Investigating “Gene Ontology”- based semantic similarity in the context of functional genomics

Danielle Welter

This thesis is submitted in partial fulfilment of the requirement for the degree of Doctor of Philosophy

School of Computer Science and Informatics

Cardiff University

May 2011

DECLARATION

This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed ..... Date .....

STATEMENT 1

This thesis is being submitted in partial fulfilment of the requirements for the degree of PhD.

Signed ..... Date .....

STATEMENT 2

This thesis is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by explicit references.

Signed ..... Date .....

STATEMENT 3

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ..... Date .....

Abstract

Gene functional annotations are an essential part of knowledge discovery in the analysis of large datasets, with the Gene Ontology [Ashburner et al., 2000] as the *de facto* standard for such annotations. A considerable number of approaches for quantifying functional similarity between gene products based on the semantic similarity between their annotations have been developed, but little guidance exists as to which of these measures are the most appropriate for different purposes. This was addressed here by comparing the performances of a number of similarity measures and associated parameters. This comparison provided some interesting new insights as well as confirming emerging trends from the literature.

There is also a pressing need for novel ways of applying these measures to facilitate the functional analysis of lists of gene products. We developed a novel algorithm, FuSiGroups, to group GO terms based on their semantic similarity and genes based on their functional similarity. This two-fold grouping results in groups of not only functionally similar genes but also an associated set of related GO terms that characterise a single functional aspect relating the genes in the group, which facilitates analysis by creating more coherent groups. Each gene can belong to multiple groups, so the groups more accurately reflect the complexity of biological reality than clusters generated using traditional approaches.

FuSiGroups was tested on a number of scenarios and in each case, successfully generated biologically relevant groups, identifying the key functional aspects of the dataset. The algorithm also managed to eliminate genes that were functionally unrelated to the bulk of the dataset and distinguish between different biological pathways. Although dataset size is currently a limiting factor, with smaller datasets performing the best, FuSiGroups has been demonstrated as a promising approach for the functional analysis of gene products.

Acknowledgements

"No man is an island", wrote John Donne in 1624[1]. While some describe doing a PhD as "the loneliest task in the world", nothing could be further from the truth. I could not have finished this PhD without the help and support of a great many people, too many in fact to name them all here without making this thesis an even heftier tome. The fact that I only thank a few by name shall however be no indication that my gratitude to those who remain unnamed is any less great.

First of all, I would like to thank my supervisors, Prof W.A. Gray and Dr P. Kille, who, with patience and experience, guided me throughout my project. Through countless discussions, they helped me further my own knowledge and understanding of my work by sharing theirs with me. They taught, guided and challenged, allowing me to make my own mistakes but never stray too far from the right path. I am forever in their debt.

My thanks also go to the "Fonds National de la Recherche" (FNR) in Luxembourg, who funded my PhD.

I owe a great deal of gratitude of my fellow PhD students at the Cardiff School of Computer Science and Informatics, in particular Alysia. They made my years here a very enjoyable experience, both academically and socially.

No words can describe how infinitely grateful I am to my parents and my sister, for their love and never-ending support. My family's unwavering faith in my ability to complete this project always encouraged me to keep going. For this and many many other things, I thank them!

Last but by no means least, my thanks go to Ian, the most important person in my life. With love and patience, he stood by me through all the ups and downs of my project, putting up with my craziness, supporting me and encouraging me. I could not have done this without him by my side.

[1]Meditation XVII in *Devotions upon Emergent Occasions*, John Donne, 1624

Contents

List of Figures

List of Tables

Glossary

AUC	Area Under the Curve, a performance index for ROC curves. See Section 3.2.1
AVG	Average, an approach for calculating functional similarity between gene products. See Section 2.3.2
BMA	Best Match Average, an approach for calculating functional similarity between gene products. See Section 2.3.2
BP	Biological Process, a branch of the Gene Ontology. See Section 2.1.1
CC	Cellular Component, a branch of the Gene Ontology. See Section 2.1.1
DAVID	Database for Annotation, Visualization and Integrated Discovery, a free online bioinformatics resource [Dennis et al., 2003]
EBI	European Bioinformatics Institute
FT	Functional Threshold, a parameter of the FuSiGroups algorithm. See Section 3.1.6
GO	Gene Ontology. See Section 2.1
GOA	The Gene Ontology Annotation database [Camon et al., 2004]
GraSM	GRAph-based Similarity Measure, a semantic similarity method that considers all disjunctive ancestors of two ontology terms [Couto et al., 2005]
IC	Information Content, a way of quantifying the information conveyed by a concept. See Section 2.2.1
IDA	Inferred from Direct Assay, a GO evidence code. See Section 2.1.2
IDF	Inverse Document Frequency, a weighting factor used in information retrieval. See Section 3.2.1
IEA	Inferred from Electronic Annotation, a GO evidence code. See Section 2.1.2
IGI	Inferred from Genetic Interaction, a GO evidence code. See Section 2.1.2
IMP	Inferred from Mutant Phenotype, a GO evidence code. See Section 2.1.2
InterPro	Database of protein families, domains and functional sites that can be used for the functional characterisation of proteins [Mulder et al., 2003]
KEGG	Kyoto Encyclopedia of Genes and Genomes [Kanehisa and Goto, 2000]
LCA	Lowest Common Ancestor. See Section 2.2.1
MAX	Maximum, an approach for calculating functional similarity between gene products. See Section 2.3.2
Meaningful groups	FusiGroups groups which contain four or more gene products. See Section 3.1.6
MF	Molecular Function, a branch of the Gene Ontology. See Section 2.1.1
MICA	Most Informative Common Ancestor. See Section 2.2.1

NTO	Normalised Term Overlap [Mistry and Pavlidis, 2008]. See Section 2.3.1
ORF	Open Reading Frame, a DNA sequence that does not contain a stop codon [Alberts et al., 2002]
PPI	Protein-Protein Interaction, the binding together of two or more proteins as part of their biological function. See Section 3.2.1
R	A free software environment for statistical computing and graphics [R Development Core Team, 2010]
RCA	Inferred from Review Computational Analysis, a GO evidence code. See Section 2.1.2
ROC curve	Receiver Operating Characteristic curve, an approach for modelling the trade-off between sensitivity and specificity of a binary classification system. See Section 3.2.1
SGD	Saccharomyces Genome Database [Cherry et al., 1998]
SV	Semantic Value, one component of the semantic similarity measure by Wang et al. [2007]. See Section 2.2.3
ST	Semantic Threshold, a parameter of the FuSiGroups algorithm. See Section 3.1.6
TAS	Traceable Author Statement, a GO evidence code. See Section 2.1.2
TO	Term Overlap [Mistry and Pavlidis, 2008]. See Section 2.3.1
UniProtKB/ Swiss-Prot	A high-quality, manually curated protein sequence database [The UniProt Consortium, 2008]

Chapter 1

Introduction

Over the last half-century, there has been a tremendous evolution in the way that gene function is studied. In the early days of molecular genetics, the elucidation of gene function was primarily reliant on characterising mutant phenotypes, with studies targeting individual genes, or a small number of related genes, at a time. With the advent of DNA sequencing techniques, new approaches for evaluating and relating gene function were required as the number of known genes grew so quickly that manual study alone was no longer practical. From Sanger sequencing [Sanger et al., 1977b], the approach by which the first full DNA-based genome, bacteriophage ΦX174 [Sanger et al., 1977a], was sequenced, to modern day high-throughput or "next-generation" sequencing strategies (see Shendure and Ji [2008] for a review), which allow the sequencing of whole genomes in a matter of days, the range and speed of DNA sequencing is ever increasing, and with it, the amount of genomic data available. There are now over a hundred sequenced eukaryotic genomes, around half of which are vertebrate genomes [Flicek et al., 2011], while the full genomes of over a thousand prokaryotes are available [Lagesen et al., 2010].

This wealth of genomic data brought with it a range of new challenges regarding its storage, maintenance and exploitation. From the need to address these problems emerged the new multidisciplinary field of bioinformatics, merging aspects of molecular biology, computer science and information technology. While bioinformatics is concerned not just with genomics but with all computational aspects of molecular biology and biochemistry, including proteomics, systems biology and evolutionary modelling, the assembly and annotation of genomes remains an important part of the field.

One of the great challenges in functional genomics is the concurrent analysis of large amounts of data to identify groups of functionally related genes. Much of the

available information is in a format suitable for human rather than computational processing, but the volume of data, thousands and thousands of gene products, makes human analysis highly impractical. Over the last decade and a half, great efforts have been made to transform human readable information into information that can be processed by a computer, as well as finding new ways of exploiting this data to transform information content into knowledge.

Ontologies were one way of representing existing knowledge in a structured format that quickly received a lot of attention [Stevens et al., 2000] and a number of different formats and domains of molecular biology were explored. One project that particularly stands out is the Gene Ontology (GO) [Ashburner et al., 2000] (see Section 2.1), a structured, unified and species-independent vocabulary of molecular functions, cellular components and biological processes. Over the last few years, the GO has become the *de facto* standard for functional gene annotations.

The availability of this kind of structured information that is accessible to both humans and computers in turn gave rise to efforts to exploit this information in novel ways. Lord et al. [2003a] first proposed the use of semantic similarity to compare gene products[1] on the basis of existing knowledge, rather than their biological properties like sequence and expression profiles. Since then, a number of approaches to calculate the semantic similarity between ontological terms and, by extension, the functional similarity between the entities (genes and gene products) the terms annotate, have either been adopted from other fields, such as natural language processing or been developed specifically for the GO. Other efforts have explored different applications for these similarity measures.

One area of particular interest is the grouping of gene products based on functional similarity. In many cases, functional annotation is used to improve clustering based on gene expression similarity by combining expression and functional similarities or using functional similarity for non-random cluster seeding. The disadvantage of most clustering approaches is that they allow each gene product to be a member of only one cluster. This rigid classification is unable to fully reflect the complexity of biological reality in which each gene product can have several different functions and be part of a number of processes, often in different parts of the cell [Khatri and Drăghici, 2005; Tari et al., 2009]. A more flexible form of grouping could address this issue.

Additionally, it is possible for gene products to be functionally similar in the

[1]The term gene product can refer to both proteins and RNA. In the dataset used in this work, all gene products are proteins. Additionally, the terms gene product and gene may occasionally be used interchangeably, which is appropriate in the context of the identifiers used here.

absence of any other form of quantifiable biological similarity, so exploring functional grouping in its own right is also of interest. In fact, Romero-Zaliz et al. [2008] argue that incorporating prior knowledge in expression clustering may lead to bias in the analysis from incomplete or unevenly specific annotations and that it may be more appropriate to use functional annotation for independent validation of clustering results instead.

Even in the absence of the argument for analysis bias, the independent exploration of functional similarity among a set of gene products can be of interest in many scenarios. Determining the common functional aspects in a list of genes derived from a large-scale experiment is usually the first step to the interpretation of the biological significance of the experimental results. Basic approaches such as statistical over-representation of annotation terms, although commonly used, are unable to capture the richness and complexity of the relationships among these terms, as expressed in the ontological structure [Grossmann et al., 2007]. This explains the need for the more computationally intensive but more sophisticated semantic similarity approaches, which are able to capture the ontological relationships between annotation terms. The functional similarity between gene products, derived from the semantic similarity between GO terms, can then be used not only to characterise the functional relationships between these gene products but also to judge the impact of different annotations within each group of related gene products.

Probably the most comprehensive tool for this type of analysis currently in existence is DAVID (Database for Annotation, Visualization and Integrated Discovery) [Huang et al., 2007], a tool that allows the functional classification of either genes or annotations in so-called "biological modules". However even the dimensional reduction from the many-to-many relationships between genes and their annotations provided by DAVID can still be insufficient. DAVID groups represent either functionally related genes or annotations, depending on which aspect of the classification tool is used, but never both. For a given group of related genes, all annotation terms associated with any of the genes in the group are also associated with that group, and there is a ranking system to indicate their level of relatedness to the group. Considering the number of heterogeneous data sources, rather than just the Gene Ontology, available for DAVID functional analysis, this can still result in a complex and time-consuming analysis to elucidate the key functional aspects that link a set of groups. It might therefore be more desirable to be able to consider both related genes and annotation terms at the same time.

1.1 Problem definition

1.1.1 Research question

The novel contribution to the field provided by this investigation is twofold. Firstly, the performances of several GO-specific semantic similarity approaches (Schlicker et al. [2006], Wang et al. [2007], Couto et al. [2005]) will be compared to and validated against each other on the same dataset. So far each of these approaches has only been compared to the three methods by Resnik [1995], Lin [1998] and Jiang and Conrath [1997], all of which were developed in a natural language context rather than for the Gene Ontology, which means they are being compared with algorithms designed for a different type of data source. Different functional similarity approaches that combine the semantic similarities between GO terms into a single score to characterise the similarity between gene products based on their GO term annotations, as well as several associated parameters such as type of annotations, will also be included in the comparison.

Secondly, we will develop and refine a new grouping algorithm, FuSiGroups, based on semantic similarity derived from GO annotations, as well as functional similarity between gene products, to identify groups of functionally similar gene products. It is expected that groups created using this algorithm will reveal additional functional relationships (i.e. similar molecular functions or belonging to a common biological pathway) between genes that cannot be detected using only traditional approaches such as gene expression similarity. In addition, these groups reduce the complexity of the many-to-many relationships between GO terms and gene products into clearly defined groups reflecting a single functional aspect of the genes they contain. This dimensional reduction of the data will allow easier analysis without loss of information.

The algorithm's output will be evaluated against clusters obtained by standard hierarchical clustering using semantic similarity as the similarity metric. The semantic groups and clusters will be compared to clusters obtained from gene expression studies to evaluate the relationship between functional (semantic) similarity and gene expression similarity.

The dataset used is the well-studied *Saccharomyces cerevisiae* gene expression dataset by Eisen et al. [1998]. The metrics used will be those proposed by Wang et al., Schlicker et al., Resnik and Lin, as well as Couto et al.'s disjunctive ancestor approach, which will be applied to the Schlicker, Resnik and Lin algorithms[2].

[2]For simplicity, the approaches considered in this study will from now on be referred to by the

1.1.2 Aims and objectives

Based on the research hypothesis in Section 1.1.1, this project has two overall objectives. Firstly, a number of semantic and functional similarity approaches and associated parameters will be compared. Secondly, a novel grouping algorithm based on semantic similarity will be developed and evaluated.

With regards to the first objective, the research question only explicitly states the different semantic similarity measures that will be compared. However, as will be discussed in more detail Chapter 2, semantic similarity is the similarity between two GO terms. As gene products are generally annotated with more than one GO term, the functional similarity between two gene products is a combination of the semantic similarities of the GO terms that annotate them. There are a number of ways to combine semantic similarity scores and there is currently no consensus in the literature as to which is the best or most appropriate method.

In fact, in many published studies making use of semantic and functional similarities, similarity measures and associated parameters are chosen without any real justification as to whether these choices are the most appropriate. Resnik's similarity measure might be used because it was the first measure to be applied in the GO, or Lin's measure because it is bounded between 0 and 1, even though both measures have known drawbacks (see Section 2.2) and without any reference to a study demonstrating that one measure or another is the most appropriate in a given context. The same applies to the different approaches of combining semantic similarity scores into a functional similarity score. Therefore a number of semantic similarity approaches and two functional similarity scores will also be studied here.

As will be discussed in the next two chapters, there are a number of additional factors that need to be considered in the calculation of semantic and functional similarities and on which there is no consensus. They are the combination or lack thereof of ontological scores, the choice of ancestors in the similarity calculation and the type of annotations. Briefly, the GO consists of three parallel ontologies covering the areas of biological process, cellular component and molecular function. Some works base the functional similarity between gene products on only one of these ontologies while others combine their results into one score.

Most of the semantic similarity approaches compared here consider in some way the ancestor or ancestors common to two GO terms in the hierarchical structure of the ontology. While the detailed meaning of this will be discussed in the next chapter, it should be noted here that the most common approach is to consider only

(primary) author's name, e.g. Schlicker's approach instead of Schlicker et al. [2006]'s approach.

the most detailed ancestor of two terms. There is some disagreement in the field as to whether ignoring all but the most detailed ancestor leads to a significant loss of information, which is why Couto et al. proposed an alternative approach that takes into consideration multiple ancestors. These two approaches, single and multiple ancestors, will be compared.

Thirdly, annotations of gene products with GO terms are detailed with an evidence code to indicate how the annotation was derived, for example from direct experimental evidence or inferred through similarity analysis. Specifically, one evidence code, IEA, which refers to uncurated, electronically derived annotation, is often excluded from similarity calculations, even though the vast majority of GO annotations are of this kind and despite a growing body of evidence that the resulting loss of annotation richness may very well outweigh any improvement in annotation precision. There also appears to often be a misplaced confidence in the accuracy of manually curated annotations, despite the GO's express warning that annotation codes should not be used as an indicator of annotation quality. The effect of including and excluding electronic annotation on semantic and functional similarity will be compared.

The second major objective of this work is the development of a grouping algorithm that makes use of both semantic and functional similarity to group together functionally similar gene products by the specific functional aspects that they share. This approach differs from standard clustering in two respects. Firstly, most traditional clustering approaches confine each gene product to a single cluster, which is a strong over-simplification of biological reality, where each gene product can have multiple functions and be part of several processes. Secondly, clustering is generally based on only one form of similarity, so while functionally similar gene products will be clustered together, it is still up to the user to determine which functional aspect or aspects they share by considering all their annotations. Even the most similar existing approach, DAVID [Huang et al., 2007], only provides either groups of gene products or groups of annotation terms. For each of DAVID's clusters, all annotations (for groups of gene products) or gene products (for groups of annotations) associated with that group are given to the user, with only a score to indicate how closely related they are to the group.

Our algorithm will address both these issues by generating groups that each represent one functional aspect common to the gene products in this group and allowing each gene product to be grouped into any group reflecting its various functional aspects. This way, there may be several groups with the same gene content, indicating that these genes share a number of functional aspects, or groups with par-

tial content overlap, showing how the non-overlapping genes from the two groups are functionally distinct. The groups also reflect the complex relationships that link gene products across different processes and different parts of the cell. The new perspective provided by the FuSiGroups algorithm is expected to lead to interesting new insights into these functional relationships.

For a given dataset, the algorithm is expected to generate biologically relevant groups and identify the main functional aspects common to the genes in that dataset. Additionally, it is expected to eliminate gene products that are functionally unrelated to the majority of the dataset by not including them in any of the groups. Through comparison with gene expression clustering, it will be determined whether functional grouping reflects gene expression similarity. The algorithm's ability to detect other forms of biological similarity, such as pathway membership, will also be tested. A number of different datasets will be used in the evaluation in order to comprehensively address these different objectives of the algorithm and demonstrate its potential in a range of situations.

1.2 Contributions to knowledge

The contributions to knowledge of this work can be divided into two broad categories, namely the evaluation of similarity measures and the FuSiGroups algorithm. More specifically, they are:

- Objective evaluation of a number of semantic and functional similarity measures and associated parameters
  - A number of semantic and functional similarity measures and associated parameters were evaluated against each other.
  - Three different types of biological similarity were used in order to reduce risk of bias of annotation similarity for or against one type of biological similarity.
  - Although no definitive answer as to which measures and parameters are "the best" - an answer that is unlikely to even exist in a field as diverse and complex as functional annotation - a set of recommendations were produced that can provide further guidance for researchers considering which approach would be most appropriate for their own work.
- Design and testing of FuSiGroups, a novel grouping algorithm
  - The algorithm groups both related GO terms and related gene products, resulting in functionally coherent groups reflecting a single functional aspect that relates the genes in the group. This is an optimisation over

current approaches, which only provide either groups of gene products or groups of annotation terms.

- FuSiGroups groups provide a dimensional reduction from the complex and diverse gene to GO term associations of functional annotations, but without the loss of precision that might be expected from such a reduction, as all the original associations are still present. Similarities and differences between groups can easily be visualised
- The algorithm successfully identified the main functional aspects of a number of datasets. It also identified by exclusion a number of random genes unassociated with the datasets. The algorithm is sufficiently sensitive to distinguish between two unrelated pathways but not between closely related clusters of gene products.

1.3 Thesis disposition

The rest of this thesis is organised as follows:

Chapter 2 covers the background to the work covered in this thesis. It gives a brief introduction to the Gene Ontology, then provides a survey of semantic and functional similarity approaches used in conjunction with the GO. This also covers the kind of applications in which these measures are used, including existing implementations.

Chapter 3 describes the different elements that are considered in this work, including the selection of semantic and functional similarity approaches and other parameters. The FuSiGroups grouping algorithm is described. The chapter also covers in detail the evaluation strategy that will be used for the different elements of the project. Finally, a number of implementation considerations will be discussed.

Chapter 4 discusses the results of an experimental comparison of the different semantic and functional similarity approaches and associated parameters. Conclusions are drawn as to which measures and parameters perform the best and will be carried forward for use in the rest of the thesis.

Chapter 5 shows how the semantic and functional thresholds for the FuSiGroups algorithm are derived. This includes a discussion of the difficulty in determining the semantic thresholds, as well as how the derived thresholds for the different approaches compare in the context of the distributions of similarity values for each approach.

Chapter 6 discusses the overall trends of the FuSiGroups results for the measures and parameters selected in Chapter 4 and at the different thresholds derived in Chapter 5. In particular, the grouping results are analysed for any noticeable bias in the algorithm, such as a favouring of deeper or shallower annotations, or disproportionate number of groups for a given ontological aspect compared to the number of annotations of that type. Conclusions are then drawn regarding the performance of the different measures with the FuSiGroups algorithm.

Chapter 7 looks in more detail at the FuSiGroups results for the full Eisen dataset for one combination of approaches and thresholds in order to establish whether the key functional aspects of the dataset have been identified. An initial limitation of the algorithm was found in an overly high level of overlap in content and definitions between many groups. This is addressed through the introduction of the concept of "supergroups". The comparison of functional groups, semantic clusters and expression clusters is discussed.

Chapter 8 considers the detailed grouping results of three smaller datasets to more directly address the different scenarios in which FuSiGroups is useful. These include the identification of the main functional aspects of each dataset, the elimination of unrelated gene products and the ability to reflect other forms of biological similarity. The potential of the algorithm to accurately capture complex biological relationships, as well as its limitations are successfully identified.

Chapter 9 draws the previous five chapters together and discusses the overall implications of the results and the potential of the algorithm. Recommendations on the use of semantic and functional similarity approaches are given, and an analysis pathway for FuSiGroups results is detailed. The chapter also provides an outlook on future work, then draws the final conclusions on the entire work.

Chapter 2

Semantic and functional similarity

Semantics is the study of meaning or, more precisely, "(the study or analysis of) the relationships between linguistic symbols and their meanings" [Oxford English Dictionary, 1989]. Consequently, comparing two terms semantically means comparing their meaning or "knowledge content", rather than comparing the two terms themselves [Lord et al., 2003a].

In order to compare two concepts semantically, one needs a frame of reference in which to do the comparison. A good example of such a context is a hierarchical structure like a taxonomy or an ontology in which the concepts can be represented as the nodes of a tree and the relationships between them as the edges that link the nodes. Over the last twenty years, a number of ways to quantify semantic similarity have been developed. Some of these consider the edges of a hierarchy (e.g. distance-based measures), while others consider its nodes (e.g. information content-based measures) [Lin, 1998].

In the context of this work, the term "semantic similarity" refers to the similarity between ontological concepts, such as Gene Ontology (GO) terms. Gene products are annotated with GO terms (described in Section 2.1.2). Using the semantic similarity between these annotation GO terms, similarity between the annotated gene products can also be quantified. This is referred to as "functional similarity". We differentiate between semantic and functional similarity as the former is a quantification of the relationship between ontological concepts based on ontological structure, whereas the latter is the quantification of the relationship between gene products based on their annotation. This quantification is often but not always based on semantic similarity.

The following sections give an overview of the Gene Ontology and the semantic and functional similarity approaches most commonly used in conjunction with the

GO. Approaches developed outside this purview and only applied in different contexts are not considered. The field of semantic similarity has expanded greatly in the last few years, for example in conjunction with semantic web and document retrieval, and giving an overview of every aspect of the field would exceed the present scope.

The approaches described here are subdivided using the classification suggested by Pesquita et al. [2009] in their recent review of the field. Tables 2.1, 2.2 and 2.3 are adapted from this paper.

2.1 The Gene Ontology

The Gene Ontology (GO) [Ashburner et al., 2000] was created in 1998 by the Gene Ontology Consortium in an effort to address the need for a controlled, structured and unified vocabulary for genome annotation. The Gene Ontology Consortium is a collaborative project whose founding members are the model organism databases Flybase [Tweedie et al., 2009], Mouse Genome Informatics (MGI) [Blake et al., 2011] and the Saccharomyces Genome Database (SGD) [Cherry et al., 1998]. In the last ten years, the list of member projects has more than quintupled and now includes, among others, dictyBase [Fey et al., 2009], Gene Ontology Annotation @ EBI (GOA) [Barrell et al., 2009], Gramene [Jaiswal et al., 2006], Rat Genome Database (RGD) [Twigger et al., 2006], Reactome [Croft et al., 2011], The Arabidopsis Information Resource (TAIR) [Swarbreck et al., 2008], WormBase [Harris et al., 2010] and Zebrafish Information Network (ZFIN) [Bradford et al., 2011].

In addition, the GO consortium has a number of "associates". The distinction between member and associate lies primarily in the level of the contribution to the GO, as well as direct funding for GO-related activities.

While GO is by no means the only project of this nature, it is probably the most comprehensive resource in existence to date, and has been adopted as a key source of genome annotation by the scientific community.

2.1.1 Structure of the Gene Ontology

The GO consists of three orthogonal structured vocabularies or "sub-ontologies", namely:

- molecular function (MF), i.e. the activity, at molecular level, of a gene product;
- biological process (BP), i.e. the larger overall process that a gene product is

involved in;

- cellular component (CC), i.e. the component of the cell that a gene product acts in.

Each of these taxonomies is structured as a directed acyclic graph (DAG). This means that any parent term can have multiple children and any child term can have multiple parents, but there can be no circular relationships. The majority of links between terms are of one of two link types, namely "IS_A" and "PART_OF", although the links "REGULATES", "NEGATIVELY_REGULATES" and "POSITIVELY_REGULATES" have been introduced in recent years. For more details on the structure of the GO, see Ashburner et al. [2001].

On April 29th 2011, the GO contained a total of 34086 terms. Of these terms, 20717 are part of the BP-ontology, 2824 part of CC and 9036 part of MF. Not included in these numbers are 1509 obsolete terms. It is worth noting that by far the smallest ontology is that of cellular component. This is unsurprising as the number of distinct cellular locations, even across different types of tissues and different species, is limited compared to the diversity of biological processes. A little more surprising might be the fact that the biological process ontology counts more than twice the number of terms of molecular function. This may be due in part to the current state of knowledge in molecular biology, as it is easier to attribute a certain gene product to a more general pathway than to elucidate its specific function. In addition, biological processes are highly diverse, whereas the functions of gene products remain similar across multiple processes.

2.1.2 Gene Ontology annotation

The GO itself does not contain any species-specific information. Instead, its terms are used to annotate gene products from different species. GO annotations are characterised by evidence codes which indicate the nature of the annotation. These evidence codes are subdivided into two classification, curated and un-curated. There are four kinds of curated classifications[1]:

- experimental evidence codes: annotations based on experimental data cited directly in the literature. Include inferences from direct assay (IDA), inference from mutant phenotype (IMP), inference from expression pattern (IEP) etc.;

[1]For full details of GO evidence codes, see the evidence code guide on the GO website at http://www.geneontology.org/GO.evidence.shtml (accessed 27/04/2011)

- computational analysis evidence codes: annotations inferred from bioinformatics analysis, e.g. sequence or structural similarity (ISS), genomic context (IGC);
- author statement evidence codes: annotations for which there is no direct experimental data cited in the literature but where this information is referenced and can be traced;
- curator statement evidence codes: annotations for which there is no direct evidence but which can be reasonably inferred from indirect evidence, e.g. for a gene product with experimentally verified molecular function "specific RNA polymerase II transcription factor activity" (GO:0003704) but no validated cellular location, a cellular location of "nucleus" (GO:0005634) can reasonably be inferred.

Un-curated annotations (IEA - Inferred from Electronic Annotation) are inferred similarly to computational analysis ones, e.g. from sequence similarity or automated transfers of records from other databases. They are not verified by a human before being added to the database, although in the case of mappings from other databases, the mappings between GO terms and other concepts are often manually curated (see Section 3.1.4 for further details and an example).

It is important to note that evidence codes do not reflect the quality of the annotations. Some studies maintain that only curated annotations should be used in, for example, the context of semantic similarity analysis, whereas other studies have found that inclusion and exclusion of un-curated annotations has no significant effects on semantic similarity.

2.1.3 What the GO is not

GO terms focus exclusively on the three aspects listed above. They do not cover information such as which cell type or body part a gene product is expressed in, or during which development or disease stage it is expressed. Other ontologies have been developed for these purposes. Many of these ontologies can found at the "Open Biomedical Ontologies" website [Smith et al., 2007].

