has little effect on determining the class of a pathway reduces the accuracy of SVM. PF pattern dataset is a relatively simple dataset, including lesser number of attributes with a higher chance of determining the pathway categories. SVM performed better than the MNB method in this dataset.

The Decision Tree (DT) [24] classification technique produces the most accurate results with both datasets with the additional training time cost. DT model iterates several times on the training data to find the primary attributes to distinguish between categories, and makes use of the rest of the other secondary attributes (e.g., GO annotations which have little effect on determining the class of a pathway), only when the classification based on primary attributes are ambiguous. As a result, when fast classification is desired, SVM on PF pattern dataset are the best learning model choices. When the training data is not very large and the best accuracy is desired, the Decision Tree model with the GO count dataset is the best option. On the other hand, if the decision rules of the Decision Tree model are desired to be manually analyzed and tuned after the training phase, PF pattern dataset is a better choice since it promises high classification accuracy with a small number of attributes.

Table 5.7: Comparison of different learning schemes

	PF Pattern	GO Count
SVM	92.80% (1199) secs)	87.62% (1341 secs)
Decision Trees	94.61% (3147 secs)	95.74% (17380 secs)
Naïve Bayes	88.34% (49 secs)	88.54% (60 secs)

6. Related Work

The Gene Ontology has a free, web-based browser called GenNav [8], which has basic string search options, and provides visualizations of trees. For concepts with complex relationships, performance of GenNav is occasionally slow, perhaps because it is implemented in Perl, an interpreted language.

Fati-GO is another web-based tool [2] that analyzes a set of genes from a high-throughput gene expression data at a particular level of the GO hierarchy. Fati-GO displays over- and under-represented GO concepts in an uploaded or typed gene set. It allows users to upload or type in a list of genes to analyze and select the GO subtree to draw annotations. After using FatiGO to analyze the input set of genes, the user is given a list of GO concepts which annotate the input genes and the rate of annotation of each concept.

Onto-Express [6] is a Java-based program which analyzes the results of a microarray experiment. Normally, the results of such an experiment would be a long, unsorted list of genes. Onto-Express uses the GO hierarchy to organize those genes by the GO concept which annotates the genes in the list.

Gene Ontology Tree Machine [27] is a web-based micro-array analysis tool, and provides statistics on sets of "interesting genes" within the context of the Gene Ontology hierarchy. GOTM statistically computes a set of GO concepts that are found to be highly associated with the specific "interesting genes" as compared to the reference genes set.

GOTM, Fati-GO, and Onto-Express extract information about genes related to GO concepts while our approach is to analyze pathways with respect to GO concepts and analyze GO concepts with respect to pathways through examining annotations of the enzymes within pathways. We are not aware of any tools which enrich *pathways* with GO concepts or vice-versa.

8. Conclusions

Biological pathways provide a high level overview of cellular mechanisms governing vital processes in living organisms. In this paper, we have proposed a pathway functionality template model based on GO annotations of the enzymes involved in the pathway. Furthermore, we presented a statistical analysis for enrichment or (deficiency) of functional annotations in a pathway. In order to illustrate the use of pathway functionality templates, we built a pathway categorization framework using existing machine learning techniques. We studied categorization accuracy by employing different approaches for constructing feature vectors. Using frequent PF patterns as features provides significant increase in categorization accuracy. The results are promising to further pursue use of PFTs for mining biological pathways data.

8. Acknowledgments

This research is supported in part by the NSF award DBI-0218061.

9. References

[1] M.A. Hall. Correlation-based feature selection for machine learning. PhD thesis, University of Waikato, Department of Computer Science, Hamilton, New Zealand, April 1999

[2] Al-Shahrour, F., Diaz-Uriarte, R., Dopanzo, J.. FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. Bioinformatics 2004. http://www.fatigo.org

[3] Caspi, R et al. MetaCyc: a multiorganism database of metabolic pathways and enzymes. Nucleic Acids Research, 2006, Vol. 34, Database issue D511–D516, available at http://metacyc.org/

[4] Benson D.A et al. GenBank. Nucleic Acids Res. 2006 January 1; 34(Database issue): D16–D20. available at, http://www.ncbi.nlm.nih.gov/Genbank

[5] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines, 2001. http://www.csie.ntu.edu.tw/~cjlin/libsvm

[6] Draghici, Sorin et al. Onto-Tools, the toolkit of the modern biologist: Onto-Express, Onto-Compare, Onto-Design, and Onto-Translate. Nucleic Acids Research 31, 2003.

