CONGO: Clustering on the Gene Ontology

Paul J. Kennedy* and Simeon J. Simoff**

Faculty of Information Technology, University of Technology, Sydney, PO Box 123, Broadway, NSW 2007, AUSTRALIA

Abstract. Rapid development of technologies for the collection of biological data have led to large increases in the amount of information available for understanding diseases and biological mechanisms. However, progress has not been as fast in comprehending the data. Developments in understanding diseases and biological mechanisms governing them may come from combining data from different sources. We describe a method of clustering lists of genes identified as important to the understanding of a childhood cancer using functional information about the genes from the Gene Ontology. The measure of distance used in the clustering algorithm is notable for considering the relationship between terms in the ontology. Meaningful descriptions of clusters are automatically generated from the Gene Ontology terms.

1 Introduction

Rapid developments in bio–technology, measurement and collection of diverse biological and clinical data have led to revolutionary changes in bio–medicine and biomedical research. The data collected in bio–medical experiments or as a result of medical examination ranges from gene expression levels measured using microarray technologies to data collected in therapy research. Researchers are looking at discovering relations between patterns of genes (sequences, interactions between specific genes, dependencies between changes in gene expressions and patient's responses to treatment). The confluence of bio–technology and statistical analysis is known as bioinformatics. The "classical" statistical techniques used in bioinformatics — a broad range of cluster, classification and multivariate analysis methods, have been challenged by the large number of genes that are analysed simultaneously and the curse of dimensionality of gene expression measurements. As a rule, the gene–to–data points ratio is high (i.e. the so–called "wide" data table, i.e. if we are looking at N genes and our sample is of size m, then usually $N \gg m$). When there are more attributes than data records (cases), problems may arise (for example, there can be strong correlations between some of the attributes, or the covariance matrix may become singular, the curse of dimensionality may begin to bite). This challenge has attracted the attention of researchers in the two very closely related fields of "data mining" (initiated by

* paulk@it.uts.edu.au
** simeon@it.uts.edu.au

researchers in databases (see [1])) and "intelligent data analysis" (initiated by researchers working in the area of mathematical statistics and machine learning (see Chap. 1 in [2])). Bearing in mind that researchers and research communities often disagree about the precise boundaries of their dedicated field of investigation, further in this paper we refer only to data mining [3] as the "analysis of large observational data sets to find unsuspected relationships and to summarise the data in novel ways that are both understandable and useful to the data owner". There is a number of ways in which data mining is expected to be able to assist the bio–data analysis (see [4] for brief overview). One important area are the tasks of similarity search, comparison and grouping of gene patterns and assisting in understanding these patterns in medical bio–data, as many diseases are triggered by a combination of genes acting together. The work presented in this paper is in this area.

Addressing the "Wide" Data Table Problem

Having many more genes than data points offers a number of strategies for the analysis of such data [5], that can be grouped in three broader categories: "summarise then analyse" (STA), "analyse then summarise" (ATS) and "summarise while analysing" (SWA). STA scenario uses an unsupervised learning technique (e.g. cluster analysis) to reduce the large number of genes to gene clusters (or gene profiles). The cluster representations then are used for predictive modelling (see [6]). In ATS scenario, modelling is conducted initially for each gene, producing some statistics, and then one can apply some threshold (for example, select all genes with value of that statistic above the threshold). SWA approach addresses the issues of possible existence of some relations between the genes, hence, suggests to proceed with summarisation and classification in a single step. For example, regression tree model [7] can be used to identify a small subset of predictive genes.

The above presented scenarios do not consider the utilisation of already existing knowledge about relations between genes to assist the outcome of the data mining step. The approach proposed in this paper extends the STA scenario, by imposing the results of the initial clustering of the genes with further clustering over an ontology that relates the genes in the input clusters. This approach can be labelled as "summarise, impose, then analyse" (SITA).

Cluster Analysis and Visualisation

As we have mentioned earlier, clustering algorithms divide the set of genes into groups so that gene expression patterns within a group are more similar than the patterns across groups. Most clustering techniques include a "magic" set of parameters, that one needs to adjust to get "good" clusters. However, in the case of gene expression data sets, the selection and "tuning" of these parameters may not be that intuitive and obvious, due to the high dimensionality of the space. Hence, clustering relies substantially on visualisation. An efficient visualisation schema allows to expose problems with the clusters, prompting towards

some intervention, for example, selection of different similarity and inter–cluster distance measures, or forcing some of the clusters into one group. The paper presents a visualisation method that supports the proposed SITA scenario.

In this paper, we use information from one source (the Gene Ontology [8]) to gain an understanding of a list of genes that were generated as the result of another data mining step. The list of genes is clustered into groups of genes with similar biological functionality. Descriptions of the clusters are automatically determined using the Gene Ontology (GO) data.

The broad goals of our bioinformatics project are to improve the understanding of genes related to a specific form of childhood cancer. Data regarding the relative expression levels of genes (in tumour cells compared with normal cells) is combined with clinical data (concerning the tumours and patients) to form a list of "interesting" genes. Details of this step are not relevant to the techniques explored in this paper.