All terms are as species-independent as possible. Certain terms, such as "chloroplast", are necessarily specific to a certain type of organism but still not directly specific to a given species.

As detailed in Ashburner et al. [2001], the GO is neither intended as a mandated standard, nor is its simple existence sufficient for the unification of biological

databases. The success of GO is due to the quality of the contributions from its members, which in turn lead to its adoption as a *de facto* standard. The ontology and GO annotations constantly evolve and become more comprehensive. As the coverage of GO grows, so does the linkage between the different resources that use GO, as the shared nomenclature facilitates the crossing of domain boundaries.

2.2 Similarity between GO terms

Pesquita et al. [2009] distinguish between three types of semantic similarity approaches: edge-based, node-based and hybrid. This classification reflects the primary ontological element used by a given approach to calculate semantic similarity. Both node- and edge-based approaches can subdivided further depending on the way the respective ontological element is used. The semantic similarity approaches presented in this section are summarised in Table 2.1.

In the classification used by Pesquita et al. [2009], it is possible for confusion to arise between the node-based subdivision "depth" and the edge-based subdivision "distance". Both concepts refer to the path between two ontological concepts, in the case of "depth" usually a term and the root node, but while a node-based path counts the number of nodes between the terms, an edge-based path counts the number of edges. The edge-path between two terms should be one element smaller than the equivalent node-path, provided that both end terms are included in the count.

2.2.1 Node-based approaches

Node-based approaches use the information contained in a graph's nodes to quantify the similarity between two terms without taking into consideration the edges that connect the nodes. The majority of node-based approaches use the concept of "information content" (IC), which requires the use of information external to the ontology, in the similarity computation. There are only very few node-based semantic similarity approaches that use only the internal node-structure of the GO, including information derived from node depth and density, in order to compute semantic similarity between concepts.

First introduced by Resnik [1995], the concept of information content is based on the idea that the deeper in the hierarchy a term is, the more informative it is (i.e. the higher its information content) and the closer to the root, the less informative it is. The information content of each term in a hierarchy is calculated through the probability of occurrence of that term in a corpus or body of knowledge, i.e. the

Measure	Approach	Notes
Resnik [1995]	Node-based	IC(MICA)
Lin [1998]	Node-based	IC(MICA) & IC(terms)
Jiang and Conrath [1997]	Node-based	IC(MICA) & IC(terms)
Couto et al. [2005]	Node-based	Disjoint common ancestors
Schlicker et al. [2006]	Node-based	IC(MICA) & IC(terms)
Wu et al. [2005]	Node-based	Largest shared path from LCA to root
Bodenreider et al. [2005]	Node-based	Cosine similarity with IDF weighting
del Pozo et al. [2008]	Node-based	Cosine similarity, then depth of LCA
Herrmann et al. [2009]	Node-based	Corpus-free variant of IC
Chiang et al. [2006]	Node-based	Shared path with IC weighting
Chiang et al. [2008]	unclear	Shortest path and depth of MICA
Rada et al. [1989]	Edge-based	Shared path
Cheng et al. [2004]]	Edge-based	Shared path with depth-based edge weighting factor
Yu et al. [2005]	Edge-based	Shared path and distance to LCA
Wu et al. [2006]	Edge-based	Shared path and distance to leaf nodes and LCA
Jakonienė et al. [2006]	Edge-based	Shared path with weighting based on edge type
Yuan and Zhou [2008]	Edge-based	Shortest path between terms
Wang et al. [2007]	Hybrid	Shared ancestors with edge weighting
Othman et al. [2008]	Hybrid	IC/depth/number of children; distance

Table 2.1: Overview of the semantic similarity approaches presented in Section 2.2

information content of a term c is

$$IC(c) = -ln\, p(c) \tag{2.1}$$

where $p(c)$ is the probability of concept c occurring in the taxonomy.

Concept frequencies in a taxonomy are derived from occurrence frequencies of a concept and its children in a corpus. In his research, Resnik used "WordNet" [Fellbaum, 1998] as the taxonomy and the "Brown Corpus of American English" [Francis and Kucera, 1982] as the corpus. The occurrence of a child term counts towards all the occurrences of all its parents. This is logical as some term β, which is a child of α, occurring in a hierarchy implies that α is occurring as well. This is called the "true path rule" [Ashburner et al., 2001].

The probability of a concept c occurring in a taxonomy is

$$p(c) = \frac{freq(c)}{N} \tag{2.2}$$

where

- $freq(c) = \sum_{n \in concepts(c)} total(n)$
- $concepts(c)$ is the set of concepts that are descendants of c;
- $total(n)$ is the number of occurrences of term n in the corpus;
- N is the total number of terms in the corpus.

The use of occurrence frequencies can be considered a disadvantage of IC-based measures as variations in the underlying corpus lead to changes in similarity results. This makes it difficult to compare results from experiments based on different corpora, such as the annotations of different species and older or newer versions of the data.

In information content-based measures, the link between two ontological terms c_1 and c_2 is established through the ancestor terms they share. As c_1 and c_2 may have more than one common ancestor, the most meaningful of those ancestors is usually considered. This is generally the "first" or "lowest common ancestor" (LCA), and also the ancestor with the smallest $p(c)$ (or largest $-ln\, p(c)$). In Lord et al. [2003a], this is defined as the "probability of the minimum subsumer". Another term for this ancestor is "most informative common ancestor" (MICA) [Pesquita et al., 2008], which is how this concept will be referred to from now on in this thesis. It should be noted that LCA will be distinguished here from MICA insofar that it is theoretically possible for an ancestor term a to be the LCA of two terms but not their MICA,

if the IC of a is lower than that of another ancestor term b, but its distance from the root is greater than or equal to that of b. For this reason, LCA will be used when referring to the distance from the root, while MICA will be used for all IC references.

Similarity between concepts c_1 and c_2 according to Resnik [1995] is given by

$$\text{sim}_{Resnik}(c_1, c_2) = \max_{c \in S(c_1,c_2)} [-ln\, p(c)] \tag{2.3}$$

where $S(c_1, c_2)$ is the set of terms that subsume both c_1 and c_2.

All other IC-based semantic similarity approaches developed after Resnik are variations on the same theme. While Resnik's approach only uses the IC of the MICA to quantify the semantic similarity between two terms, other approaches take into account the IC of the terms whose similarity is calculated as well.

Resnik tested his approach against human similarity judgement data and concluded that it performed "encouragingly well"[Resnik, 1995], and also "significantly better than the traditional edge counting approach"[Resnik, 1995]. The main drawback of Resnik's approach is that it only captures the position of the common ancestor within the hierarchy but not its distance from the query terms. This means that two terms directly connected to their most informative common ancestor would have the same similarity as two other terms with the same MICA but that are several levels removed from it in the hierarchy.

An IC-based approach by Lin [1998] addresses this problem by considering the IC of the query terms as well as that of the common ancestor. Taking into consideration the information content of the terms that are being compared as well as that of their shared parent, this approach defines the similarity between concepts c_1 and c_2 as

$$\text{sim}_{Lin}(c_1, c_2) = \frac{2 \cdot \max_{c \in S(c_1,c_2)}[-ln\, p(c)]}{[-ln\, p(c_1)] + [-ln\, p(c_2)]} \tag{2.4}$$

This approach could be considered as a normalised version of Resnik's approach because Lin's similarity coefficient lies between 0 and 1, unlike Resnik's value, which can vary between 0 and infinity[2] [Resnik, 1995]. Lin used the same test set as Resnik to test his similarity score. He found that his approach led to a marginally higher correlation with human judgements than Resnik's measure [Lin, 1998].

While addressing the drawback of Resnik's method of not reflecting the distance between two terms and their common ancestor, the Lin approach has its own disadvantage in that the similarity is displaced from the graph and does not reflect the

[2]Practically, Resnik's upper limit is $-ln\, \frac{1}{N}$, where N is the total number of terms in the corpus

overall position of the three elements in the hierarchy. This means that two very shallow terms can have the same level of semantic similarity as two very deep terms, provided the two pairs are equally close to their respective common ancestor.

This same problem also applies to the approach by Jiang and Conrath [1997], who combined the elements used in Lin's approach into an IC-based distance measure. The semantic distance between two nodes is the inverse of the semantic similarity. For a measure bounded between 0 and 1, this translates to $similarity = 1 - distance$ [Othman et al., 2008]. Semantic distance according to Jiang and Conrath [1997] however is calculated as

$$\mathrm{dist}_{Jiang}(c_1, c_2) = [-ln\, p(c_1)] + [-ln\, p(c_2)] - 2 \times [-ln\, p(c)] \tag{2.5}$$

This measure therefore ranges from 0 if c_1 and c_2 are identical to $2 \times maxIC$ for two leaf nodes which only have the root of the ontology as a common ancestor. $maxIC$ is the maximum information content for a given ontology, which corresponds to an annotation frequency of 1 as a term with an annotation frequency of 0 would not have any information content, both conceptually and mathematically as $ln\, 0$ is undefined. Jiang and Conrath's semantic distance can be transformed into a similarity measure using

$$\mathrm{sim}_{Jiang}(c_1, c_2) = \frac{1}{\mathrm{dist}_{Jiang}(c_1, c_2) + 1} \tag{2.6}$$

where the addition of one to the distance is necessary to avoid infinity values [Couto et al., 2007]. Alternatively, the semantic distance could be normalised by division with $2 \times maxIC$, which would bring it into the [0,1] range, then the converted to similarity by subtracting it from 1.

In its original form, the Jiang approach was actually a hybrid approach including edge weighting factors whose influence can be controlled by two further weighting factors. Virtually all GO applications of this measure set these parameters to exclude the weighting factors, which reduces the distance measure to the node-based approach described here. For more details on the full measure, see the work by Othman et al. [2008] described in Section 2.2.3.

The validation of Jiang and Conrath's approach used a noun portion of WordNet containing about 60000 nodes. Unlike Resnik and Lin, Jiang and Conrath did not use the entire Brown Corpus of American English to estimate the frequencies of concepts. Instead, they used SemCor [Miller et al., 1993], a subset of around 100 passages from the Brown Corpus. Their results confirmed that Resnik's information content

approach produces better results than Rada et al. [1989]'s edge-based approach. Both methods performed less well than Jiang and Conrath's approach.

In their 2003 Bioinformatics paper, Lord et al. [2003a] proposed to investigate the relationships between gene products using semantic similarity rather than sequence similarity. They considered the three IC-based approaches described so far, although only the Resnik approach was used as it was the simplest of the three. In the same year, the authors also published a conference paper [Lord et al., 2003b] in which all three approaches were compared. These two papers marked the beginning of the use of semantic similarity in the context of the Gene Ontology. Since then, a number of node-based semantic similarity measures have been developed specifically for the Gene Ontology in order to address various drawbacks of the "original three" measures used by Lord et al.

Schlicker et al. [2006] proposed relevance similarity sim_{Rel}, a measure that tackles both Resnik's flaw of disregarding the distance between two terms and their common ancestor and Lin's drawback of being displaced from the graph structure. Using the same information content concept as the other measures so far, relevance similarity is defined as

$$\mathrm{sim}_{Rel}(c_1, c_2) = \left(\frac{2 \cdot \max_{c \in S(c_1,c_2)}[-ln\, p(c)]}{[-ln\, p(c_1)] + [-ln\, p(c_2)]} \right) \cdot (1 - p(c)) \tag{2.7}$$

Couto et al. [2005] argued that considering only the MICA of the query terms ignores important ontological information. They presented GraSM (GRAph-based Similarity Measure), a method that considers all disjunctive ancestors (ancestors that can be reached by at least one distinct path) of the query terms. The IC of all the disjunctive ancestors is averaged and used instead of the MICA's IC in any IC-based approach. GraSM is technically not a semantic similarity measure in its own right but is included here because it is used in conjunction with IC-based approaches.

Taking a different approach than other researchers in the field, Chiang et al. [2006] created an algorithm for their GeneLibrarian tool which computes semantic similarity between GO terms as a sequence alignment measure where the path from a term to the root is the sequence and information content is used to weight each GO term. The same group also proposed another measure [Chiang et al., 2008] for another system, Similar Genes Discovery System (SGDS). This second measure is a function of the length of the shortest path between two terms and the depth of their common ancestor. It is unclear whether this method should be classed as node-based, edge-based or hybrid as the authors give no indication whether they

count the nodes or the edges to determine path length and term depth.

Not all node-based semantic similarity measures make use of information content to quantify the similarity between ontology terms. Using annotation data but not information content, Bodenreider et al. [2005] proposed to compute the similarity between GO terms using cosine similarity [Baeza-Yates and Ribeiro-Neto, 1999] (see Section 3.2.1 for details on cosine similarity) in a vector space model, in which each GO term is represented as a vector of the genes it annotates. The GO term vectors are weighted to balance the effect of genes that are annotated with many GO terms and the weighting method used is inverse document frequency (IDF) (see Section 3.2.1 for details on IDF). The weight for a given GO term is defined as the log of the total number of distinct genes in the database divided by the number of genes annotated to the GO term in question. This is similar to, although not the same as, the concept of information content.

Bodenreider et al. [2005] also used statistical analysis of co-occurrence and association-rule mining to find relations between GO terms. The overall purpose of their study was to find associations between GO terms from different branches of the GO. This cannot be done using most other semantic similarity approaches as these rely on GO structure-related elements such as the common ancestor of two terms

The cosine similarity approach was also taken by del Pozo et al. [2008]. In their work, the similarity between GO terms is effectively calculated twice. First, the similarity between terms is calculated using cosine similarity, based on the GO terms' annotations to InterPro [Mulder et al., 2003] entries. From the resulting similarity matrix of GO terms, a "Functional Tree" is built using spectral clustering [Ng et al., 2001]. The similarity, or rather "Functional Distance" between GO terms is then defined as the height of their LCA in the functional tree. Pesquita et al. [2009] classed this measure as edge-based. Based on the definitions in del Pozo et al. [2008], the approach is presented here as part of the node-based approaches rather than the edge-based ones, since the first level of GO term similarity takes into account only the terms themselves, while the second level is based on a hierarchical clustering tree rather than the GO graph and the "height" concept is derived as part of the clustering algorithm rather than through the counting of edges.

Finally, some approaches use only the internal graph structure of the GO, excluding all external information. One such measure was proposed by Wu et al. [2005]. Although Pesquita et al. [2009] classed this approach as an edge-based approach, the present analysis found no indication that anything other than the nodes of the GO graph were used. The confusion may be due to the language used in the paper, as the similarity between GO terms is calculated based on the "shared path" between

two terms, which usually implies edge-counting. The definition of path in this paper however makes it clear that it is the terms rather than the edges that connect them that are counted. Specifically, Wu et al. [2005] define the similarity between two GO terms c_1 and c_2 as the maximum number of common terms in any path from c_1 to the root and any path from c_2 to the root. This can be rephrased as the maximum number of terms from the LCA of c_1 and c_2 to the root.

Herrmann et al. [2009] proposed a variant of information content which also does not use frequency counts from an external corpus but is based entirely on the structure of the GO. Their measure, precision[3] $pre(c)$ of an ontological concept c, is defined as

$$pre(c) = -\frac{log\,\frac{O_d(c)}{O \cdot O_a(c)}}{log\,O \cdot O_a^{max}} \tag{2.8}$$

where O is the total number of terms in the ontology[4], $O_d(c)$ the number of (distinct) descendant terms of c, $O_a(c)$ the number of ancestor terms of c and O_d^{max} the largest possible number of ancestor terms of any leaf node in the ontology. The similarity between two terms c_1 and c_2 is then defined as the precision of their most precise common ancestor,

$$\mathrm{sim}_{simCT}(c_1, c_2) = \max_{c \in S(c_1,c_2)} pre(c) \tag{2.9}$$

The authors use their precision measure as part of a functional annotation-based clustering algorithm for gene products. The paper does not provide an evaluation of the measure and despite its advantage of being corpus-independent, the measure is not used anywhere else in the literature to date.

2.2.2 Edge-based approaches

As their name implies, edge-based semantic similarity approaches quantify the similarity between two GO terms based on the edges in the graph path from one term to the other. They can be subdivided further into approaches that consider the distance between two terms and approaches that consider the path shared by two terms. Most edge-based semantic similarity approaches applied in the GO use either the terms' shared path to evaluate similarity or a combination of shared path

[3] In the original paper, precision is defined as p. This was changed here to avoid confusion with the probability of occurrence p defined in Equation 2.2

[4] N was used in the original paper but replaced here in order to avoid confusion with N, the total number of terms in the corpus, in Equation 2.2

and distance, except for the approach by Jakonienė et al. [2006], which uses only distances.

Rada et al. [1989] proposed the first distance-based semantic similarity measure. This simplest form of edge-based similarity counts the edges of the path between two terms. If there is more than one path, the average of the paths is taken. This kind of approach is not appropriate for use in the GO as it assumes that all edges carry the same weight, i.e. represent the same difference in meaning. This is not the case in the GO where some edges connect terms that have a very similar meaning whereas others are far more loosely related. The relationship "Golgi apparatus" (GO:0005794) IS_A "intracellular membrane-bounded organelle" (GO:0043231) is intuitively closer than the relationship "membrane" (GO:0016020) IS_A "cell part" (GO:0044464), even though the two pairs of concepts are linked by the same type of edge.

In addition, edge-counting approaches require an evenly distributed hierarchy, which is also not the case in the GO. As the GO evolves with current research trends, some areas are far deeper (longer paths from root to leaf nodes) than others even though leaf nodes with shorter paths to the root can be equally specific in their meaning. Maximum root to leaf distance varies from 2 to 15 edges in the BP ontology and from 2 to 12 edges in the CC and MF ontologies. For example, in the 2011-03 release of GO, the leaf term "nuclear outer membrane organization" (GO:0071764) has a maximum depth of 5, while the leaf term "nuclear inner membrane organization" (GO:0071765) has a maximum depth of 9.

The first edge-based approach used in the context of the GO was developed by Cheng et al. [2004]. They used the shared path, counting the edges, from the LCA of two terms to the root of the ontology. They addressed the issue of increasing specificity for deeper terms by weighting each edge with a weighting factor based on the edge's depth, as well as addressing the varying levels of depth of different parts of the ontology by defining a normalising factor based on the local depth of the ontology. The similarity between two terms was then defined as the sum of the weighted edges of the longest path between their common ancestor and the root, multiplied by the normalisation factor.

Yu et al. [2005] used two edge-based measures, referred to as "taxonomy similarity", in their work on gene function prediction, namely PK-TS proposed by Pekar and Staab [2002] and SB-TS proposed by the authors themselves and inspired by PK-TS. The former of these two measures was originally developed in a linguistics context and calculates the similarity between two terms c_1 and c_2 by dividing the distance of the shortest path between their LCA c and the root by the sum of dis-

tances between c_1 and c, c_2 and c, and c and the root, again using the shortest path in each case. In their interpretation of this approach, Yu et al. [2005] changed the distances used to the longest path. They also proposed their own approach, which does not take into account a common ancestor, but divides the distance of c_1 to the root by the distance of c_2 to the root, if c_1 is above c_2 in the hierarchy, or vice versa if c_2 is above c_1. If c_1 and c_2 are not part of the same branch of the ontology, their similarity is set to 0.

"Relative Specificity Similarity" (RSS) is a multi-component semantic similarity approach by Wu et al. [2006], considering the distance between the common ancestor of two terms and the root, the distances between the terms and their leaf node descendants and the distance between the terms and their common ancestor. The RSS approach could ostensibly be classed as hybrid, rather than an edge-based approach, as it claims to incorporate the node-based approach by Wu et al. [2005], mentioned in Section 2.2.1. However, where the original node-based approach considers the maximum number of terms between the LCA of two terms and the root, the LCA to root distance component of RSS, called α, subtracts 1 from the number of terms, which equates to counting the maximum number of edges between the LCA and the root. RSS has two further components, β and γ. β represents the largest shortest path (counting edges) between term c_1 and all its descendant leaf nodes and term c_2 and all its descendant leaf nodes. γ is the sum of the distances of each query term to the LCA, which is effectively the shortest distance between the two terms c_1 and c_2. The three components are then combined into the RSS formula, which also includes the maximum distance from the ontology root to the deepest leaf node.

Jakonienė et al. [2006] proposed a measure based the number of edges between two terms. They defined three types of paths: u, the number of "IS_A" edges needed to go up in the hierarchy, d, the number of "IS_A" edges needed to go down, and o, the number of edges of other types. The three paths are weighted by division with their respective weighting factor $p_{pathtype}$. The three values are combined in an exponential function.

For their CDGMiner tool, Yuan and Zhou [2008] defined a semantic similarity measure called $go2go$, which defines the semantic similarity between two GO terms as the multiplicative inverse of 1 plus the shortest path between the two terms. In their first paper, the authors do not specify whether the distance between two terms is obtained by counting nodes or edges. Yuan et al. [2010] then add that the distance between two directly connected terms is 1, suggesting that edges rather than nodes are counted. This assumption is also supported by the fact that 1 is added to the shortest path, as a path of n edges connects $n+1$ nodes. While the purpose of both

papers is the identification of disease genes from functional information and the calculation of semantic similarity between GO terms is the same on both occasions, the remainder of the overall approaches detailed in the two papers present a number differences and result in the implementation of two distinct tools, rather than the second paper presenting an optimisation of the first approach.

2.2.3 Hybrid approaches

A couple of semantic similarity measures combine edge- and node-based approaches, although this is not a very common practice. The semantic similarity measure developed by Wang et al. [2007] uses all aspects of the GO structure, but no external information. Each GO term has a "semantic value" (SV), which is defined as the aggregate contribution of all terms in the subgraph between that term and the root. The semantic contribution or "S-value" of each term is based on the semantic contribution of its child terms within that subgraph, multiplied by an experimentally determined semantic contribution factor, which varies depending on the type of edge that connects the terms. Semantic similarity between two terms is defined as

$$\mathrm{sim}_{Wang}(c_1, c_2) = \frac{\sum_{t \in T_{c_1} \cap T_{c_2}} (S_{c_1}(t) + S_{c_2}(t))}{SV(c_1) + SV(c_2)} \tag{2.10}$$

where T_{c_1} and T_{c_2} are the subgraphs between c_1 and the root and c_2 and the root respectively and $S(t)$ is the S-value of term t. The S-values for a given term t are different in subgraphs induced by different terms as they are the result of the S-values of the terms they subsume in a given subgraph.

The measure proposed by Othman et al. [2008] made use of the GO structure as well as external information. This hybrid approach is effectively the same as Jiang and Conrath's semantic distance approach but unlike the usual interpretation of the Jiang approach, this one makes use of the two additional factors of term depth and local network density included in the original version of the Jiang approach. The term depth factor, reflecting the distance of a term from the root, is governed by an exponent parameter α. The local network density of a term, which is related to the number of children that descend from that term, is controlled by a parameter β. $\alpha = 0$ and $\beta = 1$ are the values generally used in the interpretation of the Jiang approach as they remove the effect of their respective factors by setting them to 1.

The conceptual distance between a term c and its descendant c_1, based on the shortest path between them, is calculated as the sum of, for each term in the path, the product of the depth of the term, its local network density and the difference

in information content with its descendant. The two additional factors therefore act as weights of the relationship between each term in the path and its child. The conceptual distance between two terms c_1 and c_2 is then defined as the sum of the conceptual distance of each term to their LCA. If the term depth and local network density are set to 1, this distance calculation is reduced to the one in Equation 2.5. In their work, Othman et al. set α and β to 0.5 and 0.3, respectively, although no detail was given as to how these values were chosen. The normalised conceptual distance, or rather the corresponding similarity, was used to generate the initial population for a genetic algorithm designed to improve the large-scale retrieval of semantically similar GO terms.

2.3 Similarity between gene products

As previously discussed, this work distinguishes between semantic and functional similarity. Functional similarity approaches can be divided into two categories: group-wise and pair-wise measures. Group-wise approaches consider the annotations of a gene product as a whole, whereas pair-wise approaches, as implied in their name, consider pairs of annotations. Group-wise approaches therefore do not use semantic similarity as semantic similarity is computed between pairs of GO terms, although some group-wise approaches do employ information content in order to weight the contribution of individual terms to the annotation set. A brief overview of group-wise approaches is given here, even though they are of less interest in the context of a project that focusses on semantic similarity-based functional similarity.

2.3.1 Group-wise approaches

Group-wise functional similarity approaches can be sub-divided into three groups, namely set-based approaches, vector-based approaches and graph-based approaches, depending on how the terms annotated to gene products are considered. An overview of all approaches discussed in this section is given in Table 2.2.

Set-based approaches

The simplest approach to establish the similarity between two gene products based on their functional annotations is to apply set-based similarity techniques, such as Jaccard's index [Jaccard, 1908] or the Dice coefficient [Dice, 1945], to the sets of GO terms attributed to these gene products. Whilst relatively inexpensive in terms of computing power, purely set-based approaches are very rarely used in this context,

Measure	Approach	Similarity measure	Weighting
Lee et al. [2004]	Graph-based	Term overlap	None
Mistry and Pavlidis [2008]	Graph-based	Normalised term overlap	None
Martin et al. [2004]	Graph-based	Czekanowski-Dice	None
Gentleman [2005]	Graph-based	Jaccard	None
Gentleman [2005]	Graph-based	Shared path	None
Pesquita et al. [2008]	Graph-based	Jaccard	IC
Lin et al. [2004]	Graph-based	Intersection	Annotation set probability
Yu et al. [2007]	Graph-based	LCA	Annotation set probability
Ye et al. [2005]	Graph-based	normalised LCA	None
Sheehan et al. [2008]	Graph-based	IC-based (Resnik, Lin)	Annotation set probability
Jain and Bader [2010]	Graph-based	LCA	Term-to-leave subgraph IC
Chabalier et al. [2007]	Vector-based	Cosine similarity	IDF
Huang et al. [2007]	Vector-based	Kappa-statistic	None
Benabderrahmane et al. [2010]	Vector-based	Cosine similarity	combination of evidence code and IDF

Table 2.2: Overview of the group-wise functional similarity approaches presented in Section 2.3.1

at least not on their own. This is due to the very subtle differences that can exist between adjacent levels in biomedical ontologies. Two gene products annotated with terms that are not identical but are very close in the ontology would be scored at a much lower similarity using direct set matching than using a more complex approach taking into account ontological structure.

Graph-based approaches

Graph-based approaches consider the sub-graph formed by annotation terms, thus including also indirect annotations rather than just direct annotations in the similarity calculations. They are by far the most commonly used group-wise functional similarity approaches used in the GO.

Although set-based similarity techniques are not used in the GO for comparing sets of annotations, they are used in conjunction with graph-based approaches, treating GO term induced subgraphs as sets. The earliest example of this in the GO was by Lee et al. [2004], who defined the similarity between two gene products as the intersection of their sets of GO terms. The sets of GO terms include all parent terms of the direct annotation term, i.e. the subgraphs from term to root induced by each term. Mistry and Pavlidis [2008] refer to Lee et al.'s measure as "Term

Overlap" (TO). They present a normalised version of TO (NTO) in which the term overlap similarity is divided by the size of the smaller of the two GO term sets.

Martin et al. [2004] used a slightly more sophisticated distance measure, the Czekanowski-Dice formula, which is the cardinality of the symmetrical distance between two term sets divided by the sum of the cardinalities of their union and intersection. A similar approach was proposed by Gentleman [2005], whose simUI measure divides the cardinality of the intersection of two induced subgraphs by the cardinality of their union. This is effectively Jaccard's index. In the same work, Gentleman also proposed another measure, simLP, which is not based on set similarity. simLP is defined as the longest common path found in two subgraphs. A fifth set similarity-based approach of induced subgraphs, simGIC, was defined by Pesquita et al. [2007]. They combined the Jaccard index with information content by replacing the cardinalities of intersection and union by the sums of the information content of all the terms in the intersection and union of two term sets.

Not all graph-based functional similarity approaches applied to GO annotation make use of set similarity concepts. Lin et al. [2004] proposed to establish the shared subtree, called the "intersection tree", for all pairs of proteins in a population, then calculate the similarity between each protein pair as the frequency of their intersection tree in the overall population. The "total ancestry measure" by Yu et al. [2007] is effectively a normalised version of this as it defines the functional similarity between two proteins as the number of protein pairs in a population with exactly the same set of LCAs as the proteins in question, divided by the total number of protein pairs in the population. Although the two measures differ in their conceptual definitions, the actual calculations are essentially the same.

The approach suggested by Ye et al. [2005] focusses on the depth of the shared part of the induced subgraphs of two gene products. Similarity is calculated by dividing the difference between depth of the deepest common term and the minimum depth of the ontology (always 1) with the difference between maximum and minimum depth of the ontology.

Sheehan et al. [2008] propose the SSA algorithm, a rule-based system that extends information content similarity between GO terms, particularly Resnik's and Lin's measures, to a framework for describing the similarity between sets of annotations. Based on the GO graph structure and the relationships between terms, the SSA algorithm derives a set of "contextual terms" that describe the annotations of two gene products. This term set, called "nearest common annotation" (NCA), is used as the LCA of the gene products' annotations and based on the instances of the term set in a corpus of annotations, the similarity between gene products is

calculated according to the same principle as Resnik's or Lin's similarity between GO terms.