[7] The Gene Ontology Consortium. The Gene Ontology (GO) database and informatics resource. Nucleic Acids Research 32, 2004.

[8] GenNav, http://mor.nlm.nih.gov/perl/gennav.pl

[9] Cakmak A., Ozsoyoglu G. Mining Biological Networks for Unknown Pathways. Submitted for publication.

[10] Gusfield, D. Algorithms on Strings, Trees, and Sequences. Cambridge University Press, 1997.

[11] Jiawei Han, Micheline Kamber. Data Mining: Concepts and Techniques. The Morgan Kaufmann, 2000.

[12] Jun Huan, Wei Wang, Jan Prins, Jiong Yang: SPIN: mining maximal frequent subgraphs from graph databases. KDD 2004: 581-586

[13] Joachims, T. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In Proceedings of the 10th European Conference on Machine Learning, 1998.

[14] Pedro Romero, Jonathan Wagg, Michelle L Green, Dale Kaiser, Markus Krummenacker, Peter D Karp. Computational prediction of human metabolic pathways from the complete genome. Genome Biol. 2005; 6(1): R2.

[15] KEGG: Kyoto Encyclopedia of Genes and Genomes, http://www.genome.jp/kegg/

[16] Kelley BP, Sharan R, Karp RM, Sittler T, Root DE, Stockwell BR, Ideker T. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. Proc. of National Academy of Sciences USA. 2003 Sep 30; 100(20):11394-9.

[17] Kirac M, Özsoyoglu G, Yang J. Annotating Proteins by Mining Protein Interaction Networks. ISMB, 2006.

[18] Reynolds, M.R. Visualizing, Querying, And Mining Biomedical Ontologies. M.S. Thesis, Dept. of Electrical Engineering and Computer Science, Case Western Reserve University, 2006.

[19] Mappings of External Classifications to GO http://www.geneontology.org/GO.indices.html

[20] Mccallum A, Nigam K.: A Comparison of Event Models for Naive Bayes Text Classification. In: AAAI-98 Workshop on 'Learning for Text Categorization',1998.

[21] Reactome. Cold Spring Harbor Laboratory, European Bioinformatics Institute, and GO Consortium. World Wide Web URL - http://www.reactome.org

[22] PathCase: Case Pathways Database System, available at http://nashua.case.edu/pathways

[23] PW-ANN: Available through PathCase, at http://nashua.case.edu/pathways

[24] Quinlan R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA. 1993

[25] Salton, G., Buckley. C., Term Weighting Approaches in Automatic Text Retrieval. Information Processing and Management, Vol. 24, No. 5, pages 513-523, 1988.

[26] Vladimir N. Vapnik. The Nature of Statistical Learning Theory. Springer, New York, 1995.

[27] Zhang, B., Schmoyer, D., Kirov, S., Snodd, J.. GOTree Machine (GOTM): a web-based platform for interpreting sets of interesting genes using Gene Ontology hierarchies. Bioinformatics 2004.

[28] Metabolic pathway categories in KEGG, http://www.kegg.com/kegg/pathway/map/map01100.html

[29] YALE: Mierswa et al. YALE: Rapid Prototyping for Complex Data Mining Tasks, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06), 2006.

[30] Zhou,X. et al. (2002) Transitive functional annotation by shortest-path analysis of gene expression data. Proc. Natl Acad. Sci. USA, 99 (20), 12783–8.

[31] Deng,M. et al. (2004) Mapping Gene Ontology to proteins based on protein-protein interaction data. Bioinformatics, 20, 895–902.

[32] Statistical significance, at wikipedia web site http://en.wikipedia.org:80/wiki/Statistical_significance