The Gene Ontology is a controlled vocabulary of terms that describe gene products in terms of their effect in the cell and their known place in the cell. Terms in the ontology are interrelated. For example, a "glucose metabolism" *is a* "hexose metabolism" (see Fig. 1). In this example, "hexose metabolism" is a more general concept (or term) than "glucose metabolism". There are currently around 16,000 terms in the Gene Ontology and each gene is associated with between two and ten terms. The relationships between terms in the ontology allow us to measure the similarity between genes in a functional way. For example, one gene may be associated with the term "carbohydrate metabolism" and another gene associated with "alcohol metabolism". As can be seen in Fig. 1 both of these terms are child terms (or more specific concepts) of "metabolism". Hence, they are related quite closely.

The list of genes, then, is clustered according to the associated Gene Ontology terms. The clustering considers the interrelationships of terms in the ontology. Once clusters are created, the terms in the Gene Ontology permit the automatic construction of cluster descriptions (in terms of the Gene Ontology concepts).

The method of clustering over an ontology is general and may be applied to (non–biological) data associated with other ontologies.

Applying information from the Gene Ontology to cluster genes allows for an understanding of the genes and their interrelationships in functional terms. Currently biologists search through such lists gene–by–gene analysing each one individually and trying to piece together the many strands of information. Automating the process, at least to some extent, would allow biologists to concentrate more on the important relationships rather than the minutiae of searching as well as give savings in time and effort.

Related Work

Other workers use the Gene Ontology. There are a variety of browsers for the Gene Ontology linked from their web site [9]. In general, such browsers have facilities such as: (i) traversing the large Gene Ontology and viewing their interrelationships; (ii) finding Gene Ontology terms associated with ensembles of

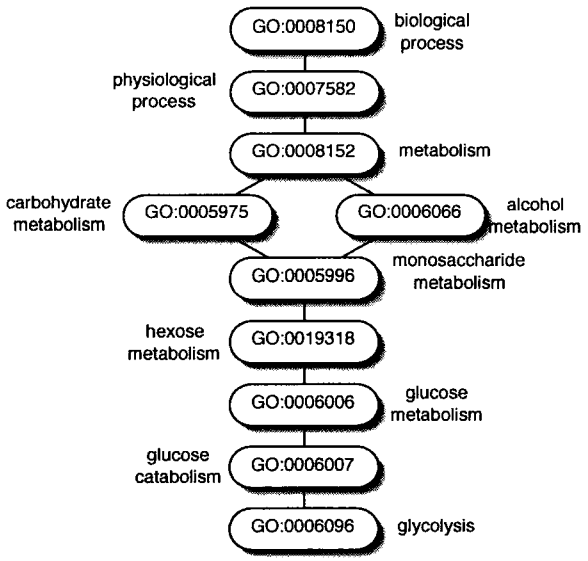


Fig. 1. A small section of the GO hierarchy from the "biological processes" ontology. Each node is a term in the ontology. Inside each node is the identifier for the term and beside is the term itself. More general terms are towards the top of the diagram. All links shown are is–a relationships that are directed upwards

genes; or (iii) finding known genes associated with particular Gene Ontology terms, to name a few.

Many tools (for example, eGOn [10] or FatiGO [11]) take as input a list of genes (often resulting from microarray experiments) and map the genes to GO categories. Most of these tools additionally allow comparison of GO mappings between different gene lists usually with some statistical measure of the similarity of distributions. The tools GOMiner [12] [13] and EASE [14] [15] additionally look for "biological themes" in lists of genes. That is, they identify the predominant set of GO terms that describe the entire gene list. They have a similar goal to the method we propose, except that we first cluster the data into subsets of related genes.

Hierarchical information is also used with other data mining techniques (possibly unrelated to biology). For example, [16] and [17] use ontological information to mine "generalized" association rules. The "basic" algorithm in [17] takes an approach that is reminiscent of ours (ie. simply including information from higher in the tree). Both the generalized association rules and the ontological clustering in this paper use the idea of combining specialised concepts but have different goals. The generalized association rules combine them to produce stronger rules, whereas we combine concepts to build looser forms of equivalence to make the clustering more flexible.

Method Overview

The cluster analysis and visualisation described in this paper takes as input (i) a list of genes highlighted from a previous data mining step and (ii) data from the Gene Ontology. The previous data mining step used gene expression data (from cDNA microarray experiments) and clinical data describing the tumour cells in detail, effect of drug protocols and (human) classifications of patients into high or low risk categories. cDNA microarray experiments are a recent technology available to cellular biologists that measure the relative expression levels of thousands of genes in cells at one instant. Expression levels of genes in a test sample (i.e. tumour cells) compared to genes in a control sample (i.e. "normal" cells) are measured.

Gene Ontology terms are associated with each gene in the list by searching in the SOURCE database [18]. The list of genes is clustered into groups with similar functionality using a distance measure that explicitly considers the relationship between terms in the ontology. Finally, descriptions of each cluster are found by examining Gene Ontology terms that are representative of the cluster. Graphs of Gene Ontology terms for each cluster together with cluster descriptions give a visualisation of each cluster in functional terms.

2 The Gene Ontology

The Gene Ontology [8] is a large collaborative public database constructed by researchers world–wide. It provides a set of controlled vocabularies (i.e. ontologies) of terms that describe gene products in terms of their effect in the cell. That is, their functionality. The goal of the Gene Ontology is "to produce a dynamic, controlled vocabulary that can be applied to all eukaryotes even as knowledge of gene and protein roles in cells is accumulating and changing" [8].