The measure by Cho et al. [2007] was classed as a separate graph-based functional similarity measure by Pesquita et al. [2009]. This measure is however essentially Resnik's information content semantic similarity measure, combined with maximum functional similarity (see Section 2.3.2). The only difference in this new measure is in the way the calculation is defined. Rather than calculating the similarity between each pair of GO terms, then combining the pairwise semantic similarities into a functional similarity score, Cho et al. use the smallest GO term "annotation size" of all GO terms shared between two gene products. Annotation size is defined as the number of proteins annotated to a GO term or any of its child terms. The smallest annotation size is divided by the annotation size of the root and the similarity between two gene products is the negative log of this ratio. The measure is mentioned here due to its inclusion in Pesquita et al.'s review but is not considered as a graph-based functional similarity measure.

Similarly, Jain and Bader [2010] also define functional similarity between two gene products as the maximum information content of the lowest common ancestor of their annotation terms. In this work, the authors transform the GO into a set of subgraphs but unlike other works, where a subgraph is generally the part of the ontology between a term and the root, these subgraphs reach from a high-level term to the leaves of the ontology. The subgraphs are defined so there is minimal overlap between them. Multiple subgraphs form a meta-graph based on the position of their respective root nodes in the original GO hierarchy. The information content for each GO term within a subgraph is calculated using only the terms and annotation frequencies within that subgraph. The higher-level terms that are not part of a subgraph have their information content calculated based on occurrence probabilities from all the subgraphs they subsume. Through this system, gene products that are annotated with terms from the same subgraph have higher similarity than terms from different subgraphs.

Vector-based approaches

Vector-based approaches generally represent the gene product annotations as multi-dimensional vectors, where each dimension represents one possible GO term. Vectors can be binary, with the presence or absence of each term in a given set of annotations denoted by 1 or 0 respectively. Alternatively, vectors can be weighted, making the contribution of each term to the vector more nuanced. While vector-based

approaches have been used in the GO context, they are far less common than graph- and information theory-based methods. This is mostly because they are highly computationally intensive, yet just like set-based approaches, fail to capture the information contained in the ontological structure. Efforts to date include Chabalier et al. [2007]'s cosine similarity-based functional similarity, Huang et al. [2007]'s kappa statistics approach used in DAVID and, most recently, Benabderrahmane et al. [2010]'s variant on weighted cosine similarity.

Chabalier et al. used the same approach described by Bodenreider et al. [2005], but calculated the similarity between gene products based on vectors of GO terms rather than the other way round. The authors also used IDF to weight the contribution of each GO term to a gene's annotation vector. Pesquita et al. [2009] equate this to weighting using information content, which is not entirely appropriate as the probability of occurrence in IC is based on the total number of annotations of a term or any of its children divided by the total number of annotations in the corpus, while IDF is based on the number of occurrences of term t divided by the total number of distinct genes.

A new GO-specific weighting approach was defined by Benabderrahmane et al. [2010]. In their approach, each dimension of each vector consists of both a coefficient that is the product of a weight that reflects the evidence code of that annotation and the IDF for that term, and a base vector. In the calculation of functional similarity between two gene products using cosine similarity, the dot product between the two base vectors for a given dimension reflect the ratio of the depth of the two terms' common ancestor and the sum of the depths of the two terms.

Huang et al. [2007] proposed to quantify the similarity between gene products using kappa statistics [Cohen, 1960], a chance-corrected measure of co-occurrence. They also represented the gene products as vectors of their annotations but included not only GO terms but also annotations from a number of other sources, such as KEGG pathways [Kanehisa and Goto, 2000], UniProt sequence features [The UniProt Consortium, 2008] and InterPro domains [Mulder et al., 2003]. Each gene product-term association is binary, with no weighting. The DAVID tool also uses the reverse approach (annotations represented as vectors of the gene products they annotated) to calculate the similarity between annotation terms.

2.3.2 Pair-wise approaches

Pair-wise approaches can be classified by the number of pairs they consider, i.e. some consider all possible pairs of GO terms from a set of annotations, whereas

others consider only the best pairs. For each approach, the chosen pairs are then combined into a single score through one of a number of techniques including average, maximum and sum. Pair-wise approaches can be used in conjunction with any type of semantic similarity measure. Table 2.3 presents an overview of all the pair-wise approaches discussed here.

First used by	Pairs	Measure	Notes
Lord et al. [2003a]	All pairs	Average	
Sevilla et al. [2005]	All pairs	Maximum	
Lei and Dai [2006]	All pairs	Sum	
Azuaje et al. [2005]	Best pairs	Average	Average of bidirectional summed scores
Couto et al. [2005]	Best pairs	Average	Average of directional averages
Schlicker et al. [2006]	Best pairs	Average	funSim combination of ontological scores
Tao et al. [2007]	Best pairs	Average	Reciprocal best matches only and minimum similarity threshold
Lei and Dai [2006]	Best pairs	Sum	

Table 2.3: Overview of the pair-wise functional similarity approaches presented in Section 2.3.2

All pairs

Some pair-wise approaches use all possible pairs of GO terms from two annotation sets in order to compute the gene products' overall similarity.

The simplest pair-wise functional similarity approach is the straightforward average (AVG). As implied in the name, it simply averages the semantic similarity between all GO term pairs that make up the set of annotations of the two gene products g_1 and g_2 [Lord et al., 2003a]:

$$\text{sim}_{AVG}(g_1, g_2) = avg_{c_1 \in GO(g_1), c_2 \in GO(g_2)}(sim(c_1, c_2)) \tag{2.11}$$

where $GO(g_1)$ and $GO(g_2)$ are the sets of GO terms annotated to g_1 and g_2, respectively.

The main drawback of this approach is that because it treats all GO term pairs equally, it produces inappropriate results for gene products that share several unrelated functional aspects. For example, two genes A and B that are annotated with the same two unrelated GO terms x and y would only have a similarity of 50% as $sim(x, y) \neq 100\%$.

Another way of calculating the functional similarity between gene products is to only consider the GO term pair with the largest semantic similarity (MAX) [Sevilla et al., 2005], i.e.

$$\text{sim}_{MAX}(g_1, g_2) = max_{c_1 \in GO(g_1), c_2 \in GO(g_2)}(sim(c_1, c_2)) \tag{2.12}$$

This approach is useful to highlight whether two gene products share a functional aspect, but it does not represent the global functional similarity between gene products as it ignores all annotations except for the most similar one. Two genes A and B which share one term x but also have other non-related annotations would have a similarity of 100% regardless of these other GO terms.

There is one instance in the literature [Lei and Dai, 2006] where the functional similarity is computed as the sum of all the pair-wise GO term similarities.

$$\text{sim}_{SUM}(g_1, g_2) = \sum_{c_1 \in GO(g_1), c_2 \in GO(g_2)} sim(c_1, c_2) \tag{2.13}$$

This technique performed worse than either AVG or MAX. It is never used elsewhere in GO semantic and functional similarity.

Best pairs

Rather than considering all possible permutations of a set of GO terms, it is usually more advisable to consider only the best pairs. This eliminates situations like the one mentioned in relation to the AVG approach, where two identical sets of annotations lead to a lower annotation score because the terms within each set are semantically unrelated.

Most commonly, the similarity between the best pairs is averaged, a functional similarity approach called "best match average" (BMA) approach [Couto et al., 2005]. This approach takes the highest similarity between each term in $GO(g_1)$ and all the terms in $GO(g_2)$,

$$\text{score}(g_1 \rightarrow g_2) = \sum_{c_1 \in GO(g_1), c_2 \in GO(g_2)} max(sim(c_1, c_2)) \tag{2.14}$$

and vice versa, then averages them.

There are two different ways of performing this averaging process. Probably the most common approach calculates the average for each direction, then averages the two directed scores [Couto et al., 2005].

$$\text{sim}_{BMA}(g_1, g_2) = \frac{1}{2} \times \left(\frac{\text{score}(g_1 \rightarrow g_2)}{m} + \frac{\text{score}(g_2 \rightarrow g_1)}{n} \right) \tag{2.15}$$

where m is the number of GO terms in $GO(g_1)$ and n is the number of GO terms in $GO(g_2)$.

Some users however add all the maximum similarity scores in both directions, then average the total [Azuaje et al., 2005].

$$\text{sim}_{BMA}(g_1, g_2) = \frac{\text{score}(g_1 \rightarrow g_2) + \text{score}(g_2 \rightarrow g_1)}{m + n} \tag{2.16}$$

The issue with this approach is that it treats the number of annotations for each gene product equally even though one gene product may have far more GO terms annotated to it than the other. The first approach provides a more realistic score in the sense that each directional score is averaged only in relation to the number of elements it is made up of. Overall, the BMA approach provides a good balance between the AVG and MAX approaches and most realistically reflects the functional similarity between gene products.

A variant of BMA was proposed by Tao et al. [2007], who only consider reciprocal best matches between two sets of GO terms. If the best match for a term a in $GO(g_1)$ is a term b in $GO(g_2)$, it will only be considered if the best match for b is a in the reciprocal comparison. In addition, the authors also used a minimum similarity threshold, so even reciprocally matched pairs were only considered if their similarity was higher than a given threshold. The similarity between acceptable matched pairs was summed, as in standard BMA, then multiplied by two, which is the same as summing the scores both ways. The overall score was then divided by the sum of the sizes of the two GO term sets.

The same work that used the SUM approach for all pairs [Lei and Dai, 2006] also used this approach in conjunction with best pairs.

2.3.3 FunSim

Many studies that use functional similarity consider only one of the three GO aspects, or consider each aspect individually. Although all of the above functional similarity measures can be applied to the full set of annotations directly, Schlicker et al. [2006] proposed an approach whereby three functional similarity scores are obtained for each gene product pair, one for each of the three aspects of the GO. These scores are then aggregated into the final functional similarity, $funSim$.

$$funSim = \frac{1}{3} \cdot \left[\left(\frac{simBP}{max(simBP)} \right)^2 + \left(\frac{simMF}{max(simMF)} \right)^2 + \left(\frac{simCC}{max(simCC)} \right)^2 \right] \tag{2.17}$$

where $max(sim)$ is the maximum possible similarity for an aspect. The advantage of using $funSim$ is that by squaring the contribution made by each GO aspect, this contribution is made even stronger for high scores and even weaker for low scores.

2.4 Evaluating semantic and functional similarity

In this thesis, a distinction is made between works that evaluate semantic and functional similarity approaches and works that make use of these approaches to answer a biological question. The former will be discussed in this section, while the latter will be covered in Section 2.5. In particular, not all semantic and functional similarity approaches described in this chapter were rigorously evaluated against other forms of biological similarity, so not all of them will be covered here. Only the examples of evaluation most relevant to the work presented in this thesis will be discussed.

One key issue with the evaluation of semantic similarity approaches is that there is no benchmark for measuring functional similarity. There are a number of other kinds of biological similarity against which functional similarity can be evaluated, although they all have both advantages and disadvantages. One of the most commonly used approaches is sequence similarity. Indeed, it has been demonstrated that in many cases, sequence similarity also implies functional similarity. There are however also a not insignificant number of examples of convergent evolution where functional homologues (i.e. gene products with the same or highly similar function) have little or no sequence similarity, as well as examples of divergent evolution, where sequence homologues (i.e. gene products with high sequence similarity, e.g. from gene duplications) have little or no functional similarity.

Another popular evaluation approach is to use gene expression similarity as a benchmark for functional similarity. As with sequence similarity, similarity in gene expression in many cases implies functional similarity but equally, there are situations where functionally similar gene products have expression profiles that bear no resemblance to each other.

The same problem applies to almost all approaches for evaluating functional similarity, with the possible exception of human judgement. This approach however

has the double disadvantage of lacking objectivity as it requires expert knowledge and of being limited to small datasets as manually evaluating hundreds or even thousands of gene product pairs is not practical. All other approaches applied to date reflect functional similarity up to a point but all include a significant element of false positive and false negative generation. This is why there is a need for functional similarity in the first place. If another form of biological similarity perfectly matched functional similarity, the latter would not be of interest.

A further difficulty in functional similarity evaluation is that it is very hard to make comparisons across studies. Information content-based measures in particular are dependent on the corpus used to calculate the results. While non-IC measures do not suffer from this drawback, they are still susceptible to changes in the ontology, so two studies using exactly the same dataset and parameters may differ in their results if based on two different GO releases. For this reason, it is essential to state exact details of all parameters used, including GO release, included or excluded evidence codes, dataset, measures and any other study-specific variables.

In their assessment, Lord et al. [2003a] validated the semantic similarity measures for Resnik, Lin and Jiang, with AVG functional similarity, against sequence similarity scores. They found a very significant degree of correlation between the two types of similarity, particularly for the molecular function aspect. They used the SWISS-PROT-Human database for the estimation of concept frequencies. GO's three aspects were considered individually and all ontological edges between concepts were treated as "IS_A" links. The authors found that none of the three methods significantly outperformed the others. There were differences in individual performances, e.g. for the molecular function aspect, the Resnik approach obtained the highest correlation with sequence similarity, but the approach performed worst for the other two ontological aspects, while the Jiang approach scored the lowest correlation for MF. It was also found that the different aspects of GO are largely independent of each other.

Other evaluation studies using sequence similarity as a benchmark include Pesquita et al. [2008] and Mistry and Pavlidis [2008]. Both included the original measures studied by Lord, but Pesquita et al. added MAX, BMA and GraSM with BMA functional similarity, as well as the graph-based approaches simGIC and simUI, while Mistry and Pavlidis also added MAX functional similarity, and kappa, cosine, weighted cosine, Term Overlap (TO) and NTO similarity measures. Corpora were UniProt [The UniProt Consortium, 2008] and NCBI Gene [Pruitt et al., 2006] for mouse genes, respectively. Pesquita's study found that simGIC performed highest overall, with Resnik's measure the best out of the IC-based measures. Mistry's

work found the best correlation between sequence and functional similarity for TO and Resnik with MAX. The study also included a comparison of different measures against each other in which TO and Resnik/MAX also had the highest correlation with each other.

Wang et al. [2004] investigated the relationship between semantic similarity and gene expression using the same approaches as Lord et al. The data for this work was derived from microarray experiments, with the Saccharomyces Genome Database [Cherry et al., 1998] used as the corpus to estimate term frequencies. Functional similarity values were averaged across five expression correlation intervals. Significant correlation was found between GO-based similarity and gene expression for all three approaches and for all three aspects of the GO, but as for Lord, none of the approaches outperformed the others.

Similar experiments, using the same semantic similarity approaches but MAX functional similarity and also considering correlation with gene expression, were carried out by Sevilla et al. [2005]. This group used data from mouse gene expression experiments. Unlike any of the previous groups, they concluded that the Resnik approach significantly outperformed both Lin and Jiang. This interpretation was given with the justification that the latter two are relative measures, which may give misleading results if the gene product annotations are too general and do not exploit the full depth of knowledge available in the GO.

In both of these, as well as any subsequent gene expression-based studies, expression correlation values were averaged across intervals of semantic similarity. Only Sevilla et al. [2005] commented on the pair-by-pair results, which were found to show poor correlation.

Couto et al. [2005] investigated the relationship between semantic similarity and protein families, using UniProt as corpus and the same semantic similarity measures as Lord, but BMA functional similarity, as well as adding their own GraSM ancestor choice. They found a good degree of correlation, with Jiang's method performing strongest in their measurements, and Lin's approach mostly outperforming Resnik. They also found that GraSM outperformed the single ancestor approach.

Schlicker et al. [2006] included both sequence similarity and family similarity in their evaluation and concluded that their sim_{Rel} measure outperforms Resnik and Lin.

Other forms of similarity used to date include protein-protein interactions and clustering with human judgement. The latter was used by Wang et al. [2007] to validate their hybrid semantic similarity approach. They used a set of manually curated pathways from the SGD and based on each pathway, stipulated which protein

pairs should be more or less similar than others, then compared this to the similarity calculated based on their approach and on Resnik's approach, and the hierarchical clusterings produced for each of these sets of similarity values. Their conclusion was that their own measure was more consistent with the human perspective than Resnik's.

Guo et al. [2006] used protein interactions derived from human regulatory pathways in KEGG [Kanehisa and Goto, 2000] to compare two graph-based functional similarity measures (simUI and simLP) and three IC-based semantic similarity measures (Resnik, Lin, Jiang) with MAX functional similarity. They used receiver operating characteristics (ROC) analysis [Fawcett, 2006] (see Section 3.2.1 for more details) to evaluate performance and found that Resnik's measure once again performed best.

In order to address the difficulty of finding a gold standard for evaluating functional similarity, Xu et al. [2008] used both protein-protein interaction and gene expression data to evaluate the performance of a number of measures, including those by Schlicker et al., Tao et al. and Wang et al., as well as Resnik's measure with MAX and AVG. They used ROC curves on a dataset of protein-protein interactions in yeast to evaluate the performance of the different measures and calculated the correlation between gene expression and functional similarity based on Sevilla et al.'s approach of averaging across set intervals. In both comparisons, the MAX approach for Resnik outperformed the other approaches.

2.5 Applications of semantic and functional similarity

Unlike the studies described in the previous section, where the primary goal of the work is the evaluation of semantic and functional similarity measures against some other form of biological similarity, the majority of work involving semantic and functional similarity uses these approaches as tools to address biological questions. The most common areas of application include gene expression analysis, where functional similarity has been used for missing value estimation and data classification using *a priori* knowledge, prediction and evaluation of protein-protein interactions, and prediction of gene function, to name but a few. There are now too many different applications of semantic and functional similarity to give a comprehensive overview, so only the most relevant examples are discussed here.

By far the most common application of functional similarity is the analysis of

gene expression data. Functional similarity is not the only approach used to characterise clusters derived from expression similarity with functional annotation, but it is becoming increasingly common. In particular, the inclusion of existing knowledge in the actual clustering algorithm to improve its accuracy is now a ubiquitous practice. It has been suggested that some caution may be advised with this type of approach, in particular in organisms where functional annotation is not comprehensive or of poor quality, in which case independent validation of clustering results with functional annotation would be more appropriate than its inclusion in the results generation [Romero-Zaliz et al., 2008].

One of the first clustering algorithms to incorporate *a priori* knowledge in the form of functional annotation and using functional similarity was that by Speer et al. [2004], who proposed a memetic co-clustering algorithm. Subsequent works focus primarily on different types of clustering algorithms and different forms of semantic and functional similarity. They include efforts by Brameier and Wiuf [2007] (SOM-based co-clustering approach), Ovaska et al. [2008] (semantic similarity-based combination of expression data and GO annotations using hierarchical clustering and heat map visualisation), Dotan-Cohen et al. [2009] (integration of semantic similarity into a hierarchical clustering algorithm), Schön et al. [2010] (optimisation with semantic similarity of the classification of gene pairs based on the difference in their expression values), Kustra and Zagdanski [2010] (combination of expression data and GO-derived information to improve clustering) and Kang et al. [2010] (SICAGO, a SemI-supervised Clustering Analysis using semantic distance between gene pairs in Gene Ontology). Most recently, Azuaje et al. [2011] applied their approach that integrates gene expression and functional knowledge in order to identify new treatment responses of endothelial progenitor cells.

The incorporation of prior knowledge into expression clustering algorithms is however not the only use of functional similarity in gene expression. Both Tuikkala et al. [2006] and Pourhashem et al. [2010] proposed approaches that use semantic similarity to aid missing value estimation. The former incorporated semantic similarity with a k-nearest neighbour algorithm, while the latter applied it to fuzzy clustering.

Efforts using semantic and functional similarity to assess the validity or functional coherence of clusters generated by traditional techniques were proposed by Bolshakova et al. [2005] (cluster validity assessment), Chagoyen et al. [2008] (protein set coherence) and Richards et al. [2010] (functional coherence of gene sets). The latter included an unusual use of information content in that they define the distance between parent-child GO terms as the difference of their respective information

contents.

Further applications of semantic similarity in gene expression analysis include the construction of gene regulatory networks using a combination of expression data and functional annotation by Jing et al. [2010] and Papachristoudis et al. [2010]'s combination of semantic and statistical knowledge to improve marker gene selection in microarray experiments. Wolting et al. [2006] proposed an approach to analyse sets of proteins using functional similarity-based clustering.

Another area in which functional similarity is now commonly applied is the study of protein-protein interactions (PPIs) and domain-domain interactions (DDIs). Ramírez et al. [2007] proposed to assess the quality and potential bias of different PPI datasets using various parameters including functional similarity, while Schlicker et al. [2007a] analysed predicted and experimental DDIs and PPIs to assess the quality of confidence ranking of predictions and also derived confidence score thresholds to class predictions as low or high quality. Jain and Bader [2010]'s TCSS (Topological Clustering Semantic Similarity) approach also uses GO for PPI confidence assessment and Pandey et al. [2010] investigate the relationship between functional coherence and topological proximity in PPI and DDI networks. An approach that uses prior knowledge to mine PPI networks to identify functional modules was proposed by Jing and Ng [2010]. Wang et al. [2010b] used a semantic similarity-based framework for the analysis of protein-protein interaction networks. Dotan-Cohen et al. [2009] analysed PPIs, co-expression and genetic interactions to derive patterns of interactions between biological processes and improve both the coverage and the accuracy of protein function predictions.

In fact, functional predictions represent a further area in which functional similarity has been applied more recently, as the quality of functional annotation has improved. Louie et al. [2009] created a statistical model of protein sequence similarity and function similarity based on experimentally validated functions to predict functional similarity between two proteins based on their sequence similarity, while Altenhoff and Dessimoz [2009] used functional and phylogenetic measures to assess the quality of ortholog inference. Hawkins et al. [2010] proposed the construction of functional similarity networks to get high-confidence function predictions and Tedder et al. [2010] tested an approach for gene function prediction using semantic similarity clustering by k-nearest neighbour and enrichment analysis. Fontana et al. [2009] proposed ARGOT (Annotation Retrieval of Gene Ontology Terms), an approach for functional annotation inference of protein sequences that combines the clustering of GO terms based on semantic similarity with a weighting scheme based on shared biological features with the sequence to be annotated. Also prediction-related but

for different entities, Roubelakis et al. [2009] incorporated functional analysis in a tool for miRNA target prediction.

However not all uses of functional similarity are limited to these three large areas. Cao et al. [2004] integrated semantic similarity in a semantic search algorithm linking heterogeneous biological databases. In their MedSim algorithm, Schlicker et al. [2010] proposed a novel approach for ranking candidate genes for a particular disease based on functional comparisons in GO and other ontologies. Hakes et al. [2007] used functional similarity as one of several parameters to compare pairs of gene duplicates from small-scale and large-scale gene duplications to determine whether there are quantifiable differences between the two types of duplication. Similarly, Li et al. [2010] studied the functional redundancy between duplicated genes in yeast using functional similarity approaches.

2.5.1 Existing tools

There are a number of tools available that calculate semantic similarity between GO terms or functional similarity between gene products. Some of these tools also make use of the similarity scores they produce as part of a larger analysis goal. The majority of available tools are web-based, although a couple are stand-alone tools or add-on packages for existing tool suites such as R [R Development Core Team, 2010].

The first publicly available tool for calculating semantic similarity between GO terms and functional similarity between genes was FuSSiMeg[5] (FUnctional Semantic SImilarity MEasure between Gene-products), a web-based tool by Couto et al. [2003]. This very basic tool allowed calculation of the similarity between only two terms or genes, using the measure by Resnik, Lin or Jiang, with single ancestor or GraSM. The tool is no longer supported and has been replaced by the more sophisticated, also web-based ProteInOn[6] (PROTEin INteractionas and ONtology) [Faria et al., 2007]. This new tool not only includes two further measures, simUI and simGIC, but also allows the exclusion of electronic annotations, computes the similarities between up to 1000 elements and provides retrieval functionality for common ancestors, interacting proteins and assigned annotations.

An even more versatile web-based tool is FunSimMat[7] (FUNctional SIMilarity MATrix) by Schlicker and Albrecht [2007]. In addition to providing even more semantic and functional similarity measures than ProteInOn, FunSimMat has a range

[5] http://xldb.fc.ul.pt/biotools/rebil/ssm/, last accessed 26/02/2011
[6] http://xldb.di.fc.ul.pt/biotools/proteinon/, last accessed 12/03/2011
[7] http://funsimmat.bioinf.mpi-inf.mpg.de/index.php, last accessed 29/04/2011

of comparison options between individual and lists of GO terms and genes. More recently, the tool also provides an option for ranking and comparing candidate disease genes to OMIM [McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University (Baltimore, MD) and National Center for Biotechnology Information, National Library of Medicine (Bethesda, MD), 2011] diseases [Schlicker and Albrecht, 2010; Schlicker et al., 2010]. The authors of FunSimMat also maintain another tool, GOTaxExplorer[8] [Schlicker et al., 2007b], a part-web-based, part-standalone tool for selecting and comparing lists of GO terms, genes and Pfam [Finn et al., 2010] families.

Other web-based tools include Du et al. [2009]'s G-Sesame, which provides various options for semantic and functional comparisons of pairs and lists of GO terms and genes, as well as a knowledge discovery module for clustering functionally similar genes. GOToolBox by Martin et al. [2004] is a more general-purpose GO-based analysis tool for gene datasets, which only provides an option for functional similarity calculations based on the Czekanowski-Dice formula (see Section 2.3.1) and does not implement any semantic similarity approaches.

Not all tools for calculating semantic and functional similarity are web-based. There are, for example, a number of packages for the statistical computation environment R [R Development Core Team, 2010]. They include SemSim [Guo, 2007], csbl.go [Ovaska et al., 2008] and GOSemSim [Yu et al., 2010]. All three of these packages implement a number of semantic and functional similarity measures including, in all cases, Resnik, Lin, Jiang and Schlicker, as well as various others, with csbl.go providing the most approaches. All three packages also provide some form of similarity-based clustering. SimTrek by Wang et al. [2010a] on the other hand is a Cytoscape [Shannon et al., 2003] plug-in that allows the calculation and display of similarity between GO terms or genes for several different measures.

In addition to these tools that calculate the semantic or functional similarity between GO terms or gene products and then make that similarity available to the user, there are also tools that make use of these measure without explicitly providing pair-wise similarity values. Examples of these include DAVID [Huang et al., 2007], GOmir [Roubelakis et al., 2009] and SICAGO (SemI-supervised Clustering Analysis using GO) [Kang et al., 2010]. The first of these is a web-based resource for the functional analysis of gene lists. It makes use of kappa-statistics and vector-based similarity to functionally cluster either a list of genes or the functional terms annotated to a list of genes. SICAGO is a stand-alone piece of software for clustering gene pairs based on the correlation between GO semantic distance and gene expression

[8] http://gotax.bioinf.mpi-inf.mpg.de/index.php, last accessed 29/04/2011

similarity, for a number of semantic similarity measures. GOmir, also a stand-alone tool, incorporates functional analysis in a process to predict or verify targets for miRNA interactions.

2.6 Summary

Semantic and functional similarity approaches, both those designed specifically for use with the Gene Ontology and those proposed in a different context, are abundant and cover a wide range of applications. There is currently no consensus as to which measure is "the best". Some attempts at comparative analysis have shown Resnik's measure to perform better than other information content approaches in a number of contexts. Information content-based measures, despite their obvious drawback of being dependent on an underlying body of knowledge, are very popular and probably the most commonly used. Applications of semantic and functional similarity have been proposed in a variety of bioinformatics contexts and novel applications are published on a regular basis.

In the next chapter, the selection of approaches to be used for the rest of this thesis will be discussed. A novel algorithm, FuSiGroups, using both semantic and functional similarity to group GO terms and gene products into groups reflecting distinct functional aspects, will be presented. An evaluation strategy for the different parts of the project will be proposed and a few important implementation consideration will be discussed.

Chapter 3

The study domain

The goal of this study is two-fold. First of all, a number of carefully selected semantic and functional similarity approaches are compared to each other, using different parameters, in order to evaluate their respective performances and determine whether one or more approaches perform better than others, and under which conditions. Secondly, a newly developed grouping algorithm is tested using the best semantic and functional similarity approaches and associated parameters.

This chapter consists of three parts. In the first section, the reasons for the selection of the different semantic and functional similarity measures are discussed. Other considerations, such as annotation types and dataset are also discussed. Finally, the new grouping algorithm is introduced. The second section details the evaluation strategy for the two aspects listed above. The third section describes the implementation of the algorithm and a number of experimental considerations such as appropriate database formats and practicalities to be considered in the dataset selection.

3.1 Study design

3.1.1 Semantic similarity approaches

In Chapter 2, a wide selection of different semantic and functional similarity approaches used in relation to the GO were discussed to provide an overview of the current state of the area. Experimentally comparing all of these approaches is beyond the scope of the current research. Therefore it was necessary to make a selection among the approaches identified. The starting point for the selection was a paper by Lord et al. [2003a], who were the first to propose the use of semantic similarity approaches in the context of the GO. They used three semantic similarity approaches,

by Resnik [1995], Lin [1998] and Jiang and Conrath [1997] for this role. All three approaches had originally been developed for use in a natural language context.