As described in Sect. 1 the Gene Ontology contains terms and their interrelationships (parent/child, general/specific, etc). Three ontologies are defined in the Gene Ontology: (i) biological processes, (ii) cellular components, and (iii) molecular functions. The ontologies are directed acyclic graphs (DAGs) where the terms form nodes and two kinds of relationships form edges: "is–a" relationships such as "glycolysis" is–a "glucose catabolism" and "part–of" relationships such as "nuclear chromosome" is part–of "nucleus". Apart from the specific individual terms, the Gene Ontology is unremarkable in this regard. All ontologies are DAGs of terms.

Each term in the ontology has a number of attributes: the term itself (eg. glycolysis), a unique accession number (eg. GO:0006096), and a definition (eg. the breakdown of a monosaccharide (generally glucose) into simpler components, including pyruvate). There may also be technical references to the definition (eg. links to PubMed articles), cross references into other biological databases, synonyms and comments.

There are a number of benefits of using the Gene Ontology as part of the data mining process. It is large (7045 terms in the Molecular Function ontology,

7763 terms in the Biological Process ontology and 1335 terms in the Cellular Component ontology as of 16 September 2003 [9]) and well worked on by researchers (16 member organisations of the Gene Ontology Consortium as of August 2003 [9]). Entries are curated before being added to the ontology. The ontology may be accessed in the RDF XML file format. In this computer legible form it is easier to apply the information to data mining methods and immediately richer than by determining similar information with text mining methods.

GO terms may be associated with genes using databases like SOURCE [18] as long as accession numbers of genes or gene names are known. See Table 1 for an example.

Table 1. GO terms associated with an example gene (named CLK1) for each of the three ontologies.

CLK1 (CDC–like kinase 1)
Molecular Function
GO:0004715 non–membrane spanning protein tyrosine kinase activity
GO:0005524 ATP binding activity
GO:0004674 protein serine/threonine kinase activity
GO:0016740 transferase activity
Biological Process
GO:0006468 protein amino acid phosphorylation
GO:0008283 cell proliferation
GO:0000074 regulation of cell cycle
Cellular Component
GO:0005634 nucleus

3 Clustering over Ontologies

Many algorithms exist for clustering data (see for example [19] or [20]). The data we wish to cluster is slightly different to normal, however, and this advises our choice of algorithm and distance measure.

There are two main differences between our clustering and "normal" cluster analysis. The first difference is that there are a different number of attributes (GO terms) for each gene to be clustered whereas usually the number of attributes in a dataset is the same for all records. Secondly, we are interested in complex relationships between terms (as a result of the structure of the ontology) so simply comparing values of terms with one another will not be sufficient.

Both difficulties stem from the fact that there is an ontology associated with the data. Once we solve the data mining problem of clustering over an ontology, the special case of clustering over the Gene Ontology will follow easily.

Similarities might be drawn with clustering text documents (for example into spam and non–spam), as there are different numbers of words in each document and complex relationships among the words (ontological ones too). One approach to clustering documents is to use a fixed length vector of word counts in each document, with each vector position representing a different word (drawn from an a priori prescribed list). In this way each document to be classified with potentially many different words and counts of words is reduced to a fixed number of attributes with all documents having the same number of attributes.

A similar approach could be applied to cluster the genes and GO terms. A fixed length binary vector of the union of all GO terms in the genes could be set up as the attributes for each gene. A bit would be set if the term was associated with the gene or unset if there was no association. Such an approach, however, suffers from two defects. Firstly, the vast majority of GO terms are only associated with one gene in the dataset. This would mean the binary vectors for genes would be very sparse and few similarities could be found with the vectors for other genes in the dataset. The other, more serious, problem with this approach is that it does not take into account the ontological relationships at all.

Our method solves the problem of different numbers of attributes by treating all the terms for a gene as essentially one attribute. The second problem of considering the ontological relationships is accomplished by using a more specialised distance function that compares a set of terms based on their relative positions in the ontologies, rather than just the value of the term, which is, essentially, meaningless.

The distance function, then, is the crucial element and the particular clustering algorithm used is a secondary consideration. We use a simple clustering algorithm named the Modified Basic Sequential Algorithmic Scheme (MBSAS). This particular algorithm was chosen because of its simplicity and because it is not necessary to specify a priori the number of clusters. One of many other algorithms (eg. k–means) could have been used instead.

In the following two subsections we will describe in more detail the distance measure and the MBSAS clustering algorithm.

Distance Measure

The elements to be clustered have different numbers of attributes and this means that a special distance measure must be used. The distance measure is special in that it measures distances across the ontology. The distance measure is in some ways more important than the actual clustering algorithm as any of many different clustering algorithms may be used, but a distance measure similar to this must be used to traverse the ontology.

We use a function adapted from the Tanimoto Measure [19] [20]. The Tanimoto measure provides a measure of similarity between sets:

$$\frac{n_{X\cap Y}}{n_X + n_Y - n_{X\cap Y}} = \frac{n_{X\cap Y}}{n_{X\cup Y}} \tag{1}$$

where X and Y are the two sets being compared and n_X, n_Y and $n_{X\cap Y}$ are the number of elements in the sets X, Y and $X \cap Y$ respectively.

In our situation, the "sets" being compared are the GO terms for two genes. However, for reasons which will become clear, "bags" (where elements may be repeated) are used rather than sets.

An important characteristic of our distance measure is that it considers terms higher in the ontology. This is because the GO terms themselves are simply constant values with no implicit relationship to other terms. As in any ontology, the relationship between terms arises from their relative positions in the hierarchy. So, for each gene we wish to compare, we add to the gene's associated GO terms all terms higher in the ontology. These terms form a "background" or context to the terms explicitly associated with the gene. However, as the ontologies are tree–like, two terms in a gene often have the same ancestors. We include the parent terms each time they are encountered, so we require bags rather than sets.

Terms higher in the ontology represent terms that are more general. Although general terms are a factor in the comparison, the more specialised terms (i.e. lower in the hierarchy) are more important. For this reason, when counting the number of terms in a bag, terms are weighted by their distance from their descendent GO term explicitly associated with the gene. In effect, we calculate a "weighted" cardinality of the bag of GO terms.

The final distance function used, then, is

$$D_{X,Y} = \frac{n'_{X\cap Y}}{n'_X + n'_Y - n'_{X\cap Y}} = \frac{n'_{X\cap Y}}{n'_{X\cup Y}} \tag{2}$$

where X and Y are the two bags of terms being compared and n'_X, n'_Y and $n'_{X\cap Y}$ are the weighted cardinalities of the bags X, Y and $X \cap Y$ respectively given by

$$n'_X = \sum_{i\in X} c^{d_i} \tag{3}$$

where X is the bag of GO terms, d_i is the distance of element of X with index i from its associated descendent in the original set of GO terms for the gene, and c is the weight constant. The weighted cardinality of the other bags is similarly defined.

The more general terms provide a context for the lower level terms directly associated with genes. The c parameter allows variation of the importance of the "context" to the comparison. A value of $c = 0$ means that ancestral terms are not considered. A value of 1 would mean that all terms are considered equally as part of the context. Plainly, in this case though, the very general terms would be regarded as overly important. The c parameter, then, may be viewed as a sort of "constant of gravity" for the clusters. The higher the value of c, the easier it is that distantly related genes gather into a cluster. We arbitrarily chose $c = 0.9$ for our experiments.

Other distance measures apart from a gene–to–gene distance are also required for use in the clustering algorithm. A measure of the distance between a gene

and a cluster of genes is determined by taking the average distance from the gene to each gene in the cluster. Similarly when calculating the distance between two clusters of genes we use the average of the distances for each gene of one cluster to the genes in the other cluster. An alternative to using the mean distances would be to use minimum (or maximum) distances. We plan to explore these possibilities in the future.

Cluster Algorithm

With the intention of attacking the clustering problem as simply as possible, we use a standard simple clustering algorithm called Modified Basic Sequential Algorithmic Scheme (MBSAS) as described by [19]. MBSAS has two advantages compared with other algorithms such as the ubiquitous k-means algorithm. It is (i) not necessary to specify a priori the number of clusters; and (ii) the data is presented to the algorithm only a few times (depending on the particular variation of MBSAS chosen).

The variation of MBSAS we use is dependent on three parameters (and one other parameter is necessary for the distance measure). These parameters are shown in Table 2. Whilst MBSAS does not require an explicit parameter for the number of clusters, the parameters (Θ, q and M_1) have the same effect.

Table 2. Parameters used in the Modified Basic Sequential Algorithmic Scheme clustering algorithm. The last parameter is used only in the distance measure and is not formally part of MBSAS. See text for a detailed description of c.

Parameter	Meaning
Θ	Minimum distance for points to be considered to be in the same cluster. (Theodoridis and Koutroumbas [19] call this the "threshold of dissimilarity").
q	Maximum allowable number of clusters.
M_1	Minimum distance for clusters to be deemed separate before they are merged.
c	Discount weight applied to GO nodes in the ontology.

The MBSAS algorithm has four main steps as described below. The first two steps are mandatory, whilst the latter two are optional.

1. `determine_clusters`
2. `classify_patterns`
3. `merge_nearby_clusters (optional)`
4. `reassign_points (optional)`

The `determine_clusters` step determine the initial clusters. It chooses up to q data points that are sufficiently distant from one another (using the Θ parameter) as point representatives.

After finding the initial clusters the next step (`classify_patterns`) classifies the rest of the patterns into the cluster that is closest using $D_{X,Y}$ as defined in (2).

Theodoridis and Koutroumbas [19] describe two general drawbacks of sequential clustering algorithms. They are (i) that clusters may arise that are very close together and (ii) that they are sensitive to the order of presentation of the data. The third and fourth steps address these problems respectively. Although optional, we always perform them.

The `merge_nearby_clusters` step identifies clusters having a distance less than the value of parameter M_1 and merges them together.

Finally, in the `reassign_points` step, all points are reassigned to their closest cluster so as to minimise the effects of the presentation order of the data and any changes due to the `merge_nearby_clusters` step.

4 Experiments

As described in Sect. 1 the data used for this paper was a list of genes highlighted as the result of a previous data mining procedure. Information from the Gene Ontology was matched to the genes using the SOURCE database.

There are, at this stage, two goals for our experiments: (i) discovery of parameter values that produce acceptable clusters and (ii) determination of ways to visualise the clusters.

The parameter values Θ and M_1 are dependent on the range of values returned by the distance measure $D_{X,Y}$ and have been determined largely by trial and error. In the experiments described in this paper, Θ is set to 0.001 and M_1 to 0.1. The maximum number of clusters (q) is set at 5 and, as described above, c, the discounting constant for more general terms is set at 0.9.