All three approaches are based on the concept of information content (IC), i.e. they use annotation frequencies to determine how informative each term in an ontology is. The similarity between terms is computed based on different combinations of the IC of common ancestors and IC of the terms themselves. As discussed in Section 2.2.1, IC-based approaches have a major disadvantage, in that similarity results are dependent on the annotation data used; using a different corpus of annotation data, such as an updated version, gives different results. This makes it difficult to replicate results or preserve consistency across different data sources.

In addition, each of the three approaches studied by Lord et al. has individual disadvantages, as discussed in Section 2.2.1. Resnik's approach reflects the distance between the common ancestor of two terms and the root, i.e. the ancestor's position in the hierarchy, but not the distance between the terms and their ancestor. Lin's method on the other hand reflects the distance between the terms and their ancestor but not the relative position of any of these in the ontology, i.e. terms close to the root can have the same level of similarity as leaf terms. The same problem applies to the measure by Jiang and Conrath. Both drawbacks were addressed by Schlicker et al. [2006], who combined Resnik's and Lin's measures in a new measure, sim_{rel}.

The paper by Lord et al. was used as the starting point for the selection of semantic similarity measures, both because it was the first paper to address semantic similarity in the GO and because most subsequently developed semantic similarity approaches are evaluated against one or all of the approaches discussed by Lord. Thus these approaches appear to be the obvious choices for use in this study. However, it was decided to exclude Jiang and Conrath from the study, as it has the same drawback as Lin, which meant it would not add a new dimension to the work. Furthermore, the drawbacks of both the Lin and Resnik methods are addressed by Schlicker et al., but the Jiang and Conrath method was not mentioned in that study. It was therefore deemed most sensible to study Resnik, Lin and Schlicker's methods as all of these approaches are related but exclude the Jiang and Conrath method.

Unlike Lin's and Schlicker's approaches, which are normalised measures and therefore give results between 0 and 1, Resnik's measure is bounded between 0 and $ln(N)$[1], where N is the size of the corpus. In order to make results based on Resnik more comparable, it is possible to normalise each result by dividing it by $maxIC$, the maximum possible information content for a given analysis [Couto et al., 2007].

[1] $ln(N) = ln(\frac{1}{N}^{-1}) = -ln(\frac{1}{N})$

In practical terms, *maxIC* is derived for a hypothetical term with an occurrence frequency of 1.

All these approaches use the most informative common ancestor (MICA) of two terms to calculate their similarity. Couto et al. [2005] argue that using only the MICA ignores a lot of the informative content of the ontological structure and they proposed "GraSM", an algorithm that utilises all the common disjunctive ancestors of a pair of terms, i.e. all the ancestors that can be reached by at least one distinct path. Since this algorithm can be applied to any of the three chosen approaches, it was decided to incorporate this alternative in the study.

Due to the disadvantage of using information content based measures, discussed in Section 2.2.1, it was decided that the study should include one non-IC measure. The precision measure by Herrmann et al. [2009], the only non-IC node-based approach found to be used with the GO, was not published until September 2009, over a year after the selection of the measures to be included in this study. Potential non-IC approaches therefore had to be either edge-based or hybrid. The main disadvantage of edge-based approaches, that they treat all edges as the same and also treat all terms on a given depth level as equally informative, was considered too great a drawback to select a purely edge-based approach. A hybrid measure, using information from both nodes and edges, was therefore the best choice. As the approach by Othman et al. [2008] uses IC as part of a hybrid approach, its use would not address the stated aim of including a non-IC measure. The remaining choice was the approach proposed by Wang et al. [2007], which was therefore selected.

As discussed in Section 2.2.3, Wang's approach assigns each GO term a semantic value that is an aggregate of the contributions of all the GO terms and the edges between them in the induced subgraph from a term to the ontology root. The edges that connect the terms are weighted according to their type. While Wang's measure does not suffer from a dependency on corpus data, the empirical nature of the edge weights may potentially be considered a disadvantage. It should also be noted that Wang's approach suffers from a similar drawback as Lin's in that very similar shallow terms can have high semantic similarity despite potentially being quite uninformative.

3.1.2 Functional similarity approaches

Semantic similarity measures are used to quantify the relationship between two ontology terms. Gene products are usually annotated with multiple terms from each GO sub-ontology, in order to capture the multiple facets of their function. It is

therefore necessary to have strategies which evaluate the overall functional similarity of two gene products based on the set of GO terms annotated to them. As discussed in Section 2.3, functional similarity approaches can address these sets of GO terms as a whole (group-wise approaches) or as individual terms (pair-wise approaches). Since group-wise approaches do not require the semantic similarity between GO term pairs and an important aspect of this study is the comparison of different semantic similarity measures, group-wise approaches are not relevant here. There are three pair-wise approaches to be considered.

The most obvious approach is to simply average the semantic similarities of all possible GO term pairs in the two annotation sets [Lord et al., 2003a]. Although the simplest, the average (AVG), or all-with-all, approach is also the least discriminating as it treats all term pairs equally. This means that two gene products annotated with the same two terms A and B, which characterise two very different functional aspects and therefore have low semantic similarity, would have at best an average level of functional similarity as the similarity calculation would be skewed by $sim(A, B)$. This would lead to two gene products being ranked as less similar than they really are. Two isoenzymes, for example, that are both active in two distinct cellular compartments, such as Golgi apparatus and mitochondria, would have identical annotations but would have at best average similarity as the semantic similarity between the two CC terms would be very low.

Another approach, which avoids the obvious problem of the all-with-all approach, is to consider only the GO term pair with the largest similarity [Sevilla et al., 2005]. While this approach, MAX, is useful for identifying the most important shared aspect of a pair of gene products, it does also have its disadvantages. Firstly, by considering only the most similar aspect shared by two gene products, it disregards all their other annotations, regardless of how similar or different they might be. It can also result in two gene products being ranked as more similar than they really are, if they are both annotated with the same or two highly similar uninformative GO terms, e.g. two gene products annotated with the term "cytoplasm" (GO:0005737, annotated to more than 100000 gene products) would have very high similarity with respect to the CC ontology without necessarily having any common function.

The third pair-wise approach, the best match average (BMA) approach [Couto et al., 2005; Azuaje et al., 2005], addresses the drawbacks of both AVG and MAX. BMA considers the pair with the highest similarity for each subset (go_A, GO_B), where go_A is a GO term annotated to gene product A and GO_B is the set of GO term annotated to gene product B and vice versa, then averages these maximum similarities. The selection of the GO term pair with the highest similarity for each

subset addresses the disadvantage of AVG by disregarding pairs of GO terms which are completely unrelated if there is a pair with a higher similarity involving one of the two terms. The selection and averaging of multiple similarities addresses MAX's issue of disregarding two gene products' full annotation sets in favour of a single term pair, thus preserving the information richness of the annotation and not unduly prioritising a single term pair. As discussed in Section 2.3.2, there are two ways of calculating BMA, namely by either first averaging the similarities for each direction ($A \rightarrow B$ and $B \rightarrow A$), then averaging the two resulting scores [Couto et al., 2005], or averaging all the scores directly [Azuaje et al., 2005]. Only the first approach will be considered in this work, as it respects the varying sizes of gene product annotations.

Out of these three pair-wise approaches, AVG is probably the least appropriate. MAX on the other hand, despite its obvious disadvantage, can be very useful if the goal of a study is to identify gene products which have a given aspect in common, in order to then study their unrelated aspects or discover if other aspects of similarity can be inferred. For this reason, the MAX approach will be included in the present study as well as the BMA approach, which is the most appropriate to characterise the overall functional similarity between two gene products.

3.1.3 Ontological aspects

In addition to the various semantic and functional similarity approaches available, there are a number of other considerations that need to be taken into account in a study of semantic similarity. One of these considerations is which ontological aspect or aspects to use. The GO consists of three orthogonal ontologies, molecular function (MF), biological process (BP) and cellular compartment (CC), and the majority of gene products are annotated with at least one term from each sub-ontology. Most semantic similarity studies only consider one of the three ontologies or consider all three but as three separate scores. If all three scores are considered together, the overall functional similarity is usually established using one of the functional similarity approaches discussed above, unless a group-wise approach is used. Regardless of which functional similarity measure is used and its individual drawbacks, any of these approaches, if applied equally to all three GO aspects, can lead to a misleading overall functional similarity. Two gene products might, for example, have very high similarity in the MF ontology, as they share a common function, but be involved in different biological processes in different cellular compartments, thus giving an overall average to low functional similarity score. In order to address this, Schlicker

et al. [2006] developed a combination measure, $funSim$ (see Section 2.3.3), which averages the squares of the three functional aspects, or any combination of two aspects. This way, the effect of lower scores is reduced compared to higher scores.

Shortly after the development of $funSim$, Schlicker et al. [2007b] also developed another approach, called $rFunSim$, which is the square root of $funSim$. Their justification for this development is that $funSim$ can generate counter-intuitively low results. They also demonstrated that the calibration error, the absolute difference between predicted confidence and actual reliability [Sing et al., 2005], is smaller for the $rFunSim$ score than for $funSim$. This means that simply taking the square root of $funSim$ improves the performance of the score.

In this study, the performance of individual ontology scores will be compared to the $rFunSim$ score to see if one approach performs better than the others.

3.1.4 Evidence codes

Another consideration when studying semantic similarity approaches is the nature of the annotation. As discussed in Section 2.1.2, each GO annotation is associated with an evidence code in order to describe how the annotation was derived. The GO guidance notes[2] state that no evidence code can be used to assess the quality of the annotation, as some methods of classification within an evidence code category produce more accurate or specific annotations than others. Nonetheless, some researchers prefer to use only specific evidence codes, such as "TAS" (Traceable Author Statement) or "IDA" (Inferred from Direct Assay), while others exclude certain annotations, particularly electronic annotations (evidence code "IEA" - Inferred from Electronic Annotation). On one hand, electronic annotation is considered to be less reliable than manually curated annotation but on the other hand, it accounts for over 50% of all annotations in the GO (1.6 million electronic annotations out of 2.6 million total annotations in GO release 072010), although only about 40% of the species in the GO have any electronic annotation. Even though ignoring electronic annotation may increase the reliability of the annotation, it considerably reduces the richness of the annotation for certain species.

The dataset exploited in the current study consists entirely of yeast (*S. cerevisiae*) genes (see Section 3.1.5), which in the GO release 072010 has 41678 electronic annotations out of a total of 89841 annotations. In yeast, all non-electronic annotations are derived in one of four possible ways:

- Mapping of SWISS-Prot keywords. SwissProt keywords are mapped manually

[2]http://www.geneontology.org/GO.evidence.shtml, accessed 17/08/2010

to GO terms in a project started by MGI and now maintained by the GOA team at EBI [Camon et al., 2003]. Any database entry that has one or more SWISS-Prot keywords assigned to it can therefore be annotated with GO terms via the mapping file. Annotation with SwissProt keywords is done manually and is generally considered to be of high quality.[3]

- Mapping of InterPro domains. InterPro domains are mapped manually to GO terms by the InterPro team at EBI [Camon et al., 2003]. The mapping is then used to automatically annotate any database entry associated with one or more InterPro domains with the corresponding GO terms.[4]

- Mapping based on Swiss-Prot Subcellular Location vocabulary annotation. A subcellular location vocabulary developed by the UniProt consortium was mapped manually to GO terms by the GOA curators at EBI. Any UniProtKB entry with SPSL annotation can thus be annotated with the appropriate GO terms.[5]

- Mapping using Enzyme Commission identifiers. Any database entry that has an EC number assigned can be annotated with a corresponding GO term [Camon et al., 2003], which is determined using the EC cross-references in the GO molecular function ontology, as described in Hill et al. [2001].[6]

With the exception of the fourth approach, each of these is based on a manually curated mapping file, so yeast electronic annotation can reasonably be expected to have a similar level of accuracy as any of the non-electronic annotations. Of course, this does not necessarily apply to other species. In the present study however, the use of both full and non-electronic annotation datasets can be justified, in order to compare the performances of semantic and functional similarity methods for either dataset.

[3] http://www.yeastgenome.org/cgi-bin/reference/reference.pl?dbid=S000124038, accessed 17/08/2010

[4] http://www.yeastgenome.org/cgi-bin/reference/reference.pl?dbid=S000124036, accessed 17/08/2010

[5] http://www.yeastgenome.org/cgi-bin/reference/reference.pl?dbid=S000125578, accessed 17/08/2010

[6] http://www.yeastgenome.org/cgi-bin/reference/reference.pl?dbid=S000124037, accessed 17/08/2010

3.1.5 The dataset

Choice of organism

The fungus *Saccharomyces cerevisiae* (or "Baker's yeast") is a species of budding yeast, a family of simple unicellular eukaryotic organisms. It can be obtained cheaply in large quantities, is non-pathogenic and easy to keep in a laboratory and has a short generation time (doubling takes 1.5 - 2 hours at 30 °C). All of these facts make yeast a perfect model organism for the study of eukaryotic genomes. The *S.cerevisiae* genome was the first eukaryotic genome to be fully sequenced, completion of sequencing having been announced on 24 April 1996 [National Human Genome Research Institute, 1996; Dujon, 1996].

The yeast genome consists of about 12.4 million base pairs, distributed across 16 chromosomes. This genome is around three times larger than the *E. coli* genome (4.6 million base pairs) and 250 times smaller than the human genome (3.4 billion base pairs). Yet despite this large size difference between the human and yeast genomes in terms of base pairs, the difference in number of genes is only about 3-fold, with the yeast genome containing around 6700 protein-coding genes[7] compared to around 21200 in humans[8].

The much smaller gene number to genome size ratio in yeast is due to its more economical genome organisation. There is very little non-coding DNA, with only 239 known introns. 4-5% of yeast genes are discontinuous, the rest consist of uninterrupted sequences of coding DNA. There are also very few genome-wide repeats, accounting for only 3.4% of the genome. A comparison of genome features between yeast and human is given in Table 3.1.

The yeast genome project lists 6569 predicted Open Reading Frames (ORFs), which include "verified" and "uncharacterised" as well as "dubious" frames. Of these, 5749 (verified and uncharacterised only) are currently classified to be ORFs for protein encoding genes whose expression has been confirmed.

As the yeast genome is the longest studied eukaryotic genome, a wealth of annotation data is available for it. This makes yeast an excellent candidate genome for a study of gene product annotation.

The Eisen dataset

In 1998, Michael B. Eisen and colleagues published a paper entitled "Cluster analysis and display of genome-wide expression patterns" [Eisen et al., 1998]. In this paper,

[7] Genome assembly EF 2, Feb 2010
[8] Genome assembly GRCh37, Feb 2009

	Yeast	Human
Genome size	12.4M bp	3.4G bp
No. of chromosomes (diploid)	32	46
No. of genes	6700	21200
No. of introns	239	300000
Genome-wide repeats	3.4% of genome	44% of genome

Table 3.1: Comparison of *S. cerevisiae* and human genome features

the authors proposed to use standard statistical algorithms to cluster the results of genome-wide microarray experiments, in order to discover gene expression patterns. In order to demonstrate their approach, they used two distinct datasets (one human, one yeast).

The *S.cerevisiae* dataset in question became widely known as the "Eisen dataset" and has since been used by many researchers, including several of the works cited in Chapter 2, including Wang et al. [2004]; Yu et al. [2007]; Xu et al. [2008]; Jing et al. [2010]. As a result, a wealth or analysis data for the Eisen dataset is available. This makes it a particularly suitable dataset for evaluating novel functional analysis approaches as the original authors' findings have been re-evaluated and thus validated many times, as well as improved on with advances in existing knowledge.

The Eisen dataset is actually a collection of four studies on diauxic shift [DeRisi et al., 1997], mitotic cell division cycle [Spellman et al., 1998], sporulation [Chu et al., 1998] and temperature shock processes (unpublished results). These expression studies covered all ORFs available at the time. The Eisen dataset itself consists of 2466 ORFs, which is all the ORFs for which function annotations were available in 1998.

Although a much larger proportion of the yeast genome is now functionally characterised, it is still desirable to focus on the 2466 ORFs analysed by Eisen et al. as these represent the best-studied genes in the yeast genome. It also allows a full comparison of the results obtained in this project with the results of Eisen et al.'s study and any other subsequent studies on the same dataset.

3.1.6 The grouping algorithm

We propose FuSiGroups, a novel approach to group functionally similar gene products into groups based on semantically similar GO terms. Before describing the actual algorithm, it is necessary to define a number of concepts:

Definitions

Definition 1. *Group definition - a set of related GO terms that represent the functional aspect of the group.*

Definition 2. *Group content - a set of related gene products that share the functional aspect represented by the group definition.*

Definition 3. *Group name - the lowest common ancestor of all the GO terms in the group definition.*

Definition 4. *Semantic threshold (ST) - an empirically determined semantic similarity value which represents the minimum level of similarity any two GO terms must have in order to occur in the same group definition.*

Definition 5. *Functional threshold (FT) - an empirically determined functional similarity value which represents the minimum level of similarity any two gene products must have in order to qualify for membership in the same group content.*

Definition 6. *Meaningful group - a group that contains 4 or more gene products. Let n be the number of gene products g in $groupContent_G$ of group G. G is a meaningful group if $n \geq 4$.*

Algorithm description

The GO terms form the *group definition* and represent a set of concepts that the gene products in the group have in common. The gene products represent the *group content*. Each group has a *group name*, which is an ancestor term of all the GO terms in the definition, and which characterises the functional aspect represented by the group.

Both group definition and group content are based on the concept of maximally complete graphs or cliques [Valiente, 2002], i.e. all GO terms of the definition and gene products of the content represent nodes that have to be connected to all other nodes in the group by an edge. In this context, an edge is defined as the semantic similarity between two GO terms being equal to or greater than a given *Semantic Threshold* (ST), or the functional similarity between two gene products being equal to or greater than a given *Functional Threshold* (FT).

The rationale for basing the groups on a clique model is that it allows elements to be in multiple groups at the same time, provided they are connected to all elements in each group. This is different from the traditional clustering approaches used in bioinformatics, which are more related to connected components, i.e. models

where each element of a group does not have to be directly related to all other elements but can be intransitively related via other elements, i.e. by paths of two or more edges. In clustering approaches, each element is usually only allowed to be in exactly one cluster at a time. This is generally sufficient in applications such as the clustering of gene products based on expression profiles, as this type of data only considers a single dimension of a gene product's function. Functional similarity on the other hand is by its nature multidimensional, as it characterises the multiple different functions that most gene products fulfil, in the different processes they are involved in. Most gene products can have elements of functional similarity with different non-overlapping sets of other gene products. Grouping gene products based on these multiple facets of their functions allows for better understanding of these functions and the relationships between gene products.

An exception to the single cluster membership is fuzzy clustering, a fuzzy logic-based type of approach which assigns each element a cluster membership indicating the similarity of that element to each cluster, so that each element can belong to multiple clusters [Bezdek, 1981]. Although not as widely used in bioinformatics as hierarchical and k-means clustering, fuzzy clustering approaches have been shown to perform well and even better than traditional clustering approaches [Gasch and Eisen, 2002; Kim and Choi, 2005; Do and Choi, 2007]. However, fuzzy clustering algorithms such as fuzzy c-means [Bezdek, 1981] have the disadvantage that the number of clusters in the solution needs to be pre-specified before the algorithm is run. This is undesirable in a knowledge-discovery process as pre-specifying the number of solution clusters forces the user to form preconceptions about the data and its underlying structure and can prevent a full investigation.

The FuSiGroups algorithm addresses some of the drawbacks of traditional clustering approaches when applied to functional similarity data. Although the algorithm makes use of the clique model, it does not include any existing algorithms addressing the clique problem as these are generally NP-complete and therefore costly to run in terms of time and computational resources [Valiente, 2002].

Conceptual model

The grouping process is performed as follows: semantic similarity between all possible pairs of GO terms annotated to a given set of gene products is calculated according to a given approach, e.g. Lin, Resnik or Schlicker. Based on these semantic similarities, functional similarity between all pairs of gene products in the set is calculated using one of BMA, MAX or AVG.

The first stage of the grouping process creates the group definition. For each available GO term, a group is created and GO terms with semantic similarity to the central term above the ST are added to the group. Then the connections between all the GO terms in the group are checked against the ST and any GO terms that do not conform to the maximally complete graph rule (i.e. have similarity smaller than ST to some of the other terms) are removed from the group, starting with the least connected GO term. If there is more than one term with the highest level of disconnectedness, the first term in the list is removed without consideration of other criteria, such as term depth or average similarity to other terms in group.

When all the groups have been created, any groups with definitions that are subsets of other group definitions are removed to avoid redundancy. Finally, each group is named with the lowest common ancestor (LCA) of the GO terms that make up its definition. We differentiate here between a set of terms' LCA and their MICA. Although in most cases LCA and MICA are going to be the same, the LCA is the term in the set of common ancestors with the highest distance to the root rather than the highest information content. In the case of multiple ancestors with the same distance, one is chosen at random. The naming of the group using the LCA is for simplicity of processing by the user as it shows the overall ontological aspect of the group. The naming completes the first stage of the grouping process.

The second stage of the grouping process creates the group content. Each gene product in the list to be analysed is added to any group that has a term annotated to this gene product in its definition. When all gene products have been processed, each group's content is checked for violation of the maximally complete graph rule against the FT. Gene products that do not conform to the rule are removed from the group, starting with the least connected gene product, until a maximally complete graph is reached. Again, removal of a term in the case of more than one term with the same highest level of disconnectedness is done by removing the first term in the list of equally disconnected terms.

Due to the nature of the grouping process, not all gene products in a given list are necessarily included in at least one group. In addition, many groups may only contain one or two gene products. These are considered to be non-informative groups. We define an informative or meaningful group as a group containing at least four gene products (see Definition 6). This means that groups containing exactly one, two or three gene products will be excluded from the analysis. The exclusion of groups with only one gene product is an obvious step as the purpose of FuSiGroups is to investigate the functional relationships between gene products and a single-gene product group does not show any functional relationships.

The cut-off of four gene products was chosen because it was considered unlikely for groups of two or three gene products to reveal any unexpected or novel relationships not found by other means such as directly looking up functional similarity scores. In addition, considering the total number of groups generally generated by the FuSiGroups algorithm (see Table 6.2), it was deemed acceptable to exclude groups with only two or three gene products in order to simplify the analysis process. This is particularly valid as it can be reasonably expected that most groups with only two or three gene products will bring together gene products of very high similarity, such as sub-units of a protein or protein complex, where functional similarity is known and therefore does not bring novel insights.

This minimum group size is consistent with that defined by Huang et al. [2007], who stated that in order to be a cluster seed, a gene had to be closely related to at least three other genes. The same definition is not applied to the group definition, as groups with only one term in their definition may still group together a set of interesting gene products.

It should also be noted that groups with one, two or three gene products are still created by the algorithm and stored in the same way as the other groups. They are merely excluded from the analysis process because they are considered uninformative. Should the re-inclusion of these small groups appear of interest at any point during the course of the analysis, this is of course always possible.

Pseudocode

The pseudocode for the grouping algorithm is given in Table 3.2.

initialise list $allGOTerms$
initialise list $allGenes$
initialise list $allGroups$
Group definition
    FOR ALL $t \in allGOTerms$ DO
        create group G for t
        add G to $allGroups$
        FOR ALL $g \in allGOTerms - t$ DO
            IF $sim(t, g) \geq ST$ THEN
                add g to $groupDef_G$
            END IF
        END FOR
        WHILE completeness rule ! = TRUE DO
            create $list${nodes that violate completeness rule with d}
            FOR ALL $d \in groupDef_G$ DO
                FOR ALL $f \in groupDef_G$ DO
                    IF $sim(d, f) < ST$ THEN
                        add f to $list${nodes that violate completeness rule with d}
                    END IF
                END FOR
            END FOR
            IF$list${nodes that violate completeness rule with d} is empty THEN
              completeness rule = TRUE
            END IF
            ELSE
                remove node with largest number of rule violations
            END ELSE
        END WHILE
    END FOR
    FOR ALL $G_1 \in allGroups$ DO
        WHILE ! $removeG_1$ || ! $done$ DO
            FOR ALL $G_2 \in allGroups - G_1$ DO
                IF $G_1 \subset G_2$ THEN
                    remove G_1 from $allGroups$
                    $removeG_1$ = TRUE
                END IF
            END FOR
            $done$ = TRUE
        END WHILE
    END FOR
RETURN allGroups

```
Group content
    FOR ALL g ∈ allGenes DO
      FOR ALL G ∈ allGroups DO
        IFt ∈ annotation_g && t ∈ groupDef_G THEN
          add g to groupContent_G
        END IF
      END FOR
    END FOR
    FOR ALL G ∈ allGroups DO
      WHILE completeness rule ! = TRUE DO
        FOR ALL g_1 ∈ groupContent_G DO
          FOR ALL g_2 ∈ groupContent_G DO
            IF sim(g_1, g_2) < FT THEN
              add g_2 to list{nodes that violate completeness rule with g_1}
            END IF
          END FOR
        END FOR
        IFlist{nodes that violate completeness rule with d} is empty THEN
          completeness rule = TRUE
        END IF
        ELSE
          remove node with largest number of rule violations
        END ELSE
      END WHILE
    END FOR
RETURN allGroups
```

Table 3.2: Pseudocode for the grouping algorithm

Summary

In this section, the novel FuSiGroups algorithm was introduced. The algorithm first creates group definitions of GO terms based on their semantic similarity, using a semantic threshold ST to ensure that all terms in a group's definition are sufficiently similar to all other terms in the group. Then gene products annotated with the terms in a group's definition are added to that group, before the functional similarity between all pairs of gene products in a group is matched against a functional threshold FT to make sure that all gene products in that group are related. In the next section, the strategy to evaluate the FuSiGroups algorithm is presented, including the process of how to determine optimal semantic and functional thresholds.

3.2 Evaluation strategy

3.2.1 Semantic and functional similarity approaches

As discussed in Section 2.4, it is very difficult to evaluate semantic and functional similarity approaches as there are no established benchmarks. Evaluation against any other form of biological similarity requires assumptions to be made about functional similarity. For this reason, it is recommended to use multiple standards against which to evaluate the measures [Xu et al., 2008]. Receiver operating characteristic (ROC) [Fawcett, 2006] curves have been chosen to determine which semantic and functional similarity approaches and annotation types are best suited to characterise the similarity between gene products.

ROC curves

ROC curves show the trade-off between sensitivity (or true positives) and specificity (or true negatives) of a binary classification system (e.g. true-false) at varying thresholds of the associated discrimination measure. They can be used to visualise and evaluate the performance of different classifiers. [Fawcett, 2006] Originally developed during World War II to improve the identification of enemy radar signals, ROC curves are now commonly used in signal detection, psychophysics, medical diagnostics, machine learning and data mining. [Green and Swets, 1966; Swets, 1988; Lasko et al., 2005]

In any test involving a discrete classifier (true/false, healthy/diseased, related/unrelated, etc.), the accuracy of a test in splitting a population into two classes corresponding to the two discrete results can be measured using the concepts defined in Table 3.3. For the actual ROC curve, sensitivity SN is plotted against false positive rate $1 - SP$, with each point on the curve representing the $TP - FP$ (true positive-false positive) trade-off at a given threshold.

An example of the different ROC curves is shown in Figure 3.1. In the case of a perfect classification measure, the ROC curve would rise in a straight vertical line from the bottom left to the top left corner, and then go across to the top right corner in a straight horizontal line (black dotted line in Figure 3.1). The diagonal (bottom left to top right, blue dashed line in Figure 3.1), also called the "line of no discrimination", corresponds to a "random guess" situation, whereas a curve running below the diagonal would correspond to a "worse than random" method (yellow dashed-dotted line in Figure 3.1). Curves between the diagonal and perfect classifications (solid red and green lines in Figure 3.1) are generally considered as

Name	Symbol	Definition
True Positive	TP	correct acceptance
True Negative	TN	correct rejection
False Positive	FP	false acceptance
False Negative	FN	false rejection
Positive dataset	P	$TP + FN$
Negative dataset	N	$TN + FP$
Sensitivity	SN	$\frac{TP}{TP+FN}$
Specificty	SP	$\frac{TN}{TN+FP}$
False positive rate	$1 - SP$	$\frac{FP}{TN+FP}$
Accuracy	ACC	$\frac{TP+TN}{P+N}$

Table 3.3: Common concepts in ROC curve analysis

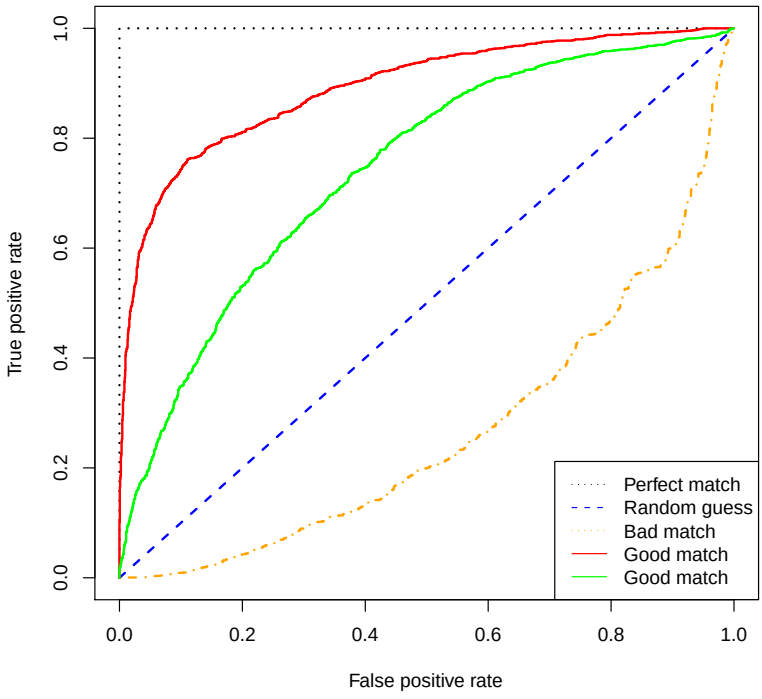

Figure 3.1: Example of different possible ROC curves

"good", although quality requirements vary with usage.

While ROC curves are in themselves a great way of visually evaluating the effectiveness of one measure and comparing the results obtained from different measures, it is usually desirable to have a single index to quantify the performance of each measure. This is particularly the case if there are only small variations between a set of curves. A number of such indexes are available for ROC curves. See Swets [1988]; Lasko et al. [2005]; Fawcett [2006] for details. The most commonly used measure is the "Area Under the Curve" (AUC) [Hanley and McNeil, 1982]. The AUC of a classifier represents the likelihood that for a given classifier, a randomly chosen positive obtains a better rank than a randomly chosen negative. This is equivalent to the Wilcoxon test of ranks [Hanley and McNeil, 1982].

The AUC of a perfect result is 1, the AUC of the line of no discrimination is 0.5. An AUC of less than 0.5 indicates a failed test. AUCs between 0.5 and 1 are generally considered to be "good", with larger AUCs being preferable. As with the overall interpretation of a ROC curve, the quality of an AUC is dependent on the context of its use.

In the present analysis, three different types of true positive datasets are used. They are derived from gene expression data, protein interaction data and phenotype data, respectively.

Gene expression data

For the gene expression data, the true positive dataset was created by clustering the centered and normalised Eisen expression data using agglomerative hierarchical clustering with Pearson's correlation and average linkage. The resulting tree was then cut at a very low level (height = 0.1 for a tree with height range [0,1]). Genes which at that level were clustered together were considered to be very closely related and therefore, gene pairs were created from these tight clusters. Many of these clusters contained only two genes, creating obvious pairs. In the few larger clusters, all distinct combinations of pairs were created. Pairs are non-directional, i.e. (A,B) = (B,A). Of the resulting 1359 distinct gene pairs, 1260 were randomly selected to form the positive dataset. The reason for the reduction in size of the dataset was to introduce an element of random selection into the data.

A true negative dataset of equal size was created by pairing each of the 37 left-most genes in the hierarchical tree with each of the 37 right-most genes in the tree. It was deemed that the difference in expression levels reflected in the tree was sufficient at the two extremes of the tree to ensure that any two genes from these

two ends would not be significantly related from a gene expression standpoint. Of the resulting 1369 gene pairs, 1260 were again selected randomly to form the true negative dataset.

Protein interaction data

For the protein interaction data, all available yeast protein interaction data was downloaded from the SGD FTP site[9] (April 2010). The complete dataset was filtered using the following sequence of criteria:

1. Interaction type - physical interactions only. The dataset consists of physical and genetic interactions, but only direct physical interactions between proteins were considered for this dataset

2. Experiment type - affinity capture - mass spectrometry only. Of all the available experiment types, mass spectrometry was considered the most reliable. For dataset consistency, only one experiment type was chosen.

3. Curation - manual curation only. Manually curated interaction data was considered to be more reliable than interaction pairs derived from high-throughput experiments.

4. Bait/hit protein - proteins present in the Eisen dataset. Only interactions for which both the bait and hit protein can be found in the Eisen dataset were selected as functional similarity had only been calculated for the Eisen dataset.

These selection steps resulted in 1961 distinct protein pairs of which 1745 were randomly selected for the true positive dataset.

The true negative dataset, also consisting of 1745 protein pairs, was created through random selection of pairs from the Eisen dataset. All selected pairs were checked against the full (pre-filtering) interaction dataset to ensure that no known interaction, physical or genetic, existed for each pair. According to Guo et al. [2006], it is very unlikely that a randomly selected pair of proteins has unknown interactions.

Phenotype data

For the phenotype data, all available yeast phenotype data [Engel et al., 2010] was downloaded from the SGD FTP site (April 2010). Only data entries which corresponded to ORFs present in the Eisen dataset were used. This corresponded to

[9] http://downloads.yeastgenome.org/, accessed April 2010

27921 distinct data entries for 2438 ORFs. For the sake of consistency, it was decided that all phenotypes used should be derived from the same mutation type. The most common mutation type in the dataset (76.57% of entries) was "null", i.e. complete loss of function either through a point mutation or deletion of part or all of a gene. As the phenotype data was derived from experiments using different yeast strains, only phenotypes for the most common strain, "S288C" (80.57% of entries), were used.

In order to quantify phenotype-based similarity between gene products, a vector space model (VSM) was used [Baeza-Yates and Ribeiro-Neto, 1999]. Each gene product g is represented as a vector of all phenotypes,

$$g = (p_1, p_2, \ldots, p_n),$$

where p_i is the numeric value of phenotype i for g, e.g. $p_i = 0$ if there is no association between phenotype i and gene product g.

For the purpose of this similarity calculation, a distinct "phenotype" consists of the actual phenotype description as well as any chemical and other experimental condition associated with that phenotype, for example "resistance to chemicals: decreased - ethanol (10%)" consists of the phenotype "resistance to chemicals: decreased" and the chemical "ethanol (10%)", and is distinct from "resistance to chemicals: decreased - methyl methanesulfonate (0.2%)", consisting of the same phenotype but the chemical "methyl methanesulfonate (0.2%)". This combination of phenotype description and experimental conditions is particularly important for phenotypes such as "resistance to chemicals: decreased", which are associated with a wide range of different chemicals, or phenotypes that are associated with different temperature or growth medium conditions, as the experimental conditions provide detail about the very different forms of a given phenotype.

As different phenotypes occur with different frequencies, it is advisable to weight their contribution to the vector, giving

$$g = (w_1, w_2, \ldots, w_n)$$

For this purpose, "Inverse Document Frequency" (idf) is used, so that

$$w_p = idf_p = log\frac{N}{n_p}$$

where w_p is the weight of phenotype p, N is the total number of gene products and n_p is the number of gene products annotated with phenotype p. idf is commonly used in VSM-based information retrieval and is in fact similar in concept to information content in semantic similarity.

The similarity between a pair of gene product vectors g_1 and g_2, $sim(g_1, g_2)$ is calculated as the cosine of the angle between these two vectors:

$$sim(g_1, g_2) = \frac{g_1 \bullet g_2}{\|g_1\|\|g_2\|} = \frac{\sum_{i=1}^{N}(w_{ig_1} \cdot w_{ig_2})}{\sqrt{\sum_{i=1}^{N}(w_{ig_1}^2)}\sqrt{\sum_{i=1}^{N}(w_{ig_2}^2)}}$$

where $g_1 \bullet g_2$ is the dot product of vectors g_1 and g_2 and $\|g_1\|\|g_2\|$ is the product of the magnitude or norm of g_1 and g_2. This approach is known as "cosine similarity" or "cosine normalisation" [Baeza-Yates and Ribeiro-Neto, 1999; Chabalier et al., 2007].

However, initial test runs suggested that even after weighting, certain phenotypes were so common that their contribution was meaningless. In addition, many gene products were associated only with one phenotype, in many cases a generic one such as "viable" or "inviable". For these reasons, an additional filtering step was inserted prior to the creation of the gene product vectors. Only gene products associated with at least three phenotypes were included in the calculation and any phenotype annotated to more than 250 gene products even as "phenotype - chemical - condition" combination was excluded as being insufficiently meaningful. Note that these phenotypes were excluded completely, not just in the respective combinations.

Additionally, gene products associated with fewer than three phenotypes were excluded from the similarity calculation. The reason for this is to avoid pairs of gene products with apparently high similarity that in fact only have a very common phenotype in common, without actually having any significant functional similarity. The previous filtering step had already removed many of the gene products with the targeted annotation frequencies.

It should be noted that some gene products may have a lower similarity than would be expected. This is due to the fact that the data in the SGD phenotype dataset is not always consistent, with entries such as "37 deg", "37 deg C", "37 degrees" etc associated with the same phenotype description. These inconsistencies, although easily spotted and interpreted by a human, are classed as different by the direct string matching algorithm used in this approach, resulting in different phenotype combinations for what should in reality be one phenotype combination.

After calculating phenotype similarity between all pairs of gene products under these restrictions, 2876 pairs with a similarity of 1 were obtained. Of these, 2000 were randomly selected to form the true positive dataset. The corresponding true negative dataset was created by randomly selecting 2000 gene product pairs with a phenotype similarity of 0.

Data processing

Each of the three datasets was uploaded to the "R software environment for statistical computing" (hereafter referred to simply as R) [R Development Core Team, 2010]. All data processing for ROC curves and AUC calculations was done using the ROCR library [Sing et al., 2005].

In order to allow comparison both within each dataset (bootstrapping analysis) and across the three datasets, each dataset was randomly resampled into ten subsets of 500 true positive and 500 true negative observations each. Resampling was done without replacement within each subset but entries could be used in multiple subsets. Each subset then generated a slightly different ROC curve. By averaging these curves both horizontally and vertically, it is possible to obtain a measure of variance for each dataset, which in turn makes the comparisons between datasets more accurate.

ROC curves were generated and AUCs computed for all combinations of approaches and for the following scores: "MF only", "BP only", "CC only", "funSim", "rFunSim". It is important to note that as rFunSim is the square root of funSim, the ROC curves for the two scores for a given dataset have the same AUC, i.e. their ROC curves have the same shape. The thresholds for each data point on the curve are however different, i.e. a given data point on the ROC curve for rFunSim has a threshold that is the square root of the threshold for that same data point on the equivalent funSim curve.

This property of the ROC curves also allows comparisons to be drawn between thresholds for different similarity approaches. Functional similarity for Resnik for a given dataset is not the same as for example functional similarity for Lin. By comparing the shape of the ROC curves, approximations can be made about corresponding thresholds, e.g. functional similarity according to Resnik of 0.3 might have the same level of accuracy as functional similarity according to Lin of 0.8.

Full results of these experiments will be discussed in Chapter 4. The semantic and functional similarity approaches, and other associated parameters, that are found to perform the best will be carried forward for use in the FuSiGroups algorithm. This reduction in the number of approaches is necessary to keep the subsequent analysis to a manageable size.

3.2.2 Threshold determination

Overall strategy

Once the best semantic and functional similarity approaches and other associated parameters have been selected, they will be tested in conjunction with the FuSiGroups algorithm. As the FuSiGroups algorithm includes two variable thresholds, the "semantic threshold" (ST) and the "functional threshold" (FT), it is first of all necessary to determine the optimal threshold ranges for each approach.

In addition to the ROC capabilities described above, R's ROCR library can also generate accuracy graphs (see Table 3.3 for definition of accuracy). An accuracy graph shows the predictive ability of each cut-off, i.e. the ability of a given approach to distinguish between true positives and true negatives at each point of its range of values. An example of an accuracy graph is given in Figure 3.2. Note that the graph's y-axis, representing the accuracy, ranges from 0.5 to 0.8 (or 50% to 80% accuracy). Accuracies of less than 0.5 are not usually found as they would represent a worse than random approach.

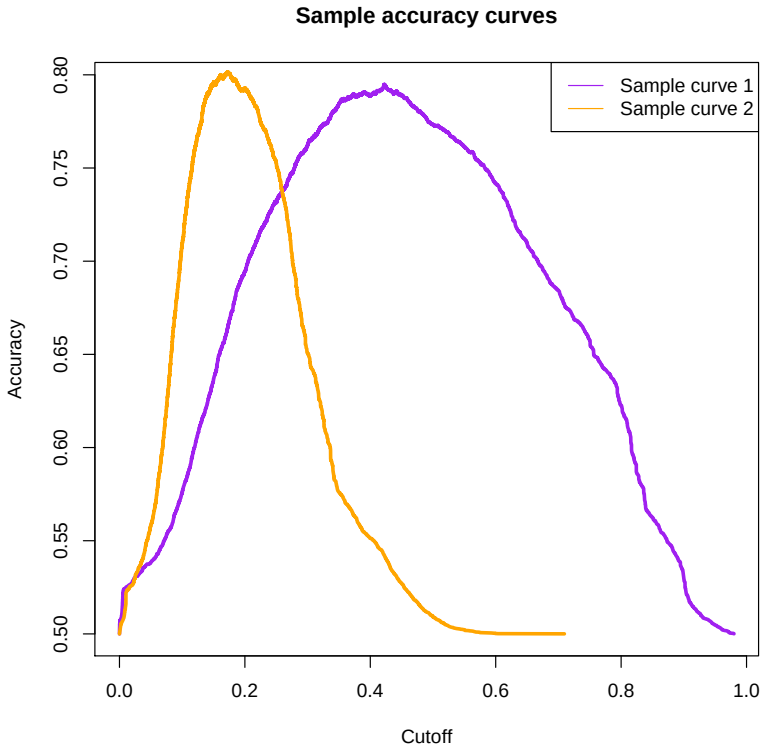

Figure 3.2: Example of different possible accuracy curves

In order to determine a good range of thresholds, the following concepts are defined:

Definition 7. *Minimum threshold - semantic or functional similarity value that achieves the highest accuracy for a given approach. Let* $accuracy = f(similarity) \Leftrightarrow similarity = f^{-1}(accuracy)$. *Minimum threshold* $= f^{-1}(max(accuracy))$.

Definition 8. *Maximum threshold - semantic or functional similarity value that is greater than the minimum threshold and corresponds to an accuracy, rounded to the nearest 0.05, of the maximum accuracy minus 15% of the range of accuracy values for a given approach. Let* $accuracy = f(similarity) \Leftrightarrow similarity = f^{-1}(accuracy)$ *and let* $r = max(accuracy) - min(accuracy)$. *Maximum threshold* $= f^{-1}(max(accuracy) - (r * 0.15))$.

The definition of the minimum threshold is based on the assumption that it is more desirable to exclude some true positives from the groups generated by the FuSiGroups algorithm than to include false positives and the maximum accuracy represents the best possible trade-off between true positives and true negatives for a given approach. The maximum threshold definition was derived from the need to minimise the number of false positives while at the same time obtaining a threshold that is distinct from the minimum threshold. 15% of the range of accuracy values was determined as the point fulfilling these criteria from the analysis of a number of different datasets. The rounding to the nearest 0.05 was included for ease of analysis. The actual accuracy values for the thresholds calculated in this work are derived in Chapter 5.

Functional thresholds

The thresholds for the functional similarity approaches can be derived directly from the functional similarity data. The 30 (3 times 10) sub-datasets from the resampling described in Section 3.2.1 are analysed in parallel and 30 accuracy curves are obtained which can be assimilated into a single curve using vertical threshold averaging. Unfortunately, while it is possible to obtain a single curve on a graph, this approach does not allow the extraction of a single similarity value for the minimum and maximum thresholds as the underlying data is not averaged. Although deriving the thresholds visually from the curve would be a possibility, the quality of the curves was judged to be too low for this approach. The problem can however be solved by aggregating the 30 subsets into one big dataset. This results in a single accuracy curve identical to the vertically averaged individual curves, i.e. there is no loss of precision from the 30 sub-datasets, but with specific x and y values available for each point on the curve. As there is no difference in the actual curves derived from either approach but the aggregate dataset gives more scientifically accurate

data points than visual analysis, the aggregate dataset was chosen for the present analysis.

For each selected functional similarity approach, the cut-off (x-value) corresponding to the highest accuracy data point (y-value) is selected to obtain the minimum functional threshold. Then the data point with the largest cut-off corresponding to an accuracy of 15% of the accuracy range is selected to obtain the maximum functional threshold. For the two curves in Figure 3.2 for example, the accuracy values lie between about 0.5 and 0.8, i.e. a range of 0.3, so 15% of the range is 0.05, leading to an accuracy of 0.75.

Semantic thresholds

Semantic thresholds are not as easy to establish as functional thresholds as they cannot be derived directly from the data. The semantic threshold determines the appropriate level of semantic similarity between the GO terms that make up a group's definition. The true positive and true negative datasets constructed for the functional similarity analysis are based on gene products, related to a range of GO terms. At present, there is no equivalent dataset of GO terms qualified as similar or related based on a given property that is not semantic similarity. At best, such a dataset could be generated by a human curator using expert understanding, a laborious task for a dataset of sufficient size (1000+ term pairs). However, such a dataset would still be based on semantic relatedness rather than an independently verifiable property.

For this reason, semantic thresholds need to be determined using an indirect approach. The "MAX" functional similarity approach selects the GO term pair with the highest semantic similarity from a set of term pairs. This single most similar term pair is the closest that the gene products in the positive and negative datasets can be related to the GO terms on which their functional similarity is based. On the assumption that two biologically related gene products are most likely to be annotated with highly similar GO terms, while two unrelated gene products are most likely to be annotated with equally unrelated GO terms, the "MAX" functional similarity scores for the individual sub-ontologies are used to establish the semantic thresholds for each approach.

FuSiGroups uses only one semantic threshold for all GO term pairs but there are three GO ontologies, i.e. three sets of minimum and maximum thresholds can be deduced. There are two ways of addressing this issue. One option is to deduce three sets of thresholds using the method described for the functional thresholds, then to

average the three thresholds into a single one. This approach is justifiable, as it is fairly common that two gene products have highly similar annotations in one or two ontological categories, but not in the other(s). An average of the three thresholds would therefore present a balanced overall threshold. A second option is to use a similar process to the one used to reduce the 30 sub-datasets into a single dataset. The datasets for the three ontological scores are aggregated into a single very large dataset which, after performing the usual analysis on it, generates a single accuracy curve. The minimum and maximum semantic thresholds can then be deduced from this curve in the same way as the functional thresholds.

The threshold determinations will be discussed in Chapter 5. Once the semantic and functional thresholds have been determined, combinations of a range of semantic and functional thresholds can be run for the best performing semantic and functional similarity approaches. The resulting groups can then be analysed using the strategy described in the next section.

3.2.3 FuSiGroups grouping results

Grouping trends

The analysis of the grouping results will be divided into two parts. The first part, which will be discussed in Chapter 6, considers, for different approaches and thresholds, the overall trends of the results, such as group sizes, definition sizes and the distribution of the groups across the three GO ontologies, among others. The purpose of this is to establish an overview of the grouping results generated by the FuSiGroups algorithm, in particular in relation to the performance of the individual semantic and functional similarity approaches, at their respective minimum and maximum thresholds. The combination of semantic and functional similarity approaches and ST and FT that gives the most promising results at this stage will be used for the second stage of the groups analysis.

Groups analysis

The problems surrounding the evaluation of functional similarity have been discussed at lengths in Sections 2.4 and 3.2.1. They apply equally to applications of functional similarity such as the FuSiGroups algorithm. Any attempt to evaluate the groups generated by the algorithm against another form of biological similarity inherently makes assumptions about the content and nature of the groups. As in previous cases, a multi-stage strategy is adopted to evaluate the content of the groups.

First of all, the largest groups and most common group names are considered in order to establish whether they provide any interesting insight about common functions of the genes in the dataset. In a large dataset, such as the Eisen dataset, this step may not provide useful results as the dataset may be too diverse and noisy but in the interest of thoroughness, this step should not be omitted.

As the Eisen dataset was originally used for the study of functional relatedness based on gene expression similarity, it would be interesting to relate the grouping results back to the gene expression clusters identified by Eisen et al. [1998]. Since there is no direct equivalent between functional and expression similarity, it would be neither useful nor appropriate to simply compare FuSiGroups groups to expression clusters. For this reason, the Eisen dataset is also clustered based on functional similarity in order to compare functional and expression clusters. The similarity between clusters derived from the Eisen expression data and clusters derived from the corresponding functional similarity data is evaluated at different clustering levels using external cluster validation techniques, such as purity, F-measure, normalised entropy and mutual information [Handl et al., 2005; Jakonienė et al., 2006]. In these measures, the expression clusters are used as classes for the functional clusters and vice versa. The expression clusters can then be compared to the functional groups via the intermediate of the functional clusters.

Finally, three smaller subsets of the Eisen dataset, a proteasome dataset, a ribosome dataset and a dataset of two biochemical superpathways, will be created in order to perform a more detailed analysis of the content of the resulting groups. The reason for selecting smaller subsets is that the FuSiGroups algorithm, applied to a dataset the size of the Eisen dataset, produces anything from a few hundred to a few thousand groups (depending on thresholds and other variables). Even after elimination of "meaningless" groups, i.e. groups with too few gene products, the number of groups is too numerous to perform a detailed analysis of each group. All meaningful groups in each sub-dataset will be analysed in detail and in relation to existing literature in order to establish whether the inclusion of the genes in the group is appropriate, both in relation to the other genes and in relation to the functional aspect described in the group definition. All steps of the groups' analysis will be discussed in Chapter 8.

3.3 Implementation considerations

In this section, a few details regarding the implementation of the FuSiGroups algorithm as well as practical aspects of the dataset used and the experiments that were

performed are discussed. As the implementation is at this stage for proof-of-concept purposes only, no coding details will be discussed and no consideration will be given to runtime and memory use optimisations. The development of FuSiGroups was carried out using an evolutionary prototyping approach with constant refinements in both design and implementation before the version of the FuSiGroups algorithm described in Section 3.1.6 was obtained.

3.3.1 FuSiGroups

The FuSiGroups algorithm was implemented using the Java programming language and a MySQL database. Java was chosen as the programming language because it is platform independent and versatile, and interfaces well with MySQL. MySQL was chosen as it is non-proprietary and also because it is the only database type in which the full GO database is available. A smaller version (latest-lite) of the GO database, excluding UniProtKB electronic annotations, is available in OBO XML and RDF XML formats. The most complete version of the database was chosen in order to allow completely free choice of annotation and species parameters for the programme.

Database considerations

The complete Gene Ontology database, including annotations (but excluding sequence information), was downloaded from the GO FTP site in April 2009 (go_200904-assocdb-tables.tar.gz). The use of a local copy of the Gene Ontology is preferable and, in fact, necessary for two reasons. Firstly, the GO is updated on a weekly basis and even small changes in annotations or the ontological structure can have an effect on semantic and functional similarity results. In order to keep results consistent across tests and experiments run at different times, it is therefore desirable to use a local copy of the database which is stable. Furthermore, connection to a database mirror such as that provided by EBI is limited to one connection at a time as this is a shared resource. Although the database queries in the FuSiGroups implementation are designed to retrieve information efficiently, they still require more resources than EBI's acceptable usage policy will allow, resulting in disconnection from the database and temporary blocking of the client machine's IP address.

Similarity approach considerations

As most of the approaches listed in Section 3.1 only differ in the way they use information content, the majority of the implementation was done in a generic format,

with only the final calculation step for each approach requiring some individual code. For the IC-based approaches, frequency and probability of occurrence values for each GO term were batch-retrieved and stored in main memory to minimise database access as frequent database access can cause bottle-neck issues. For large datasets, the trade-off with this approach is a large memory requirement but most modern desktop machines have sufficient RAM to handle this kind of computational task.

The Wang et al. [2007] approach required some minor variations in the implementation in order to accommodate its differences from IC-based approaches. In particular, this approach gives each ontological edge a weight, with the weights depending on the nature of the edge. The authors only provide weighting factors for "is_a" and "part_of", as these were the only two types of edges available in the early version of the GO. Between GO releases 2008-01[10] and 2008-04[11], the GO included three further types of edges, "negatively_regulates", "positively_regulates", "regulates". While the absence of weighting factors for these three edge types from the original version of the Wang approach is clearly due to the fact that these types did not exist yet, the authors never updated their approach to allow for these new edges. The latest version of the G-SESAME tool[12] [Du et al., 2009], which was developed by the same team as the Wang approach, still only requires weighting factors for the two original edge types. In the absence of any guidance on how to weight the new edge types, it was decided to give these three additional edges the same weight as the "part_of" edge, rather than exclude them from the calculation as this might lead to unconnected branches in the tree. Overall, it is expected that the impact of the three new edge types should be low as together, they only represent 7.5% of all relationships in the GO release used here (2009-04), compared to 8.6% of "part_of" relationships and 83.9% of "is_a" relationships. The weights recommended by Wang et al., namely 0.8 for "is_a" edges and 0.6 for other edge types, were used.

The correctness of the implementation of Wang's approach could be verified by comparing results obtained from the FuSiGroups implementation to results obtained from the G-SESAME tool. Results obtained for a test-run in July 2008, using GO release 2007-10, were found to be identical, bar rounding differences at 3 significant figures. This same method of direct implementation verification was not possible for any of the other approaches, as none of other semantic similarity tools available

[10]Database ensembl_go_49 on the Ensembl's [Flicek et al., 2011] public MySQL server, ensembldb.ensembl.org

[11]Database ensembl_go_50 on the Ensembl's [Flicek et al., 2011] public MySQL server, ensembldb.ensembl.org

[12]http://bioinformatics.clemson.edu/G-SESAME/, using GO release 2011-02, accessed on 16/03/2011

online state which GO release the computations are based on or provide details on what corpus was used. Some tools also use only a specific type of identifier such as UniProt identifiers, which are unavailable for yeast in GO, or only cover certain species, which makes using them for testing purposes impractical.

The GraSM algorithm was implemented according to the pseudocode suggested by the authors in Couto et al. [2007]. Again, direct comparison of results obtained from FuSiGroups to results from the authors' own tools (FuSSiMeG [Couto et al., 2003] and ProteInOn [Faria et al., 2007]) was not possible as essential information such as GO release is not given in the paper.

Programme input

There are two sets of input required to run FuSiGroups. The first is the list of gene products to be included in the run, in a plain text file with one gene product per line. Identifiers need to be of the type listed in GO in the "xref_key" field of the "dbxref" table, e.g. "S000001234" for yeast or "FBgn0000490" for fruit fly. The second set of input is a list of parameters, namely genus and species of the gene products to be tested; the semantic similarity approach (Lin, Resnik, Schlicker or Wang); whether to use GraSM or not; the functional similarity approach (BMA or MAX); whether to use all or only non-electronic annotation; and the semantic and functional thresholds.

It is also possible to specify whether to calculate semantic and functional similarities from scratch or load pre-calculated similarities from a database. This is the recommended approach if a number of different thresholds are to be tested as calculation of similarities for a dataset the size of the Eisen dataset (approx. 2500 gene products) takes a couple of hours whereas reading in existing similarities from a database, then performing the groupings generally takes about quarter of an hour for a dataset of this size.

Multi-species comparisons are not supported at present. The proof-of-concept implementation does not include a user interface, so parameters have to be entered directly into the code of the programme's "main" method.

Hardware requirements

The FuSiGroups programme was run on a desktop computer with a Linux operating system (Ubuntu 10.04 LTS - Lucid Lynx). Due to the amount of data held in memory at any time, the programme requires an above-average amount of RAM. 2.5GB were allocated to the programme although of course different dataset sizes

require different amounts of memory. A larger dataset may even require more RAM.

3.3.2 Dataset

Data sources

Two versions of the Eisen dataset were downloaded, from two different locations. The first version was an online supplement to Eisen et al. [1998]'s paper found on the Stanford Genomic Resources website[13]. Available were the tab-delimited and Excel versions of the data used to generate figure 2 of the Eisen paper, including the systematic and full names for each gene and the expression ratios of all 2466 gene products for 79 experimental conditions, as well as the enhanced version of the published image and a key to the columns of the figures (experimental conditions). The second version was downloaded from the "KEIA" (Knowledge Extraction, Integration and Applications) research group website of the Université de Nice Sophia-Antipolis[14]. Available files included a tab-delimited version of the expression ratios of 2465 gene products for 79 experimental conditions, using primary SGD IDs as identifiers, the 79 experimental conditions, the under- and over-expressed cutoff thresholds for each condition (computed using the group's NorDi algorithm [Martinez et al., 2007]), the discretised expression measures for all genes and finally a list of the 2465 genes with 737 columns of gene annotations, pathway information, transcriptional regulators, phenotypes, PubMed IDs and the discretised expression measures.