Visualisation of clusters is made difficult by the fact that there is no clear way to transform genes into coordinates to plot on a single graph because each gene is identified by different numbers of GO terms. So we plot the terms for all the genes on a graph with their relationships shown in different shades for each cluster. We also automatically build cluster descriptions from the terms in each cluster.

5 Results

With the parameters values given above (i.e. $\Theta = 0.001$, $M_1 = 0.1$, $q = 5$ and $c = 0.9$) five clusters are found as shown in Table 3. Half of the genes have been allocated to one cluster. The rest of the genes have been split into four smaller clusters with one cluster containing only two genes. Such a tabular representation does not increase our understanding of the clusters as the gene accession codes are not descriptive.

With this in mind, we plotted the subset of terms associated with the clustered genes as nodes on a graph with relationships represented by edges and the

Table 3. Clusters found with the MBSAS clustering algorithm. The codes AA*nnnn* are GenBank accession codes.

Cluster Number	Gene Count	Genes
0	6	AA040427 AA406485 AA434408 AA487466 AA609609 AA609759
1	2	AA046690 AA644679
2	6	AA055946 AA398011 AA458965 AA487426 AA490846 AA504272
3	9	AA112660 AA397823 AA443547 AA447618 AA455300 AA478436 AA608514 AA669758 AA683085
4	20	AA126911 AA133577 AA400973 AA464034 AA464743 AA486531 AA488346 AA488626 AA497029 AA629641 AA629719 AA629808 AA664241 AA664284 AA668301 AA669359 AA683050 AA700005 AA700688 AA775874

GO nodes of a cluster localised to one part of the graph as much as possible (Fig. 2). The clusters are represented by the five large boxes with the cluster numbers (as listed in Table 3) given inside each box. Nodes inside the clusters are the GO terms associated with genes in that cluster. More general terms are on the right hand side of the diagram. Edges between nodes represent the links in the ontology. Some terms, particularly the more general ones at the right hand side of the diagram, have links from terms in a different cluster. Each node is shown in only one cluster box, but links between the boxes show where GO terms are shared by genes in the different clusters. The grey scale of the link represents the cluster that link is in. Also, a darker grey scale is used for links in the original dataset whilst a lighter shade is used for relationships inferred from traversing the ontology. Inside some cluster boxes may be seen links from a different cluster (if both child and parent terms are drawn in one cluster box, but the link is also in another cluster). For example, inside the large middle cluster (representing cluster 4) may be seen some links associated with the second top cluster (representing cluster 3) although this is difficult to see on the diagram. It is likely that these are either outliers or indicators of poor clustering.

Figure 2 is reminiscent of the dendrograms that are used in hierarchical clustering. This is hardly surprising since both methods are dealing with hierarchies. However, in Fig. 2 the length of edges is not correlated to the distance between nodes (as in dendrograms). We will apply a hierarchical clustering algorithm in the future.

Figure 3 shows essentially the same information as Fig. 2 except that the more general terms are at the bottom of the diagram. To improve the readability of the diagram, the cluster boxes are in a different order than in Fig. 2. Again, cluster numbers are given inside each box. The GO terms lying along the bottom edges of the cluster boxes are clearer in this diagram, particularly those on the left– and rightmost cluster boxes (clusters 3 and 4). These terms are part of

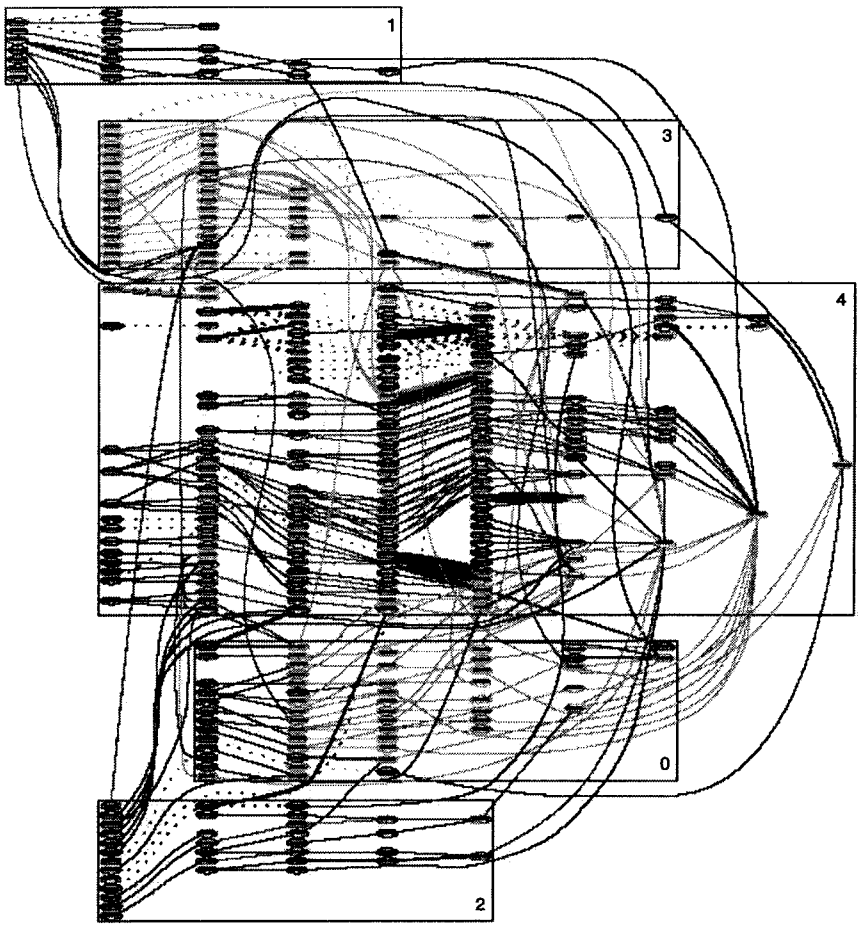


Fig. 2. Parts of the GO hierarchy associated with genes being clustered. More general terms are at the right of the diagram. See text for description of graph