The reason for downloading two versions of the dataset was that the dataset downloaded from the Stanford website (published in conjunction with the original paper) uses the ORF's systematic name (or "feature name") as identifier for each gene product whereas the second version of the dataset uses SGD identifiers. The annotation data in the downloaded version of the GO database only contains SGD IDs, so no other identifiers can be used in the program without including an initial ID matching step. Mapping between different types of identifiers can be difficult as there is not always a one-to-one mapping, especially if one identifier refers to a gene and another to a protein (several proteins can derive from one gene). Checking the "KEIA" dataset, which also provides more meta data, against Eisen et al.'s version (considered more reliable) was an additional step to ensure the consistency of the dataset.

[13] http://genome-www.stanford.edu/clustering/
[14] http://keia.i3s.unice.fr/?Datasets:Eisen_et_al._dataset

Reconciling inconsistencies

Initial comparison between the two datasets showed that the Eisen version had one more entry than the KEIA version. The mapping between SGD IDs and systematic identifiers was downloaded from the SGD database using their batch download tools[15]. Then the two datasets were compared manually by visually matching, for each row in the dataset, the identifier and first two expression values of each version against the SGD mapping.

In 14 cases, SGDIDs from the KEIA dataset could not be mapped to a systematic name from the Eisen dataset and vice versa. In these situations, the mapping was done using the expression values. Table 3.4 summarises the discrepancies between the two datasets and how they were resolved.

In the majority of cases, the SGDID in the KEIA version was associated with an updated systematic name, e.g. due to a change in an ORF following a new release of the yeast genome. If the description associated with a systematic name in the Eisen version matched the description associated with a different SGDID in the yeast genome database, the SGDID from the KEIA version was used.

The additional data entry in the Eisen dataset was found to be due to the ORF YER108C having been merged with YER109C at some point between the publication of the Eisen dataset and the generation of the KEIA version. Rather than simply removing the deprecated ORF, the KEIA group averaged the expression values for the two ORFs and associated the resulting values with the SGDID for YER109C. While the label was retained for this study, the expression values were replaced with the original values for YER109C.

Annotations

All annotation information used in the FuSiGroups software was taken from the GO database rather than from SGD in order to achieve the maximum amount of consistency (i.e. in order to avoid using annotation information from different releases). If a gene product did not have any annotation in one (or two) ontological aspects, the root term for that ontological aspect was assigned as annotation to the gene product in question in order to allow computation of semantic similarity for that aspect. This is an appropriate course of action as any gene product has some kind of molecular function, is active in some cellular compartment and is part of some biological process, even if the details of any of these have not yet been characterised.

Although all ORFs in the Eisen dataset were originally chosen by the authors

[15] http://www.yeastgenome.org/cgi-bin/batchDownload

	Eisen version	KEIA version	Explanation and Resolution
1	YHR047C (systematic name for S000001089, gene name: AAP1)	S000007267 (alias of gene name: AAP1)	Confusion of identifier in KEIA version due to gene name alias. Replacement not justified, use S000001089 in KEIA version.
2	YOR235W (dubious ORF unlikely to encode a protein)	S000007294 (small nucleolar RNA)	The two elements are located very closely together on chromosome XV, with an overlap of about 80 bases. The replacement is justified due to the dubious nature of YOR235W.
3	YPL144W (systematic name for S000006065)	S000007441 (small nucleolar RNA)	The two elements are not related in any way but are located adjacently on chromosome XVI. The replacement is not justified as YPL144W is a verified ORF for a proteasome chaperone;s use S000006065 in KEIA version.
4a	YJL102W (systematic name for S000003638)	S000001683 (SGDID for YKL200C)	The two elements in 4a and 4b both have nothing in common. The case is treated as a mix-up of two SGDIDs. Exchange SGDIDs in KEIA dataset.
4b	YKL200C (systematic name for S000001683)	S000003638 (SGDID for YJL102W)	
5	YCL007C (dubious ORF unlikely to encode a protein)	S000028508 (SGDID for YCL005W-A)	YCL007C overlaps verified ORF YCL005W-A. Both ORFs have alias CWH36. Replacement justified.
6	YAR044W (alias for YAR042W)	S000000081 (SGDID for YAR042W)	Replacement justified.
7	YBR090C-A (alias for YBR089C-A)	S000002157 (SGDID for YBR089C-A)	Replacement justified.
8	YJL206C (systematic name for S000003741)	S000003742 (SGDID for YJL205C)	According to SGD, gene name for YJL206C is uncharacterised. According to Eisen version, gene name for YJL206C is NCE101, which in SGD is listed as the gene name for S000003742. Replacement justified.
9	YHR039C	S000002100 (SGDID for YHR039C-A)	According to SGD, gene name for YHR039C is MSC7. According to Eisen version, gene name for YHR039C is VMA10, which in SGD is listed as the gene name for S000002100. Replacement justified.
10	YER060W	S000002958 (SGDID for YER060W-A)	According to SGD, gene name for YER0060W is FCY21. According to Eisen version, gene name for YJL206C is FCY22, which in SGD is listed as the gene name for S000002958. Replacement justified.
11	YHR001W	S000003529 (SGDID for YHR001W-A)	According to SGD, gene name for YHR001W is OSH7. According to Eisen version, gene name for YJL206C is QCR10, which in SGD is listed as the gene name for S000003529. Replacement justified.
12	YHR005C	S000003530 (SGDID for YHR005C-A)	According to SGD, gene name for YHR005C is GPA1. According to Eisen version, gene name for YJL206C is MRS11, which in SGD is listed as an alias for S000003530. Replacement justified.
13	YCR028C	S000007222 (SGDID for YCR028C-A)	According to SGD, gene name for YCR028C is FEN2. According to Eisen version, gene name for YJL206C is RIM1, which in SGD is listed as the gene name for S000007222. Replacement justified.
14	YER108C & YER109C (Two merged ORFs)	S000000911 (SGDID for YER109C)	Expression values for S000000911 correspond to the average of the expression values for YER108C & YER109C. Expression values in KEIA dataset replaced with expression values for YER109C.

Table 3.4: Dataset inconsistencies. Column 1 shows the systematic name from the Eisen version, column 2 shows the SGDID listed for the corresponding expression values in the KEIA version. Column 3 briefly describes how the conflict was resolved. "Replacement justified" means that the choice of SGDID made my the KEIA group is considered to be appropriate.

because annotation was available for each of them at the time of the original study, one ORF, "S000001683" was found to not have any annotation whatsoever in the GO release used in the present analysis. This ORF was therefore removed from the dataset, as it was not considered useful or informative.

In order to compute semantic and functional similarities, each direct annotation of a GO term to a gene product recorded in the GO is used once and only once, regardless of how many times that annotation is characterised in the database, with different evidence codes or references. This convention is used for the gene products in the dataset and for the set of all annotated gene products of a species used as the corpus for the calculation of information content. If only non-electronic annotations are considered, the same convention still applies for all non-electronic annotation. If a gene product was annotated with the same GO term three times, with one association characterised with the evidence code "IEA", the other two with other, distinct evidence codes, the gene product would have this annotation. A gene product with two "IEA"-characterised annotations of the same GO term on the other hand would not have that annotation considered in an experiment excluding electronic annotation.

Less than one percent of gene-GO term associations in the GO are qualified with "is_not". This means that the gene product in question should specifically not be associated with that particular GO term. This information is included in the GO in two forms, namely through the binary field "is_not" in the "association" table, in which a value of 1 signifies dissociation between the gene product and GO term, and in the "association_qualifier" table. In order to avoid these exclusion relationships being counted towards association counts, any database query specifically selects only associations where "is_not" is set to 0.

3.3.3 Experiments

Table 3.5 lists all the combinations of variables for which functional similarity was computed. After selecting the best combinations of variables (see Chapter 4) and optimum threshold ranges (see Chapter 5), the grouping algorithm was used for each of the selected measures at the determined thresholds.

3.4 Summary

In this chapter, design considerations for the study in general and the FuSiGroups software in particular, as well as a number of implementation considerations were

Sem. sim.	*Func. sim.*	*Annotation*	*Ancestors*
Lin	BMA	all	LCA
			GraSM
		nonIEA	LCA
			GraSM
	MAX	all	LCA
			GraSM
		nonIEA	LCA
			GraSM
Resnik	BMA	all	LCA
			GraSM
		nonIEA	LCA
			GraSM
	MAX	all	LCA
			GraSM
		nonIEA	LCA
			GraSM
Schlicker	BMA	all	LCA
			GraSM
		nonIEA	LCA
			GraSM
	MAX	all	LCA
			GraSM
		nonIEA	LCA
			GraSM
Wang	BMA	all	NA
		nonIEA	NA
	MAX	all	NA
		nonIEA	NA

Table 3.5: All combinations of approaches and other factors for which functional similarity values were computed.

discussed. This included justifications for why some measures and parameters were included and others excluded from the study, and a detailed description of the algorithm. The implementation considerations specifically addressed several aspects that are often omitted from publications on functional similarity, making the replication of results difficult.

The chapter also included an outline of the evaluation strategy for the different aspects of the project, including how to compare the different approaches and parameters, how to experimentally derive the thresholds for the algorithm and how to validate the grouping results. In the next chapter, the different semantic and functional similarity approaches and other associated variables are going to be compared in order to select the combinations of variables with the best overall performance, which will then be used for the rest of this work.

Chapter 4

Semantic and functional similarity approaches

As discussed in previous chapters, a key problem in semantic and functional similarity research is that new approaches are usually only evaluated against the "original three", Resnik [1995]; Lin [1998]; Jiang and Conrath [1997], which they are always found to outperform. As part of this study, an experimental comparison of several semantic similarity approaches developed for the GO context [Schlicker et al., 2006; Wang et al., 2007; Couto et al., 2005] was undertaken. Different approaches to calculate the functional similarity between gene products annotated with multiple GO terms were also compared.

The evaluation was performed for three different biological properties - gene expression, protein interaction and phenotype. The datasets were generated as described in Section 3.2.1. An aggregate dataset was created by combining the three individual datasets into one dataset of 30 times 1000 observations (500 true positives and 500 true negatives). ROC curves and the associated AUC index were used to compare the performance of the different measures for the aggregate dataset as well as for each dataset separately. All datasets were analysed using R's ROCR library [Sing et al., 2005].

In this chapter, the performances of the different measures are compared. Results are primarily presented for the aggregate dataset and differences in behaviour compared to the individual datasets are discussed. At the end of the comparisons, the approaches and parameters that perform the best overall will be selected for use with the FuSiGroups algorithm.

4.1 ROC curves

Although it is often difficult to tell which ROC curve shows the best performance if several curves are very close, it is still important to get a general overview of the curves before looking at an index such as the AUC to evaluate overall performance. The following ROC curves were all obtained using ROCR's *plot()* function on performance objects obtained using the same library's *performance("tpr", "fpr")* function.

As the complete dataset is an aggregate of three datasets, each of which in turn consist of 10 subsets, each individual subset generates one separate ROC curve. In order to study the overall trend for a given combination of semantic and functional similarity approaches and ancestor and annotation choices, the ROC curves of the individual subsets are averaged using threshold averaging. An example of ROC curves before and after threshold averaging is shown in Figure 4.1. No thresholds are shown on the unaveraged curves to avoid cluttering. The averaged curve has thresholds shown at intervals of 0.1, as well as error bars showing one standard deviation in each direction. Three "bands" can clearly be distinguished on the graph with the unaveraged curves. The top band corresponds to the gene expression dataset, the middle band to the protein interaction dataset and the bottom band to the phenotype data. The differing levels of ROC curves give a general indication of the level of relatedness of the similarity aspect of the respective datasets and functional similarity, e.g. gene product pairs with high expression similarity are more likely to have equally high functional similarity than pairs with high phenotype similarity. In fact, these ROC curves imply an almost perfect match between expression and functional similarity, an excellent match between protein interaction and functional similarity and a fair correspondence between phenotype and functional similarity. Despite these differences, even the phenotype dataset is still sufficiently far above the diagonal line to be acceptable.

For space reasons, only a sample set of ROC curves that can be derived from the full dataset are shown here. Figure 4.2 shows the ROC curve for "BMA-all annotation-MICA-rFunSim" for all four semantic similarity approaches and for the aggregate as well as individual datasets. For the aggregate dataset, the ROC curves for Lin and Schlicker have the most evenly distributed thresholds of the four curves, whereas most of the thresholds for Resnik are clustered in the bottom left corner and a lot of the thresholds for Wang are clustered in the top right corner. For Resnik's approach, this suggests the numbers of both true and false positives are low at high thresholds. In fact, due to the normalisation step necessary to bring similarities in

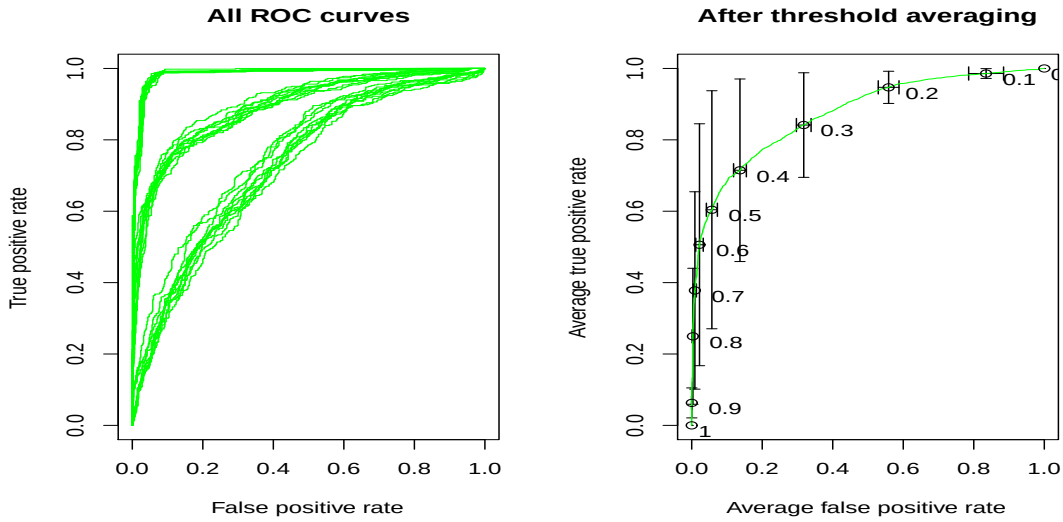

Figure 4.1: ROC curves for "Schlicker-BMA-all-MICA-rFunsim" before and after threshold averaging. No thresholds are shown for the unaveraged curves to avoid cluttering. Thresholds on the averaged curve at intervals of 0.1 and are given with error bars of one standard deviation in each direction. The three bands of unaveraged curves correspond to, from top to bottom, the gene expression dataset, the protein interaction dataset and the phenotype dataset.

Resnik's approach between 0 and 1, the majority of similarities are below 0.5, which explains the distribution of the thresholds. For Wang's approach, the threshold distributions mean that even the majority of true negative gene product pairs have a fairly high similarity.

In terms of performance, the order of approaches from best to worst appears to be Resnik, Schlicker, Lin, Wang, except for the gene expression dataset, where Resnik's approach can be observed to the right of the other approaches, i.e. performing slightly worse. It is however very difficult to conclusively determine which approaches perform better than others. These four graphs are a good example of the difficulty of judging the relative performances of different qualifiers solely on the basis of their ROC curves. In order to determine in each case which of the scores has the overall best performances, it is necessary to compare them using a single-figure index such as the AUC (area under the curve) measure.

The rest of this section includes only ROC curves for the full dataset. The corresponding curves for the original datasets can be found in Appendix A.

Figure 4.3 shows the individual ontology scores and aggregate rFunSim score for each semantic similarity approach. While MF clearly performs worst for each approach, there are always at least two ROC curves which are very close together and even cross at one or more points such as rFunSim and BP for Lin and rFunSim and CC for Resnik. This again illustrates the need for an objective measure to really

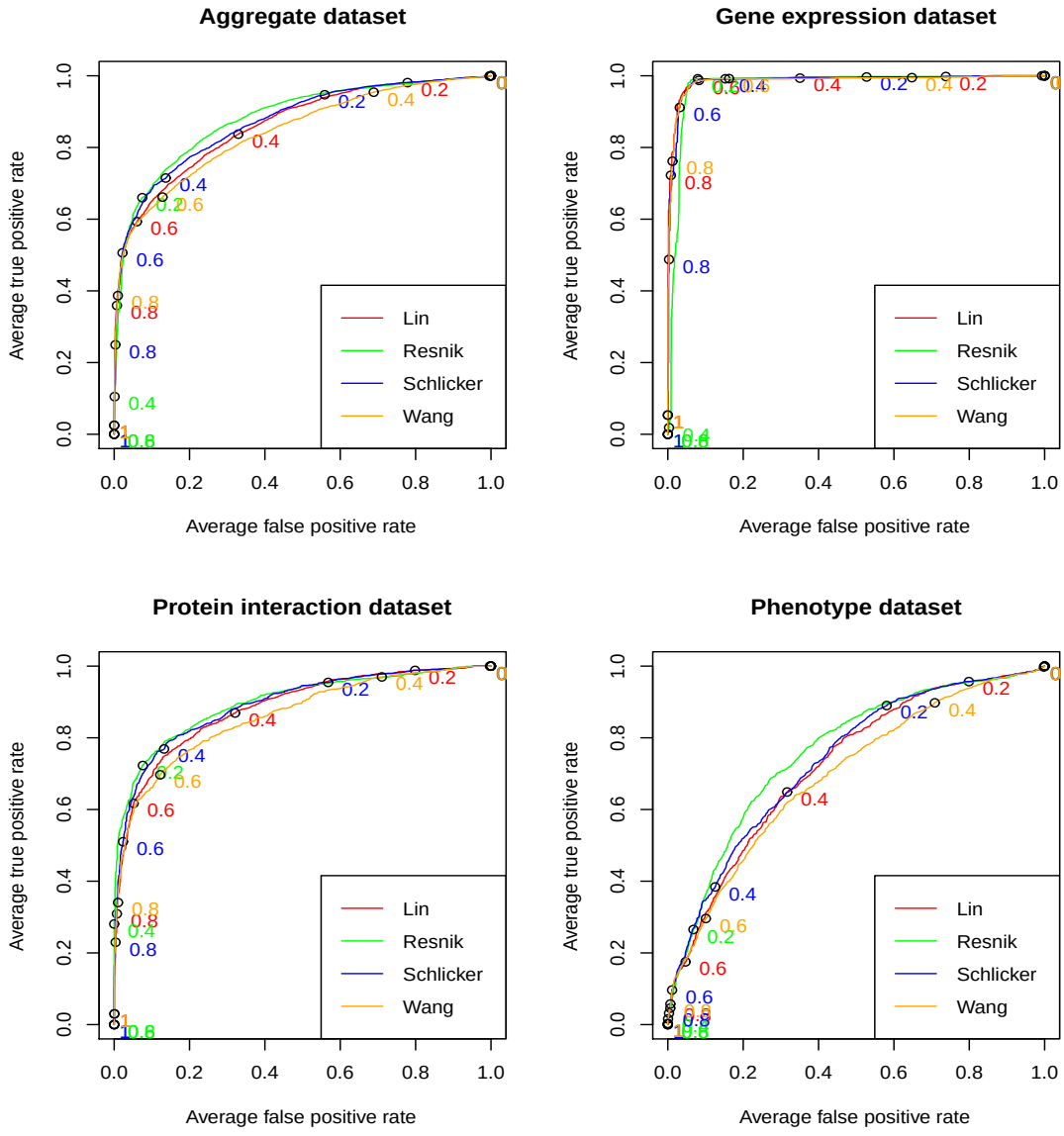

Figure 4.2: ROC curves for all approaches for "BMA-all-MICA-rFunSim" and for the aggregate dataset and the three individual datasets.

evaluate performances. This trend is relatively consistent across the three individual datasets as well.

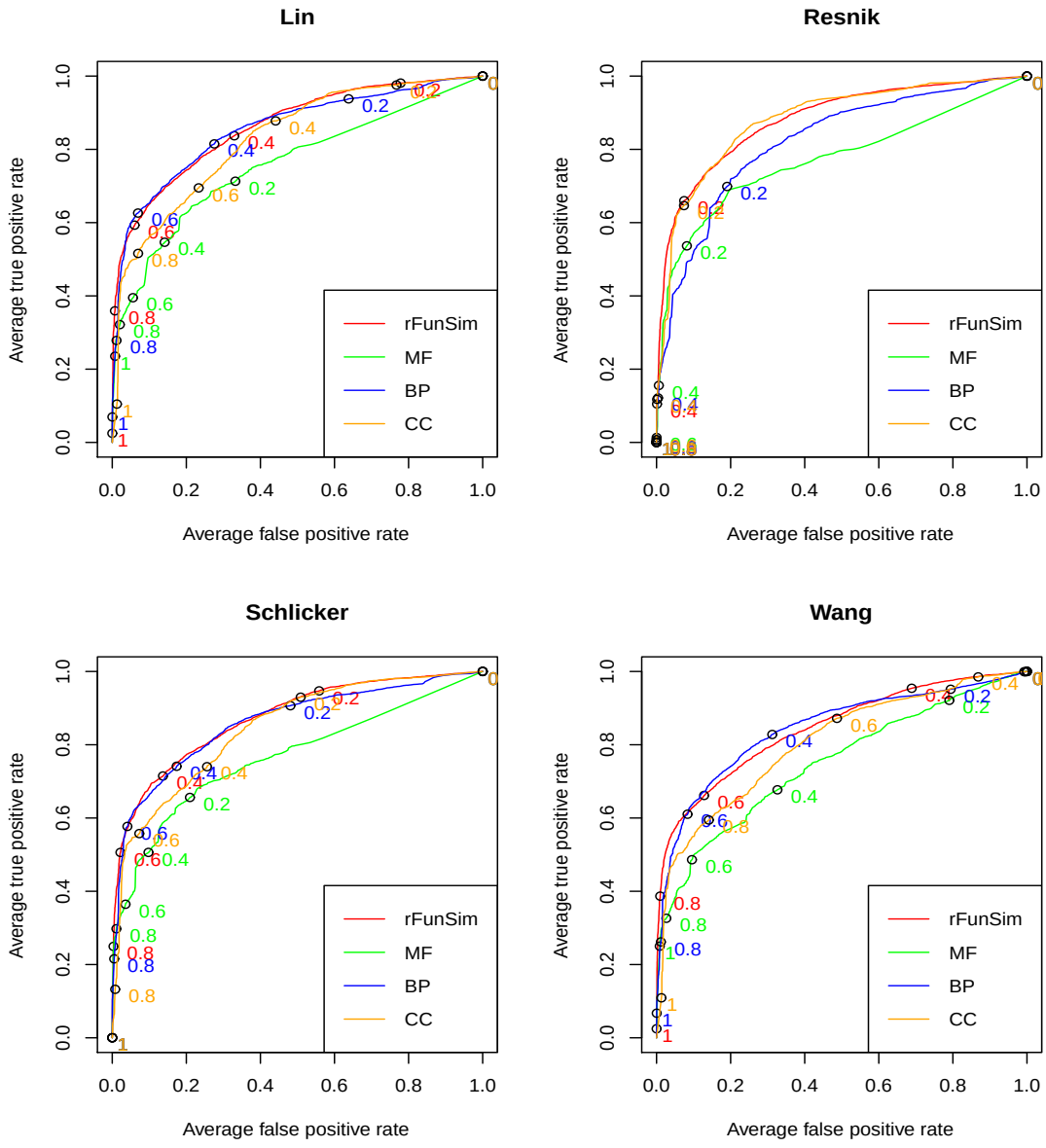

Figure 4.3: ROC curves for individual ontology and rFunSim scores for all approaches with "BMA-MICA'

Figure 4.4 shows the ROC curves for BMA and MAX for each approach for "all annotation-MICA-rFunSim". While the overall shapes of the two curves are always relatively similar, it is noteworthy that for Lin and Wang, the first threshold (1.0) on the ROC curve for MAX does not appear until an average true positive rate between 0.4 and 0.6. This means that technically, the ROC curve does not start until that point, although the graph drawing tool used (ROCR's *plot()* function) places the

point of origin of each curve at the point (0,0) and then draws a straight line to the first actual threshold. In the present case, the first threshold is still close to the origin of the X-axis (average false positive rate). However, in other cases, where the first threshold point is displaced along both axes, it is necessary to analyse the ROC curve and AUC together in order to take this trend into account.

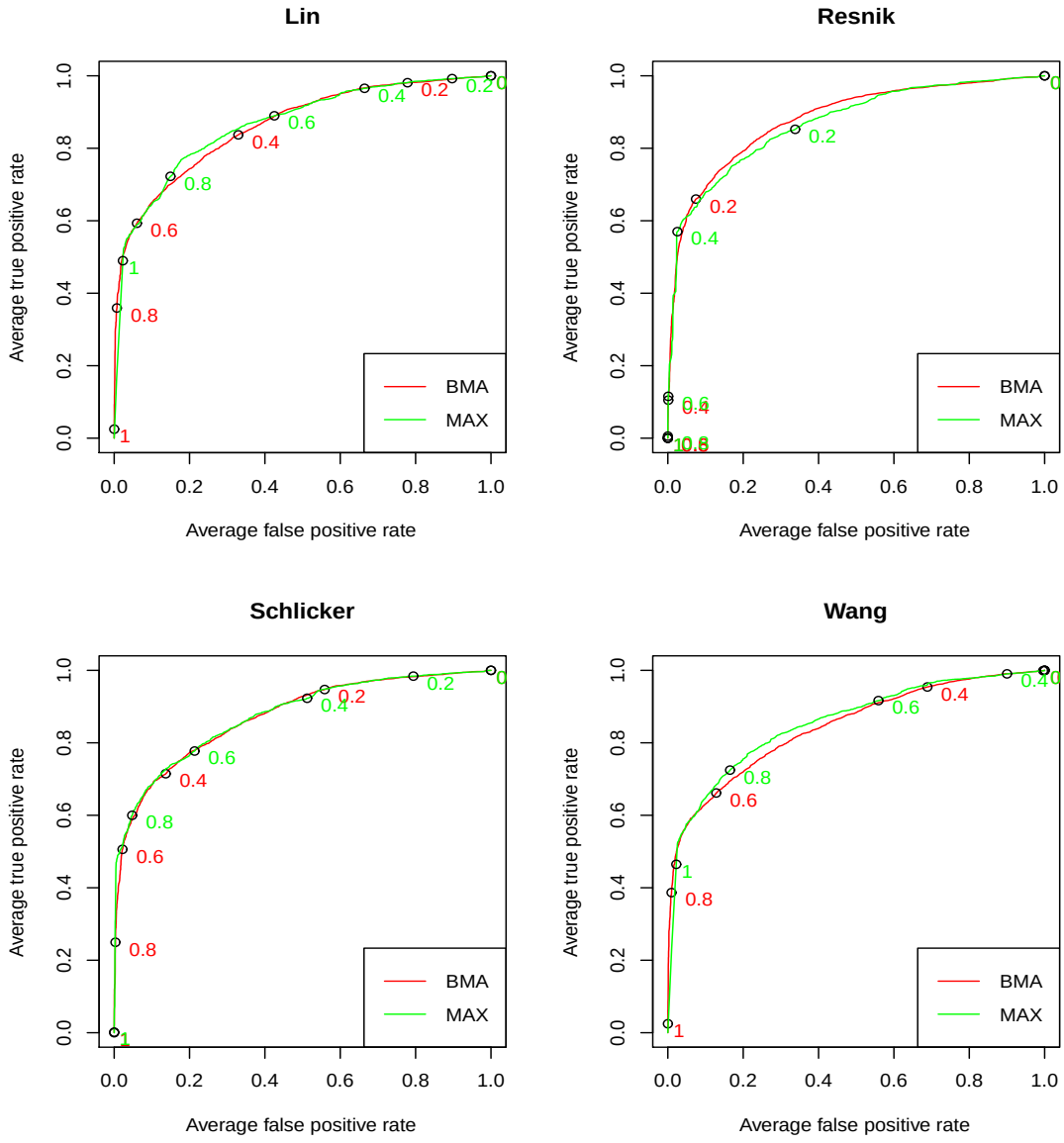

Figure 4.4: ROC curves for all approaches for BMA and MAX with "all-MICA-rFunSim"

In Figure 4.5, the use of the MICA and the GraSM algorithm as ancestor selections are compared for "all annotation-BMA-rFunSim" and for all IC-based semantic similarity approaches. The approach by Wang et al. is included for reference purposes only as the ancestor choice does not apply to it. For the IC-based approaches,

it is immediately clear that the thresholds for GraSM are not as evenly distributed along the curve as those for MICA. This is to be expected as using multiple disjunctive ancestors to compute the semantic similarity between two terms leads to a lower similarity, compared to using only the most informative common ancestor. All pairs of ROC curves are relatively close together, with only the two cases for Resnik's approach showing enough difference to support the conclusion that MICA overall performs better than GraSM. The curves for Lin and Schlicker are too close and require analysis of the respective AUC indexes to draw any conclusions. The same trend is observed in the three datasets if they are considered individually.

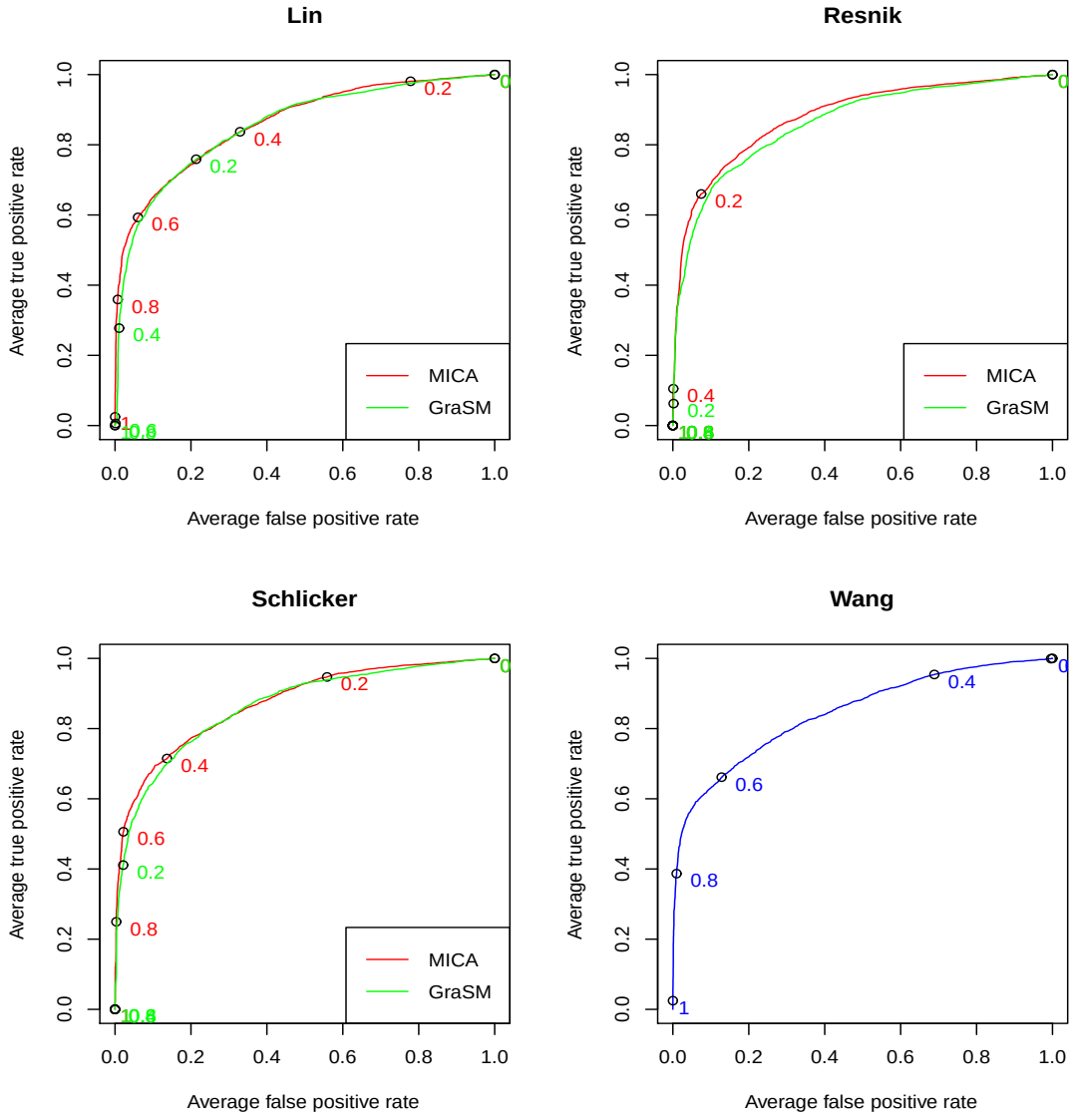

Figure 4.5: ROC curves for all IC-based approaches for MICA and GraSM with "all-BMA-rFunSim". Wang's approach is included for comparison only.

Finally, Figure 4.6 show the ROC curves for the full and non-electronic annotation dataset for all four approaches, using "BMA-MICA-rFunSim". Although AUCs need to be analysed to confirm this, it appears that with the exception of Wang's approach, the full annotation dataset usually performs better than the non-electronic dataset. This trend is particularly pronounced in Resnik's approach. The same observation can be made for the protein interaction and phenotype datasets. In the gene expression dataset on the other hand, the non-electronic annotation data appears to perform marginally better than the full dataset, although the curves are overall too close to draw a definitive conclusion.

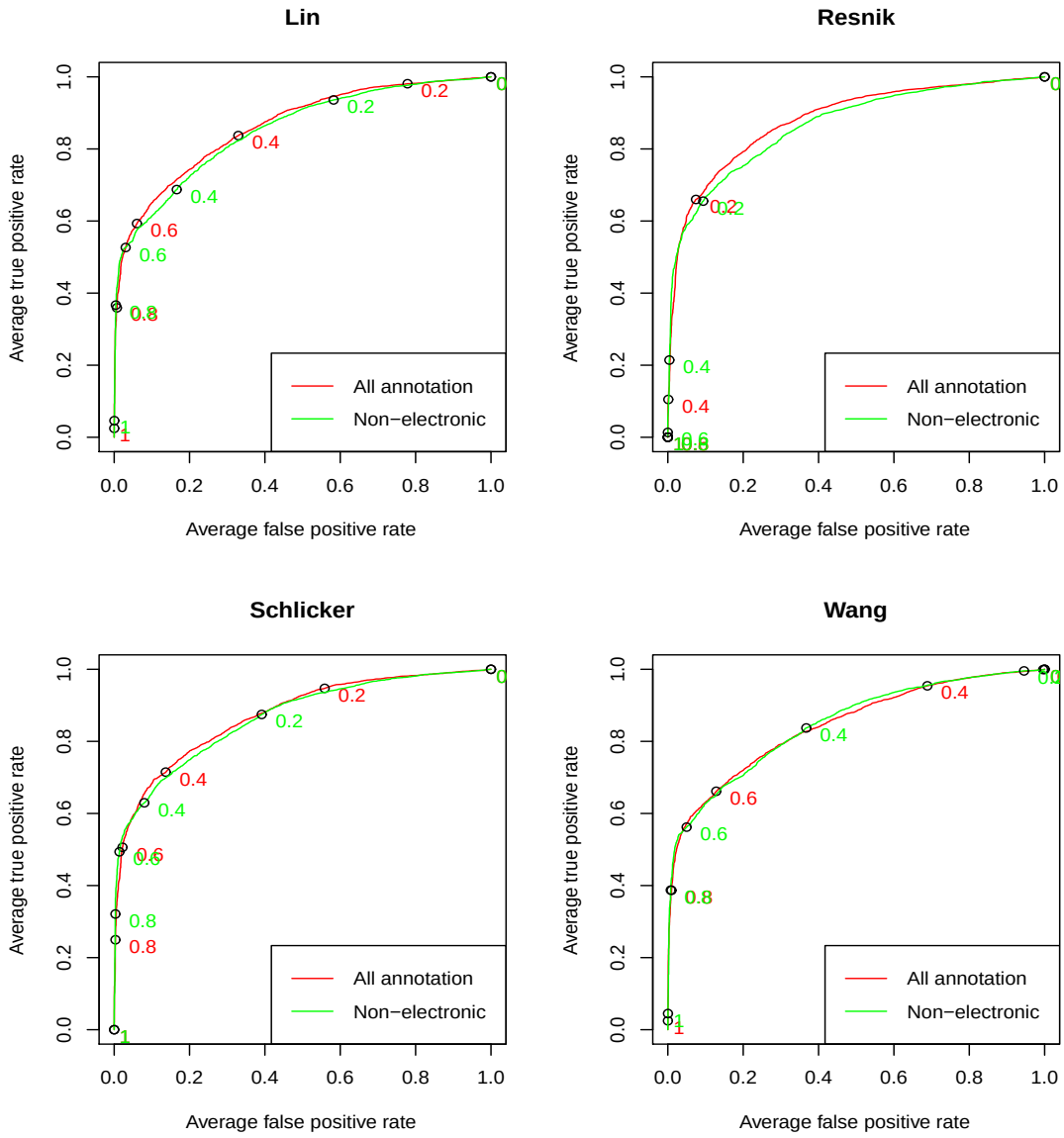

Figure 4.6: ROC curves for all approaches for "BMA-MICA-rFunSim" with full and non-electronic annotation

After this quick overview of the general trends of the ROC curves for this dataset, the next section covers a more in-depth analysis of the AUC scores.

4.2 AUC results

Using ROCR's *performance(auc)* function, AUCs for each subset were computed. The AUCs for each score were obtained by averaging the AUCs of the 10 individual subsets for that score. Overall AUCs are shown in Table 4.1 for the full dataset and Appendix A for the individual datasets (Tables A.1, A.16 and A.31).

Variables				AUCs			
Sem. sim.	*Func. sim.*	*Dataset*	*Ancestors*	*MF*	*BP*	*CC*	*rFunSim*
Lin	BMA	all	MICA	0.762	0.858	0.826	0.865
Lin	BMA	all	GraSM	0.765	0.842	0.816	0.858
Lin	BMA	nonIEA	MICA	0.751	0.800	0.857	0.850
Lin	BMA	nonIEA	GraSM	0.750	0.850	0.811	0.853
Lin	MAX	all	MICA	0.761	0.819	0.753	0.863
Lin	MAX	all	GraSM	0.775	0.783	0.794	0.839
Lin	MAX	nonIEA	MICA	0.759	0.838	0.786	0.857
Lin	MAX	nonIEA	GraSM	0.756	0.821	0.811	0.850
Resnik	BMA	all	MICA	0.778	0.820	0.871	0.878
Resnik	BMA	all	GraSM	0.767	0.806	0.860	0.862
Resnik	BMA	nonIEA	MICA	0.736	0.868	0.811	0.864
Resnik	BMA	nonIEA	GraSM	0.738	0.799	0.873	0.855
Resnik	MAX	all	MICA	0.778	0.754	0.873	0.864
Resnik	MAX	all	GraSM	0.779	0.734	0.872	0.854
Resnik	MAX	nonIEA	MICA	0.744	0.757	0.871	0.858
Resnik	MAX	nonIEA	GraSM	0.745	0.744	0.881	0.849
Schlicker	BMA	all	MICA	0.768	0.860	0.839	0.872
Schlicker	BMA	all	GraSM	0.772	0.842	0.833	0.864
Schlicker	BMA	nonIEA	MICA	0.748	0.849	0.841	0.862
Schlicker	BMA	nonIEA	GraSM	0.748	0.849	0.848	0.865
Schlicker	MAX	all	MICA	0.784	0.822	0.851	0.873
Schlicker	MAX	all	GraSM	0.771	0.782	0.841	0.838
Schlicker	MAX	nonIEA	MICA	0.753	0.833	0.840	0.864
Schlicker	MAX	nonIEA	GraSM	0.752	0.820	0.858	0.857
Wang	BMA	all	NA	0.750	0.846	0.802	0.848
Wang	BMA	nonIEA	NA	0.787	0.771	0.856	0.848
Wang	MAX	all	NA	0.756	0.815	0.743	0.853
Wang	MAX	nonIEA	NA	0.798	0.842	0.766	0.856

Table 4.1: AUCs for all experiments in the aggregate dataset

The AUCs confirm the trend already seen in Figure 4.2, with the gene expression

dataset showing the best performance overall, the phenotype dataset the worst performance, with the protein interaction dataset in between and the aggregate dataset showing a very similar trend. The average AUC for rFunSim is 0.858 for the aggregate dataset, 0.980 for expression, 0.877 for protein interaction and 0.716 for the phenotype dataset.

This suggests that out of the three aspects selected for the comparison of the different semantic and functional similarity approaches, phenotype-based similarity between gene products is least comparable to annotation similarity. There are a number of possible explanations for this. It is possible that the phenotype annotation is simply of a lower quality, thus making for a poorer dataset. A more plausible explanation is however that similar phenotypes can be obtained in so many different ways that simple phenotype similarity does not automatically imply functional similarity. This is particularly true for very common phenotypes, which may appear to associate genes that do not in fact have any common functional aspects.

Very high gene expression similarity, especially in a dataset with as many samples as the Eisen dataset, is generally a good indicator of a functional relationship between two genes as it means their expression is affected in the same way by the same external stimuli. This is reflected in the very high AUC values obtained for all measures for the expression dataset alone. The intermediate position of protein interaction similarity between gene expression and phenotype similarity can equally be explained through the nature of protein interaction. Although an observed interaction between two proteins implies some shared functional aspect and co-localisation in the cell during the interaction, many proteins have more than one function and may have different activities in different parts of the cell. Therefore a documented interaction is not automatically a guarantee for full functional similarity across all of the proteins' functional aspects, which is the most likely explanation for the good but not perfect AUCs obtained for the protein interaction dataset.

A quick overview of the AUCs in Table 4.1 makes it obvious that in most cases, rFunSim has higher AUCs than any of the other scores, suggesting that the aggregate score performs better than the individual ontologies. Exceptions to this are listed in Table 4.2.

The MF ontology score never performs better than rFunSim. The BP score only outperforms rFunSim for "Resnik-BMA-nonIEA-MICA". The CC score is the only score that performs better than rFunSim in a number of cases, although the cases suggest no obvious pattern, such as CC always outperforms rFunSim for dataset X or ancestor selection Y. There is also no single combination of variables where rFunSim is outperformed in all cases.

Ontology	*Sem. sim.*	*Func. sim.*	*Dataset*	*Ancestors*
BP	Resnik	BMA	nonIEA	MICA
CC	Lin	BMA	nonIEA	MICA
	Resnik	BMA	nonIEA	GraSM
	Resnik	MAX	all	MICA
	Resnik	MAX	all	GraSM
	Resnik	MAX	nonIEA	MICA
	Resnik	MAX	nonIEA	GraSM
	Schlicker	MAX	all	GraSM
	Schlicker	MAX	nonIEA	GraSM
	Wang	BMA	nonIEA	NA

Table 4.2: Cases in which individual scores outperform aggregate scores

While the overall better performance of rFunSim compared to the single ontological scores can be deduced by looking at the AUC values in Table 4.1, it is not immediately obvious whether the difference between the four scores is statistically significant. For this reason, a single-factor analysis of variance (ANOVA) of the four sets of scores was performed using the Microsoft Excel data analysis tools. This analysis confirmed that the difference between the scores is statistically significant, with a p-value of $1.6E^{-22}$.

As rFunSim performs better than any of the single ontological scores in over 88% of the cases (74 out of 84 comparisons), the analysis hereafter will be based on the rFunSim scores. In order to determine which semantic and functional similarity approaches and dataset and ancestor selections performed best, a two-step analysis was performed. First, the AUC values for different combinations of variables were submitted to an ANOVA test in order to determine whether the differences between the AUC scores are significant from a statistical point of view. Secondly, the performances for a given set of combinations are ranked (with 1 for the highest AUC, 2 for the second highest etc). Then the ranks are added across columns and the results sorted from lowest to highest. The measure or combination of variables with the lowest summed ranks is considered to have the best overall performance. The sum of ranks was chosen rather than average of ranks to clearly differentiate between individual ranks and the global rank of each measure.

Cases where ranks differ strongly from the general trend (e.g. an approach that usually ranks highest performs worst) are marked in red in the results tables in the rest of this chapter and discussed. A result is considered to "differ strongly" from the general trend if its rank is greater or smaller by one standard deviation rounded to zero decimal places than at least one other rank in the same row. In situations

where this applies to two different ranks, the more contradictory one is discussed, e.g. if the overall worst performing approach out of four is ranked 2nd, 3rd and 4th for three different measures (standard deviation = 1), the ranks of 2 and 4 differ by more than one standard deviation from each other but the rank of 4 is more in agreement with the overall ranking of the approach, so the rank of 2 would be discussed.

4.2.1 Statistical analysis

All statistical analysis was performed using the Microsoft Excel data analysis tool kit. Although some of the analysis was performed for pairs of variables (e.g. BMA vs MAX), ANOVA was used in all cases rather than using t-tests in some cases and ANOVA in others. For all analyses, a significance level $\alpha = 0.05$ was used. The first set of analyses consisted of a number of single-factor ANOVAs summarised in Table 4.3. The reason for performing ANOVA on several combinations of semantic similarity measures is due to the fact that there are only half as many observations for Wang's approach as for the others, since Wang is not subject to the ancestor variable. This means that for the first row in Table 4.3 for example, the sample size for Wang was half the sample size of the other approaches, whereas it was the same for the analyses in the second and third rows.

Category	Variables	p-value
Semantic similarity	Lin, Resnik, Schlicker, Wang - *all values*	0.17
Semantic similarity	Lin, Resnik, Schlicker, Wang - *MICA values only*	0.01
Semantic similarity	Lin, Resnik, Schlicker, Wang - *GraSM values only*	0.72
Semantic similarity	Lin, Resnik, Schlicker	0.28
Functional simialrity	BMA vs MAX	0.16
Dataset	all vs nonIEA	0.40
Ancestors	MICA vs GraSM (*excl Wang*)	$4.4E^{-3}$

Table 4.3: p-values from single-factor ANOVA, for several combinations of variables.

From the p-values in Table 4.3, it is clear that in most cases, the differences across the full set of AUC values are insufficient to draw meaningful conclusions of the type "approach X always performs the best". The only real exception is the ancestor category, which has a p-value of $4.4E^{-3}$, which is much lower than the significance level of 0.05. It can therefore be concluded that the AUC values obtained if the MICA ancestor was used are significantly different from the AUC values if the GraSM algorithm was used. As the MICA AUCs are higher (average = 0.864) than the GraSM AUC values (average = 0.854), this allows the conclusion

that the use of MICA leads to better results than the use of GraSM. For this reason, the next set of analyses was performed on the MICA AUCs only.

Additionally, the results in Table 4.3 show that if only MICA-derived AUCs are considered, there are significant differences between the AUCs for the four semantic similarity approaches. As this is the result of a single-factor ANOVA, the other two sets of variables (functional similarity and dataset) were not considered and all AUCs for a given semantic similarity approach were treated equally.

As different sets of variables should not just be considered independently, a pair of two-factor ANOVAs with replication were performed. "with replication" means that if the two factors under consideration were for example semantic similarity and functional similarity, the AUCs obtained for each semantic similarity measure for different datasets were considered as part of the same "sample" rather than as individual ones. The results of these analyses are summarised in Table 4.4.

First factor	second factor	p-value first factor	p-value second factor	p-value interaction
Semantic similarity	Functional similarity	0.03	1.00	0.39
Semantic similarity	Dataset	4.9^{-2}	0.02	0.28

Table 4.4: p-values from two-factor ANOVA, for several combinations of variables.

The results of the two-factor ANOVAs show that there is no significant difference between the AUCs for semantic and functional similarity, nor is there any significant difference in the interaction of the two factors, i.e. no combination of semantic and functional similarity approaches leads to a significantly different set of AUCs. The same is the case for the interaction of semantic similarity and dataset variables. Individually however, both factors obtained p-values that are lower than the significance level of 0.05. This means that there are significant differences between the AUCs for different semantic similarity approaches for a given dataset and vice versa. As the AUCs for the full dataset are on average slightly higher (average = 0.865) than the AUCs for the non-electronic dataset (average = 0.857), this allows the conclusion that using the full dataset leads to overall slightly better performance than using the non-electronic one.

A final set of single-factor ANOVAs was therefore performed on the AUCs for all semantic similarity approaches and both functional similarity approaches, using only the full dataset and the MICA ancestor approach. If the factor under consideration was functional similarity, the resulting p-value was 0.76, i.e. there is no significant difference between the AUCs obtained for BMA and for MAX. If semantic similarity was used as the factor, a p-value of 0.05 was obtained, suggesting that the difference between the AUCs for the various semantic similarity approaches is statistically

significant.

4.3 Semantic similarity approaches

In this section, the four semantic similarity approaches by Resnik, Lin, Schlicker and Wang are compared under a number of different sets of variables to see if one or more of the approaches consistently performs better than the others. The Wang approach can only be included in a comparison if the ancestor parameter is kept constant as this approach is not subject to different types of ancestor selection.

4.3.1 Ancestor

	BMA		MAX			
	all	nonIEA	all	nonIEA		
			MICA		Total	*StDev*
Resnik	1	1	2	2	**6**	*1*
Schlicker	2	2	1	1	**6**	*1*
Lin	3	3	3	3	**12**	*0*
Wang	4	4	4	4	**16**	*0*

Table 4.5: Semantic similarity approaches for MICA

Table 4.5 shows that the approaches by Resnik and Schlicker perform equally well, with Resnik performing better for functional similarity approach BMA, while Schlicker performs better for functional similarity approach MAX. Semantic similarity using Lin or Wang consistently rank in third and fourth place, respectively.

	BMA		MAX			
	all	nonIEA	all	nonIEA		
			GraSM		Total	*StDev*
Schlicker	1	1	4	1	**7**	*2*
Resnik	2	2	1	4	**9**	*1*
Lin	3	3	3	3	**12**	*0*
Wang	4	4	2	2	**12**	*1*

Table 4.6: Semantic similarity approaches for GraSM

When the ancestor choice is "GraSM" (shown in Table 4.6), the ranking is slightly different. The approach by Schlicker performs better than any other approach in most cases except for "MAX-all", where it actually performs the worst. A similar situation exists for Resnik's approach, which performs well in most cases except for

"MAX-nonIEA", where it performs worst. Lin's approach is consistently ranked third. The approach by Wang performs worst for BMA but is ranked second under MAX, due to the two poorer performances by Resnik and Schlicker.

	BMA				MAX					
	all		nonIEA		all		nonIEA			
	MICA	GraSM	MICA	GraSM	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	2	1	1	3	1	1	**12**	*1*
Resnik	1	2	1	2	2	1	2	3	**14**	*1*
Lin	3	3	3	3	3	2	3	2	**22**	*0*

Table 4.7: Semantic similarity approaches, all combinations. The Wang approach has to be excluded from this comparison because the ancestor parameter varies across the comparison.

If Wang's approach is not considered, as in Table 4.7, which shows the rankings for Lin, Resnik and Schlicker for all possible combinations of functional similarity, annotation and ancestor choice, Schlicker's approach ranks highest overall, Resnik's second and Lin's third. Exceptions are found only for the previously discussed two cases, which in turn give Lin two higher rankings.

4.3.2 Annotations

	BMA		MAX			
	all					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	2	2	1	**6**	*1*
Schlicker	2	1	1	3	**7**	*1*
Lin	3	3	3	2	**11**	*1*

Table 4.8: Semantic similarity approaches, full dataset

If only results from the full annotation dataset (Table 4.8) are considered, Resnik's approach ranks highest as this includes the exceptionally poor ranking for Schlicker's approach. Lin's approach ranks worst overall. For the dataset of non-electronic annotation (Table 4.9) on the other hand, Schlicker performs best overall, whereas this comparison includes the exceptionally poor performance of Resnik's approach. Lin's approach again ranks the lowest.

4.3.3 Functional similarity approaches

For the BMA functional similarity approach, Schlicker and Resnik perform equally well, while Lin's approach performs consistently the worst. With MAX, Schlicker performs overall better than Resnik. This comparison again includes the two cases

	BMA		MAX			
	nonIEA					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	1	1	**5**	*1*
Resnik	1	2	2	3	**8**	*1*
Lin	3	3	3	2	**11**	*1*

Table 4.9: Semantic similarity approaches, non-IEA

	BMA					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	2	1	**6**	*1*
Resnik	1	2	1	2	**6**	*1*
Lin	3	3	3	3	**12**	*0*

Table 4.10: Semantic similarity approaches, BMA only

of exceptionally poor performances from Schlicker and Resnik already observed in other tables. Despite these cases, Lin's approach again ranks worst overall.

	MAX					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	1	3	1	1	**6**	*1*
Resnik	2	1	2	3	**8**	*1*
Lin	3	2	3	2	**10**	*1*

Table 4.11: Semantic similarity approaches, MAX only

4.3.4 Summary

Overall, Schlicker and Resnik perform almost equally well, with only a slightly better performance by the Schlicker approach. Lin's approach ranks consistently lower than the other two IC-based methods, although it performs better than Wang's approach, if the type of comparison allows the inclusion of the latter.

This trend is also found for the protein interaction and the phenotype datasets. In the gene expression dataset on the other hand, Resnik's method performs consistently worst, usually with Schlicker's method ranking highest and Lin second. If the ancestor choice is the constant factor, Lin even performs best for MICA and Wang for GraSM, although this latter ranking may be an artefact of the overall lower performance of the IC-based methods with GraSM, that makes the Wang approach,

which is independent of ancestor selection, appear to perform better when in fact its performance does not change.

In the next few sections, comparisons for all possible combinations of functional similarity, ancestor and annotation choices will be performed, either for all four semantic similarity approaches or, where this is not appropriate, for the three IC-based approaches.

4.4 Ancestors

For the combinations of parameters in Table 4.12 and Table 4.13, it is appropriate to include the approach by Wang in the comparison as the one parameter that does not apply to the Wang approach, ancestor choice, is kept as a constant. In this comparison, the full annotation dataset outperforms the non-electronic dataset if only the most common ancestor is used, while the BMA functional similarity performs worse than MAX due to the very poor performance of "BMA-all" for Wang's measure and its exceptionally good performance on "MAX-nonIEA". Table 4.13 shows that, unlike the results obtained with MICA, the results for GraSM show a better performance for BMA than for MAX. In addition, the two functional similarity approaches show opposite performance trends when the type of dataset is considered. BMA performs better on the full dataset whereas MAX performs better with non-electronic data.

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
MAX	all	MICA	2	2	1	2	**7**	*1*
BMA	all	MICA	1	1	2	4	**8**	*1*
MAX	nonIEA	MICA	3	4	3	1	**11**	*1*
BMA	nonIEA	MICA	4	3	4	3	**14**	*1*

Table 4.12: All annotation-MICA vs. non-IEA-MICA

Overall, it appears that while the IC-based measures perform better on the full dataset than the non-electronic one, Wang's performance is more related to the choice of functional similarity as the hybrid approach performs better for MAX than for BMA, regardless of the dataset. In addition, for each functional similarity approach, Wang's measure performs better with the non-electronic annotation data than with all annotations. This trend can be explained by the way the approach works. First of all, the better performance on the non-electronic data can be explained by the average depth of this dataset, which is around 7.7, compared to the dataset of all annotations, which is about 6.9. The deeper in the hierarchy two

terms are, the more likely it is that parts of their respective sub-graphs from term to root differ. As the full dataset contains more annotations with fairly shallow terms, the known drawback of Wang's approach can result in misleadingly high similarities between these shallower terms. This explains why, for either functional similarity approach, the non-electronic dataset performs better than the full annotation dataset. In addition, these misleadingly high similarities can bias the BMA approach and lead to a higher overall score between two gene products than might reasonably be expected. The MAX approach on the other hand performs better because the truly similar deeper terms will be those with the overall highest similarities.

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
BMA	all	GraSM	1	1	2	4	**8**	*1*
BMA	nonIEA	GraSM	2	2	1	3	**8**	*1*
MAX	nonIEA	GraSM	3	4	3	1	**11**	*1*
MAX	all	GraSM	4	3	4	2	**13**	*1*

Table 4.13: All annotation-GraSM vs. non-IEA-GraSM.

Considering that the Wang approach produced two outliers when compared to the MICA data (Table 4.12), compared to four outliers for the present comparison (Table 4.13), this may suggest that it is more appropriate to compare the Wang approach to the IC-based approaches using MICA than using GraSM.

As in Section 4.3, the protein interaction and phenotype datasets follow the results for the aggregate dataset fairly closely and there are only a few minor exceptions in the individual rankings. The gene expression dataset on the other hand once again opposes the general trend by performing better with the non-electronic dataset than with full annotations. In all cases, the performance of the functional similarity approaches is too variable to allow any conclusions to be drawn.

4.5 Annotations

When only results for the full annotation dataset are considered (Table 4.14), MICA always performs better than GraSM in terms of ancestor choice and BMA performs better than MAX, for both ancestor choices. The individual semantic similarity approaches show trends that are fairly consistent with the overall trend, with all minor variations within the defined limits.

Results for the non-electronic dataset (Table 4.15) show a different trend than those for the full dataset. "MAX-MICA" performs best of all, while "MAX-GraSM" performs worst of all. For BMA, GraSM performs better than MICA. In addition,

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	all	MICA	1	1	2	**4**	*1*
MAX	all	MICA	2	2	1	**5**	*1*
BMA	all	GraSM	3	3	3	**9**	*0*
MAX	all	GraSM	4	4	4	**12**	*0*

Table 4.14: All annotation - MICA vs. GraSM

Resnik's approach shows several outliers when compared to the overall trend. More specifically, Resnik shows the same trend for the non-electronic dataset as for the full dataset, suggesting that the kind of annotation used does not affect this approach as much as Lin and Schlicker.