the most general descriptions for a cluster that *do not also* describe another cluster. Figure 4 shows a closer view of the terms at the bottom edge of the large rightmost cluster (number 4). These terms are used to automatically determine cluster descriptions. Another feature visible in Fig. 3 are the links that fly from one cluster to another. These are important because they show where cluster meanings overlap or blur together. The rope of links at the bottom right of the diagram is unimportant as these links are to the most general terms and therefore, the least descriptive for our purposes.

A good visualisation of clusters should make evident the properties that genes in a cluster share. Essentially this entails a functional description of a cluster. A good description might also state how the cluster differs from other clusters.

The ontology is able to describe how genes are similar. Cluster descriptions are inferred in the following way. Starting with all the GO terms directly associated with genes in a particular cluster, we climb the hierarchy replacing GO terms with their parent terms. Terms are replaced only if the parent node is *not* associated with genes in another cluster (or is one of any of the ancestor terms in another cluster). This results in a list of terms for a cluster that describe in the most general way possible the union of functionalities of all genes in that cluster (but not so general that it describes another cluster).

Cluster descriptions derived in this way are shown in Table 4. Only the *is–a* relationships were followed to build this table. We expect to trace the *part–of* relationships in future work. There are far fewer *part–of* relationships in the hierarchies so we do not believe that omitting them affects the results. The cluster descriptions give some insight to the genes in the cluster and also give feedback on the quality of the clustering. The terms listed in the table are associated only with genes in each cluster and not in any other cluster.

Cluster 0 in Table 4 has no terms that are associated with more than one gene. This suggests that the genes in the cluster are either unrelated or related only in ways that are sufficiently high level that the terms exist in other clusters. This suggests that the quality of the cluster is not good.

The other clusters, however, have genes that are more strongly interrelated. Cluster 1 contains at least two genes that are related to the cell cytoskeleton and to microtubules (microtubules are components of the cytoskeleton). Cluster 2 contains three or four genes associated with signal transduction and cell signalling. Cluster 3 contains three or four genes related to transcription of genes and cluster 4 seems to contain genes associated with RNA binding.

The question, however, may be asked: what about the other genes in the clusters? What is their relationship? Are these genes unrelated to the "core" description of the cluster and just bundled into the cluster because the maximum number of clusters q has been reached, or are there more subtle relationships? The simple statistic of the number of genes associated with each GO term in the cluster is insufficient to answer the question. The names of the individual genes are required. This will be investigated further in future work. Also, we plan to cluster the data into more clusters, perhaps with an hierarchical clustering algorithm to determine whether better descriptions and "tighter" clusters result.

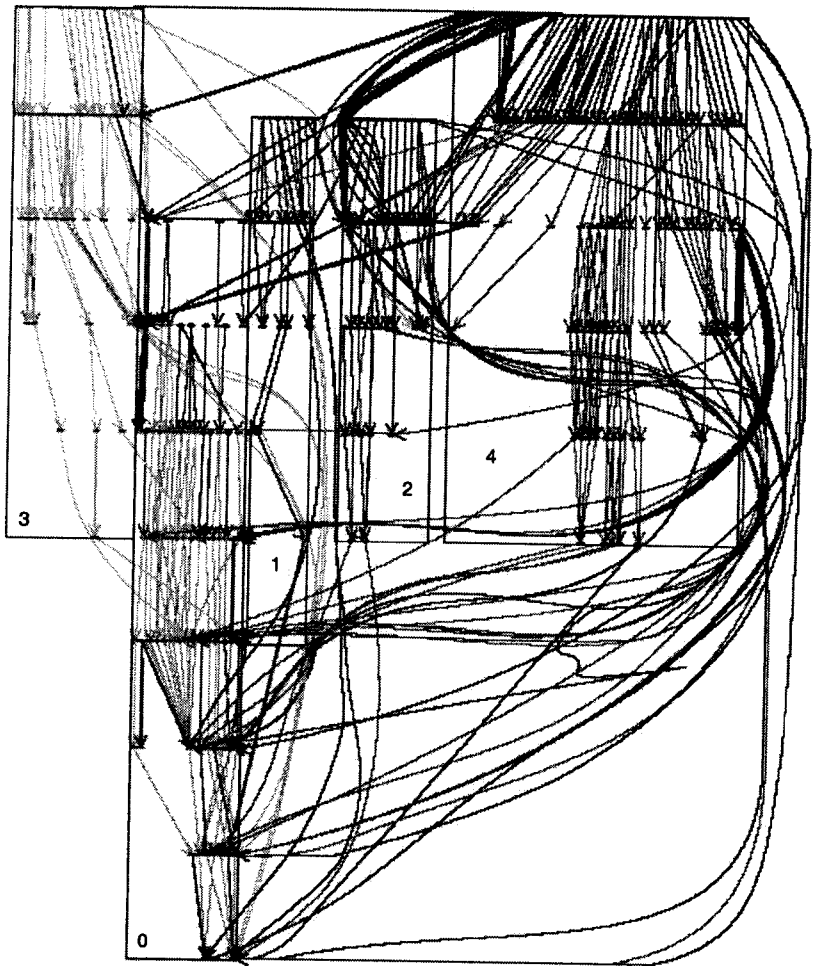


Fig. 3. Diagram showing essentially the same information as Fig. 2 except that important descriptive GO terms are more visible. See text for description of graph

Table 4. Principal cluster descriptions for the genes clustered with the MBSAS algorithm derived as stated in the text. The last column gives the number of genes in the cluster associated with the term.

GO ID	GO Term	Number of Genes
Cluster 0 — 6 genes		
	20 GO terms but each associated with only one gene	1
Cluster 1 — 2 genes		
GO:0008092	cytoskeletal protein binding activity	2
GO:0007028	cytoplasm organization and biogenesis	2
GO:0003774	motor activity	2
GO:0005875	microtubule associated complex	2
	5 GO terms but each associated with only one gene	1
Cluster 2 — 6 genes		
GO:0004871	signal transducer activity	4
GO:0007154	cell communication	4
GO:0005887	integral to plasma membrane	3
GO:0005886	plasma membrane	3
GO:0005194	cell adhesion molecule activity	2
	11 GO terms but each associated with only one gene	1
Cluster 3 — 9 genes		
GO:0030528	transcription regulator activity	4
GO:0008134	transcription factor binding activity	3
GO:0006366	transcription from Pol II promoter	3
GO:0003700	transcription factor activity	3
GO:0006357	regulation of transcription from Pol II promoter	3
	5 GO terms but each associated with only two genes each	2
	13 GO terms but each associated with only one gene	1
Cluster 4 — 20 genes		
GO:0003723	RNA binding activity	10
GO:0030529	ribonucleoprotein complex	9
GO:0009059	macromolecule biosynthesis	9
GO:0006412	protein biosynthesis	9
GO:0005829	cytosol	9
GO:0003735	structural constituent of ribosome	8
	2 GO terms but each associated with only four genes each	4
	5 GO terms but each associated with only three genes each	3
	1 GO term associated with only two genes	2
	33 GO terms but each associated with only one gene	1

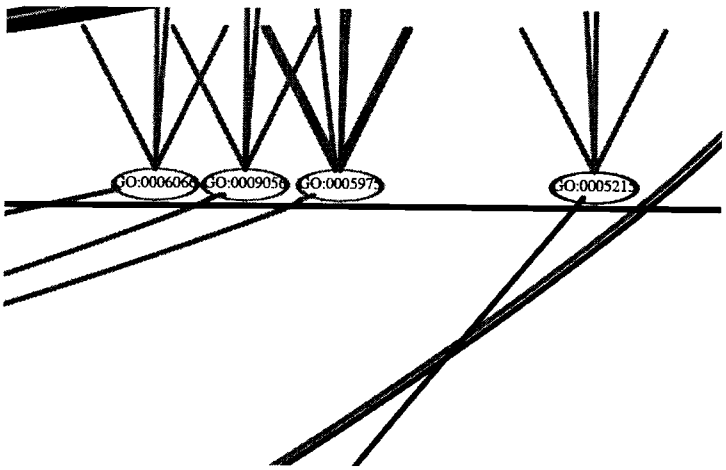


Fig. 4. Diagram showing a close up of the most general GO terms in the large cluster. See text for further description

Another consideration with the possibility of clusters being overly large is that the value of c, the "constant of gravity", might be too large for this dataset. We plan to examine the consequences of lower values of this parameter.

It is also instructive to understand how clusters are different. In a similar way to that described for finding descriptions of clusters, we can build a list of terms that are shared by one other cluster (at their most general level possible). This tells us how two clusters are similar, but different to other clusters. It is essentially an ontological measure of the distance between clusters. The same sort of algorithm could be used for different groupings of clusters. However, an explosion of computational complexity soon occurs.

6 Future Work

Future work may be categorised into four areas: cluster validation, cluster refinement, experimentation with other algorithms and integration of feedback from domain experts.

Validation of the clustering algorithm and the resultant clusters is required to ensure that the clusters describe anything worthwhile. We plan to validate the clustering in three ways: (i) hand choose a set of genes for known GO relationships and then determine whether the clustering algorithm infers at least those relationships; (ii) examine the effect of different sets of q and Θ parameters (as well as the other two parameters) with the aim of seeing whether clusters break up and combine smoothly; and (iii) compare the results of our clustering algorithm with other similar systems.

The clustering algorithm will be refined in the following two ways: (i) the stability of clusters needs to be analysed with respect to the order of presentation of data; and (ii) choice of parameter values requires more understanding.

Different clustering algorithms will be tried. MBSAS was simply a starting point. At least k–means and hierarchical clustering algorithms will be attempted.

The clustering behaviour must be refined based on feedback from medical experts who understand the different genes and will be able to determine whether the clustering increases their understanding of the genes. Cluster analysis like this project is, in some ways, an exercise in prototyping. Once the domain experts gain some knowledge they are able to ask other questions.

7 Conclusions

This paper describes a technique for clustering genes according to their functionality as defined by associated terms in the Gene Ontology. The clustering algorithm is notable for considering the relationships between terms by traversing the ontology.

The Gene Ontology is used to visualise the clusters by automatically building cluster descriptions. Preliminary results clustering genes give insights into the clusters and the efficacy of the clustering algorithm.

Acknowledgements

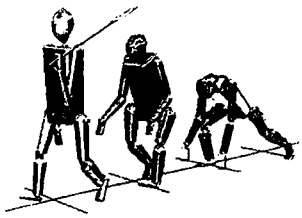
We would like to thank the other members of our group, Daniel Catchpoole and David Skillicorn for their assistance developing the ideas in this paper and for their analysis and gathering of the data and Andre Skusa and Jacob Koehler for their comments and discussion on drafts of the paper.

References

1. Fayyad, U.M., Piatetsky-Shapiro, G., et al.: From data mining to knowledge discovery in databases. AI Magazine **17** (1996) 37–54
2. Berthold, M., Hand, D.J., eds.: Intelligent Data Analysis. Springer, Heidelberg (2003)
3. Hand, D., Mannila, H., et al.: Principles of Data Mining. The MIT Press, Cambridge, MA (2001)
4. Han, J.: How can data mining help bio–data analysis. In: Proceedings 2nd Workshop on Data Mining in Bioinformatics BIOKDD02, in conjunction with ACM SIGKDD 8th International Conference on Knowledge Discovery and Data Mining, Edmonton, Canada, ACM Press (2002)
5. Parmigiani, G., Garrett, E.S., et al.: The analysis of gene expression data: An overview of methods and software. In Parmigiani, G., Garrett, E.S., Irizarry, R.A., Zeger, S.L., eds.: The analysis of gene expression data, Heidelberg, Springer–Verlag (2003) 1–45
6. Rosenwald, A., Wright, G., et al.: The use of molecular profiling to predict survival after chemotherapy for diffuse large–B–cell lymphoma. New England Journal of Medicine **346** (2002) 1937–1947

7. Hastie, T., Tibshirani, R., et al.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer–Verlag, Heidelberg (2001)
8. The Gene Ontology Consortium: Gene Ontology: tool for the unification of biology. Nature Genetics **25** (2000) 25–29 PubMed ID:10802651.
9. Gene Ontology Consortium: Gene Ontology Consortium. Available on: `http://www.geneontology.org` (2003) Viewed at 15 October 2003.
10. Norwegian University of Science and Technology: eGOn (explore Gene Ontology). Available on: `http://nova2.idi.ntnu.no/egon/` (2003) Viewed at 23 October 2003.
11. Al-Shahrour, F., Díaz-Uriarte, R., Dopazo, J.: FatiGO. Available on: `http://fatigo.bioinfo.cnio.es/` (2003) Viewed at 23 October 2003.
12. Zeeberg, B.R., Feng, W., Wang, G., Wang, M.D., Fojo, A.T., Sunshine, M., Narasimhan, S., Kane, D.W., Reinhold, W.C., Lababidi, S., Bussey, K.J., Riss, J., Barrett, J., Weinstein, J.N.: GOMiner: A resource for biological interpretation of genomic and proteomic data. Genome Biology **4** (2003)
13. Zeeberg, B.R., Feng, W., Wang, G., Wang, M.D., Fojo, A.T., Sunshine, M., Narasimhan, S., Kane, D.W., Reinhold, W.C., Lababidi, S., Bussey, K.J., Riss, J., Barrett, J., Weinstein, J.N.: GOMiner. Available on: `http://discover.nci.nih.gov/gominer/` (2003) Viewed at 23 October 2003.
14. Hosack, D.A., Dennis Jr., G., Sherman, B.T., Lane, H., Lempicki, R.A.: EASE. Available on: `http://david.niaid.nih.gov/david/ease.htm` (2003) Viewed at 23 October 2003.
15. Hosack, D.A., Dennis Jr., G., Sherman, B.T., Lane, H., Lempicki, R.A.: Identifying biological themes within lists of genes with EASE. Genome Biology **4** (2003)
16. Han, J., Fu, Y.: Discovery of multiple–level association rules from large databases. In: Proceedings 1995 International Conference on Very Large Data Bases. (1995) 420–431
17. Srikant, R., Agrawal, R.: Mining generalized association rules. In: Proceedings 1995 International Conference on Very Large Data Bases. (1995) 407–419
18. Diehn, M., Sherlock, G., Binkley, G., Jin, H., Matese, J.C., Hernandez-Boussard, T., Rees, C.A., Cherry, J., Botstein, D., Brown, P.O., Alizadeh, A.A.: SOURCE: a unified genomic resource of functional annotations, ontologies, and gene expression data. Nucleic Acids Research **31** (2003) 219–223
19. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. Academic Press, San Diego, USA (1999)
20. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Second edn. John Wiley and Sons, New York (2001)

ADM03



Proceedings
Australasian Data Mining Workshop

8th December, 2003, Canberra, Australia

Edited by
Simeon J. Simoff, Graham J. Williams and
Markus Hegland

---

in conjunction with
The 2003 Congress on
Evolutionary Computation
Canberra – Australia,
8th – 12th December, 2003

University of Technology Sydney
2003

Proceedings of the 2nd Australasian Data Mining Workshop – ADM03, in conjunction with the 2003 Congress on Evolutionary Computation, 8th – 12th December, 2003, Canberra, Australia.

S. J. Simoff, G. J. Williams and M. Hegland (eds).

Workshop Web Site:
http://datamining.csiro.au/adm03/

Published by the University of Technology Sydney

ISBN 0-9751724-1-7