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	nonIEA	MICA	1	2	2	**5**	*1*
BMA	nonIEA	GraSM	2	3	1	**6**	*1*
BMA	nonIEA	MICA	4	1	3	**8**	*2*
MAX	nonIEA	GraSM	3	4	4	**11**	*1*

Table 4.15: Non-IEA - MICA vs. GraSM

Overall, approaches using MICA can be said to outperform approaches with GraSM, while there is no overall conclusion for the functional similarity approaches. In the gene expression and phenotype datasets on the other hand, BMA always performs better than MAX, while MICA performs better than GraSM on the full dataset and vice versa if only non-electronic annotation is used. The protein interaction dataset shows less clear trends. Although the single ancestor approaches generally outperforms the disjoint ancestor approach, the performance of the functional similarity approaches is too varied to draw any conclusions.

4.6 Functional similarity approaches

If only the results for functional similarity calculated using BMA are considered (Table 4.16), the full dataset performs better than the non-electronic dataset, but while MICA outperforms GraSM on the full dataset, the reverse occurs for the non-electronic data. In addition, there are two outliers to the overall trend.

For functional similarity using the MAX approach (Table 4.17), MICA performs overall better than GraSM but while the full dataset outperforms the non-electronic dataset for MICA, the reverse is true for GraSM. The overall trend for combinations of either dataset with MICA are the same for both functional similarity approaches, but they are different when it comes to GraSM.

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	all	MICA	1	1	1	**3**	*0*
BMA	all	GraSM	2	3	3	**8**	*1*
BMA	nonIEA	GraSM	3	4	2	**9**	*1*
BMA	nonIEA	MICA	4	2	4	**10**	*1*

Table 4.16: BMA only

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	all	MICA	1	1	1	**3**	*0*
MAX	nonIEA	MICA	2	2	2	**6**	*0*
MAX	nonIEA	GraSM	3	4	3	**10**	*1*
MAX	all	GraSM	4	3	4	**11**	*1*

Table 4.17: MAX only

In the gene expression dataset on its own, non-electronic annotation always performs better than the full annotation, for both functional similarity approaches. GraSM performs best in conjunction with non-electronic annotation, while the opposite is true for MICA. In the other two individual datasets, full annotation always performs best with BMA and MICA outperforms GraSM, while with MAX, full annotation outperforms non-electronic annotation if MICA is used. Of the three individual dataset, the behaviour seen in the protein interaction dataset most closely matches the aggregate dataset.

4.7 Summary

From these comparisons, it is clear that Resnik's and Schlicker's approaches perform better overall than the approaches by Lin and Wang. In fact, the only times that either of the latter two rank higher than third or fourth place respectively is in one of the two cases when the former two perform exceptionally badly, namely "MAX-nonIEA-GraSM" for Resnik and "MAX-all-GraSM" for Schlicker. In terms of the individual datasets, the protein interaction and phenotype datasets provide very similar trends, in that Resnik's and Schlicker's approaches have the best performance although their respective rankings vary. Lin's and Wang's approaches rank worst in both cases. The gene expression dataset agrees with Schlicker's high performance but has a very poor performance for Resnik. Nonetheless, the approaches to be carried forward into the next part of the analysis will be Resnik and Schlicker as they perform best in the highest number of cases.

In terms of variable choices, the full annotation dataset usually performs better

than the non-electronic dataset, while the MICA usually performs better than the GraSM algorithm, except for the gene expression dataset. In some cases, the better performance of MICA is tied to the annotation dataset. In that respect, the full annotation dataset generally performs better than the non-electronic dataset, except in conjunction with "MAX-GraSM". This combination will be eliminated by carrying forward MICA as ancestor choice and the full dataset for annotation.

The results are somewhat less clear-cut for the two functional similarity approaches. BMA usually performs better than MAX on the full dataset whereas MAX usually performs better than BMA on the non-electronic dataset. There is however a high enough level of variability in all the datasets so that no clear conclusion can be drawn. In some cases, BMA appears to perform best overall, while MAX has a better performance in others. For this reason, both approaches will be carried forward.

In the next chapter, the selection of thresholds for the grouping algorithm will be discussed before results for the grouping algorithm itself are presented.

Chapter 5

Threshold determination

In the previous chapter, the most appropriate semantic and functional similarity approaches and other variables were determined. The semantic similarity approaches that performed the best overall were those by Resnik [1995] and Schlicker et al. [2006], while neither the "BMA" nor the "MAX" functional similarity approach consistently performed better than the other. The best results were obtained from the aggregate rFunSim score rather than for the individual ontological scores. In most cases, the full annotation performed better than the non-electronic annotation data and the single ancestor selection gave overall better results than the GraSM algorithm using disjoint ancestors.

However, before these selected approaches can be applied in the FuSiGroups algorithm, it is necessary to determine the minimum and maximum semantic and functional thresholds to be used for each combination of variables.

Determining a set of appropriate semantic and functional thresholds for each approach is essential in order to generate optimum groups. The strategy for determining grouping thresholds was described in Section 3.2.2. In short, accuracy curves were generated using ROCR's *acc* parameter for the *performance()* method. For the minimum threshold, the highest accuracy value for a given curve was determined using the *max()* method on the Y-axis values of the graph. From the index or coordinate in the list of datapoints of the highest Y-value, the corresponding threshold on the X-axis was then determined. Each threshold was rounded to two significant figures as there is no evidence to suggest that a higher level of accuracy is necessary.

The maximum threshold was defined as the largest similarity value corresponding to an accuracy that is smaller than the maximum accuracy by 15% of the range of accuracy values. The reason for using the largest similarity value is that due to the quasi-parabolic nature of the accuracy curve, each accuracy value below maximum

corresponds to at least two similarity values, one on each "side" of the curve. Since the minimum threshold corresponds to the point of maximum accuracy, the maximum threshold needs to be a larger value and therefore needs to be the largest of the two or more, in the case of a very irregular curve, similarity values corresponding to the required accuracy.

For the data presented here, the maximum thresholds correspond to accuracies of 70% for the semantic similarity approaches and 75% for the functional similarity approaches. As can be determined from Figures 5.1, 5.3, 5.6 and 5.8, and from Tables 5.1, 5.3, 5.6 and 5.8, the ranges for all curves are between 0.26 and 0.3, and therefore 15% of those ranges would be 0.04 and 0.05, respectively. By subtracting these values from the maximum accuracies and rounding the result to the nearest 0.05, accuracies of 70% and 75% were obtained.

For the maximum threshold, the selection procedure of the cut-off value was slightly less direct than for the minimum threshold. The set of accuracy values corresponding to thresholds greater than the minimum threshold and ranging from 0.748 to 0.752 (or 0.698 to 0.702, respectively) were selected as the *performance()* method would not necessarily generate Y-values of 0.75 (or 0.70) exactly. It was therefore not possible to pick the exact accuracy and find its matching threshold, as with the minimum threshold. From the range of Y-values, the value closest to the specified accuracy was selected and its respective X-value, rounded to two significant figures, was used as the maximum threshold. In the majority of cases, the difference between the threshold values corresponding to the accuracies closest to the specified accuracy level was no greater than 0.01, i.e. rounding to two significant figures would generate the same threshold for any of these values. For this reason, the potential unavailability of an accuracy of exactly 0.75 (or 0.70) has no significant impact on the selection of the maximum threshold.

5.1 Semantic thresholds

The semantic thresholds for each approach were determined using the individual ontology scores (rather than the composite rFunsim) for a set of gene product pairs. The ontological scores were calculated using the "MAX" functional similarity approach so that each score represents the closest GO term pair of that ontological aspect for a pair of gene products. In the absence of a benchmark dataset of true positive GO term pairs and true negative GO term pairs, this approach is the closest approximation to such a benchmark. It is based on the assumption that two highly similar gene products are annotated with highly similar GO terms, whereas two very

dissimilar gene products are annotated with very dissimilar GO terms. Since there are three ontological aspects in the GO but only one single semantic threshold across all aspects, the three sub-datasets for a given approach are aggregated into a large dataset in order to generate a single accuracy curve from which to deduce a single set of semantic thresholds.

5.1.1 Resnik

The accuracy curve for Resnik's semantic thresholds is shown in Figure 5.1. As discussed in Section 3.2.2, the minimum and maximum semantic thresholds are too close together to establish a range of thresholds if an accuracy of 75% is used. In fact, a closer look at the actual values reveals that the highest accuracy value is 0.7595667 and corresponds to the point in the graph immediately preceding the drop to 0.7161556. This would result in identical minimum and maximum semantic thresholds, which is why the accuracy for the maximum threshold was redefined to 70%.

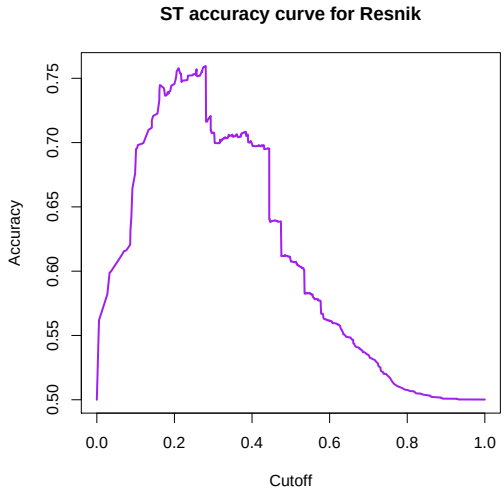

Figure 5.1: Accuracy curve for the semantic thresholds for Resnik

From the data underlying the accuracy curve, the point of maximum accuracy and the corresponding threshold can be deduced. The result is shown in Table 5.1:

Accuracy	Cutoff	Minimum ST
0.760	0.281	0.28

Table 5.1: Minimum ST for Resnik

By selecting data points with a cutoff greater than the minimum ST and an

accuracy between 0.698 to 0.702, the maximum semantic threshold can also be deduced, as shown in Table 5.2.

Accuracy	Cutoff	Rounded	Maximum ST
0.701	0.397	0.40	0.40
0.701	0.397	0.40	
0.701	0.398	0.40	
0.697	0.402	0.40	

Table 5.2: Maximum ST for Resnik

Figure 5.2 shows the ROC curve for the same dataset, with the range of semantic thresholds displayed. The thresholds are clustered around the highest left-most part of the curve, which represents the optimum trade-off in identification of true positives and true negatives.

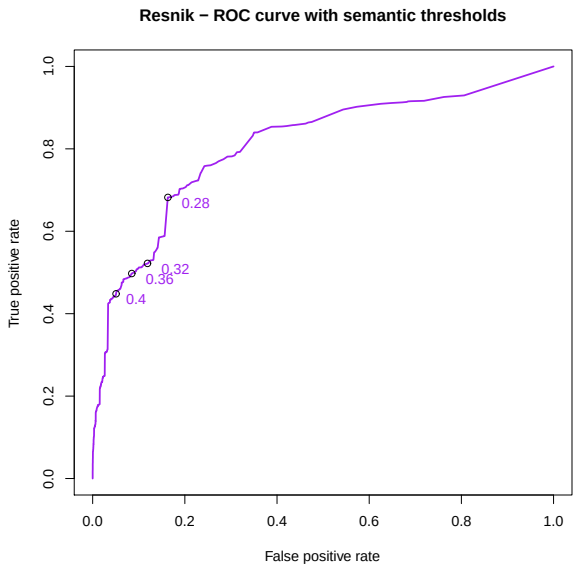

Figure 5.2: ROC curve showing the semantic thresholds for Resnik

5.1.2 Schlicker

Figure 5.3 shows the accuracy curve for Schlicker's semantic thresholds. Unlike the ST accuracy curve for Resnik, Schlicker's accuracy rises gradually along almost all cutoffs, then drops suddenly at very high thresholds.

Due to the constant rise in accuracy found in semantic similarity according to Schlicker, the minimum semantic threshold is very high, as shown in Table 5.3:

The sharp drop following the curve's peak means that the minimum and maximum thresholds are extremely close together, as seen in Table 5.4.

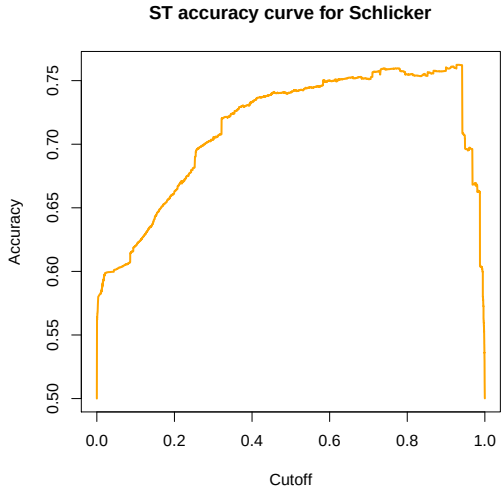

Figure 5.3: Accuracy curve for the semantic thresholds for Schlicker

Accuracy	Cutoff	Minimum ST
0.762	0.927	0.93

Table 5.3: Minimum ST for Schlicker

Accuracy	Cutoff	Rounded	Maximum ST
0.704	0.948	0.95	0.95
0.704	0.949	0.95	
0.697	0.949	0.95	
0.696	0.950	0.95	

Table 5.4: Maximum ST for Schlicker

Figure 5.4 clearly shows that the minimum and maximum thresholds for Schlicker are located at the highest left-most part of the ROC curve. It should be noted that in TP/TN trade-off terms, the thresholds of 0.93 and 0.94 are so close together that they are indistinguishable at the resolution used in Figure 5.4.

The semantic thresholds for Resnik and Schlicker are clearly very different. While Resnik's thresholds are lower and cover a greater range of similarity values, Schlicker's thresholds are very high and the minimum and maximum thresholds are very close together. Figure 5.5 shows the distribution of semantic similarity values for both approaches for all possible GO term pairs of terms annotated to the Eisen dataset. Similarity is only calculated between terms from the same sub-ontology as terms that are not from the same sub-ontology do not have any common ancestors and therefore have a similarity of 0. Nonetheless, similarities of 0 (or so close to 0 that any rounding reduces them to 0) are clearly very frequent. The percentage of similarity values equal to 0, smaller than the minimum ST, greater than the maxi-

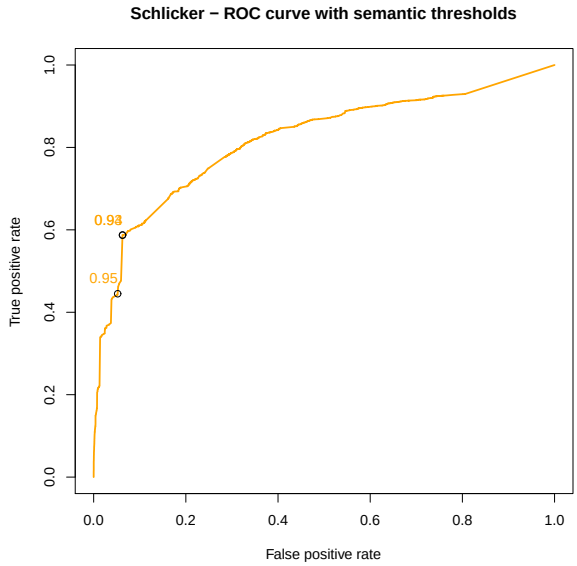

Figure 5.4: ROC curve showing the semantic thresholds for Schlicker

mum ST and lying within the range of STs are all shown in Table 5.5. For Resnik, similarities of 0 make up almost 50% of all values, while the percentage for Schlicker is around 33%. For both approaches, the majority of similarity values (94% and 99% respectively) are lower than the minimum semantic threshold. This is not as surprising as it may seem, especially considering the distributions in Figure 5.5 and what they represent. Clearly, a very large number of similarity values are very low, which can be expected in an all-against-all comparison of a set of terms as diverse as GO term annotations.

The large difference in the respective ranges of the semantic thresholds for the two approaches also makes more sense if Figure 5.5 is considered. Although Schlicker has a very high proportion of term pairs without any similarity, the distribution of similarity values greater than 0.1 is much more even than that of Resnik's results. However, it is also clear from both the histograms and Table 5.5 that the percentage of similarity values within the semantic threshold range is much lower for Schlicker than it is for Resnik. Even though the two sets of thresholds were derived experimentally based on a set of pre-defined criteria, their use may reveal that they are not equally suitable to the FuSiGroups algorithm.

5.2 Functional thresholds

Determining functional thresholds is a lot simpler than determining semantic thresholds as the true postive/true negative datasets described in Chapter 3 represent pairs

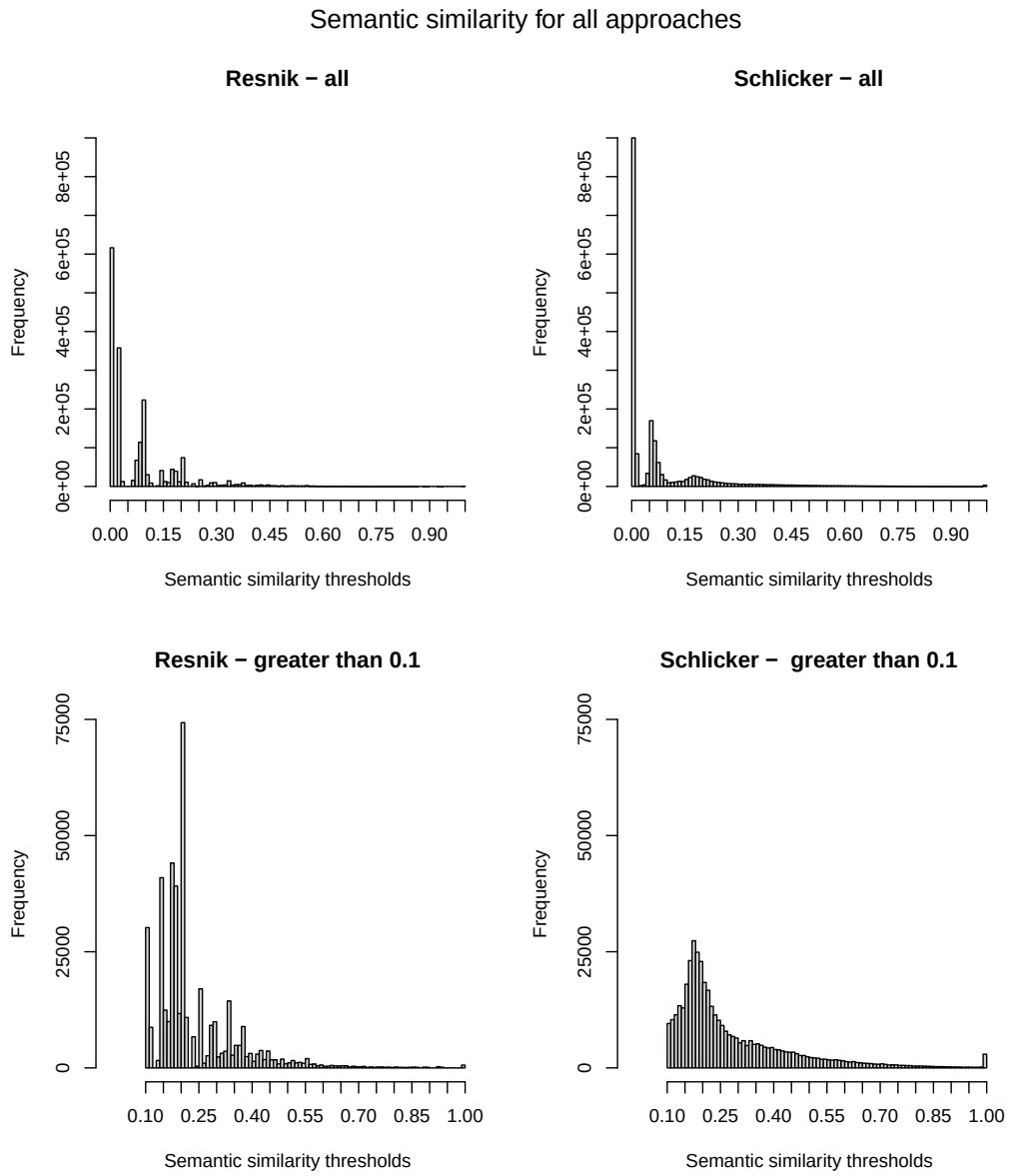

Figure 5.5: Semantic similarity distributions for Resnik and Schlicker for all GO terms found in the annotation of the Eisen dataset. As there are a great number of GO term pairs with a similarity of 0, histograms for all thresholds and for thresholds greater than 0.1 are shown so that thresholds with lower frequencies can be seen. Note that the histograms for all values and the histograms for values greater than 0.1 have different Y-axes.

Range	Resnik	Schlicker
similarity = 0	49.27%	33.73%
similarity < minST	94.05%	99.79%
similarity > maxST	2.14%	0.19%
minST ≤ similarity ≤ maxST	3.80%	0.02%

Table 5.5: Percentage of GO term pairs within different ranges of semantic similarity values, for both Resnik and Schlicker

of gene products. Therefore, accuracy curves based on rFunSim are going to directly represent the accuracy of each approach for the datasets.

5.2.1 Resnik - BMA & MAX

The accuracy curves for functional similarity for both the BMA and the MAX functional similarity approaches based on Resnik's semantic similarity are shown in Figure 5.6. The curve for BMA (red) is taller and narrower than the MAX curve (green) and it also covers a lower range of thresholds. This is reflected in the distribution of functional similarity values for the full Eisen dataset (Figure 5.10): while Resnik-BMA spans a smaller range of values at higher frequencies, Resnik-MAX spans a wider range with a lower peak in frequencies.

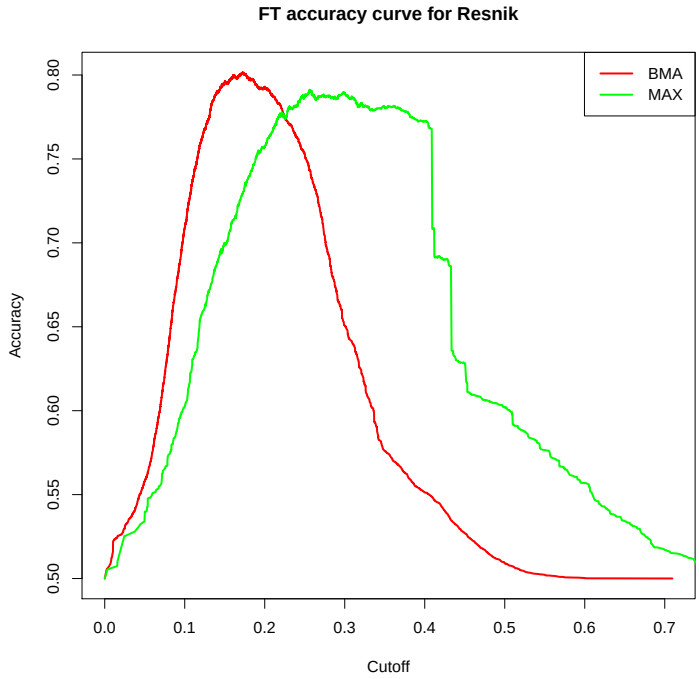

Figure 5.6: Accuracy curves for the functional thresholds for Resnik, for both BMA and MAX

From the accuracy curves and the underlying data, the thresholds corresponding to the point of maximum accuracy can be deduced. The corresponding functional similarity value represented the minimum functional threshold (FT) for the respective approach (Table 5.6).

While the maximum semantic threshold was defined as the largest cut-off corresponding to an accuracy of 70%, the maximum functional threshold is defined for an accuracy of 75% as the accuracies for functional similarity are generally higher

Approach	Accuracy	Cutoff	Minimum FT
BMA	0.802	0.172	0.17
MAX	0.791	0.256	0.26

Table 5.6: Minimum FTs for Resnik

than for semantic similarity. Table 5.7 shows the accuracies closest to 75% (if no accuracy value of exactly 0.75 is available) for both Resnik-BMA and Resnik-MAX. The corresponding cut-offs are shown at both the level of precision calculated by the FuSiGroups software and rounded to two decimal places (the precision used for analysis purposes). Once rounded, all the thresholds are the same and give the value of the maximum FT.

Approach	Accuracy	Cutoff	Rounded	Maximum FT
BMA	0.750	0.252	0.25	0.25
	0.750	0.252	0.25	
	0.750	0.252	0.25	
	0.750	0.252	0.25	
MAX	0.768	0.408	0.41	0.41
	0.768	0.409	0.41	
	0.708	0.409	0.41	
	0.706	0.410	0.41	

Table 5.7: Maximum FTs for Resnik

Highlighting the range between the minimum and maximum FTs on the respective ROC curves (Figure 5.7) shows that the thresholds are in the top left-most section of the curve, i.e. the area representing the best trade-off between true positives and true negatives. It can also be noted that the thresholds for BMA are slightly more clustered than the thresholds for MAX. In fact, the maximum FT for MAX is at the lowest point of all the values shown on the curve. In addition, the red curve seems to suggest that BMA performs marginally better than MAX (see Chapter 4).

5.2.2 Schlicker - BMA & MAX

The accuracy curves for BMA (blue) and MAX (yellow) functional similarity using Schlicker, shown in Figure 5.8, have the same origin and similar end points but the bodies of the two curves are offset in relation to each other. More specifically, the curve for BMA rises more sharply than the MAX curve and it is only slightly asymmetric, i.e. it falls at a similar rate as it rises. The curve for MAX on the other hand

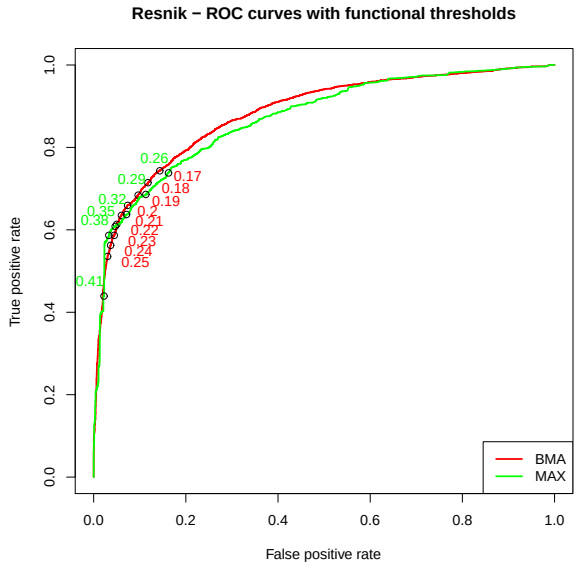

Figure 5.7: ROC curves showing the functional thresholds for Resnik, both for BMA and MAX. A selection of functional similarity values between the minimum and maximum FTs are included to illustrate their distribution on the ROC curve.

rises far more slowly, then drops fairly sharply after its highest point. The similarity in start and end points as well as the shapes of the two curves are reflected by the distribution of the functional similarity values on the full Eisen dataset for the two methods (Figure 5.10). Both approaches have at least some values at both extremities of the functional similarity range ([0,1]). The histogram for Schlicker-BMA is left-skewed with higher individual frequencies while the histogram for Schlicker-MAX shows a more even distribution across the range of possible values, with lower individual frequencies.

The functional thresholds for both approaches using Schlicker's semantic similarity were determined using the same approach as for Resnik. The minimum FTs are given in Table 5.8.

Approach	Accuracy	Cutoff	Minimum FT
BMA	0.795	0.422	0.42
MAX	0.796	0.664	0.66

Table 5.8: Minimum FTs for Schlicker

The maximum FTs, also determined in the usual fashion and for an accuracy of 75%, are listed in Table 5.9.

Once again, the thresholds occupy the highest left-most part of the ROC curves for the respective approaches (Figure 5.9). The two curves are very close, suggesting similar performance qualities for the two approaches, and while the actual values

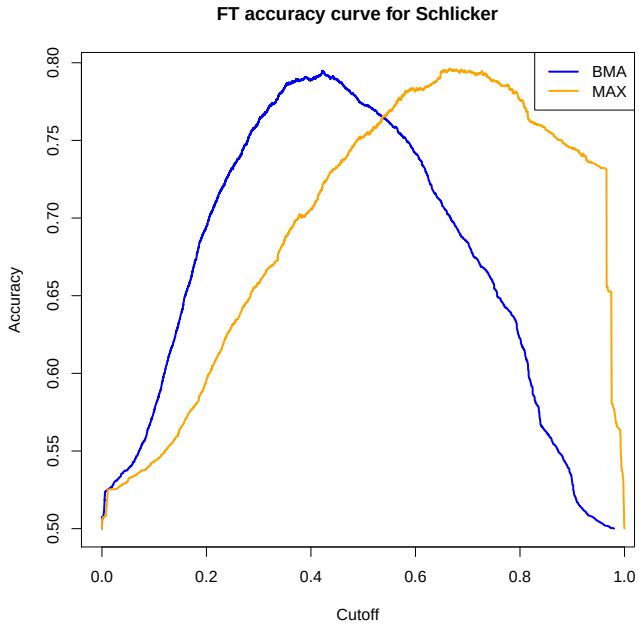

Figure 5.8: Accuracy curves for the functional thresholds for Schlicker, for both BMA and MAX

Approach	Accuracy	Cutoff	Rounded	Maximum FT
BMA	0.750	0.582	0.58	0.58
	0.750	0.582	0.58	
	0.750	0.583	0.58	
	0.750	0.583	0.58	
MAX	0.750	0.879	0.88	0.88
	0.750	0.879	0.88	
	0.750	0.880	0.88	
	0.750	0.881	0.88	

Table 5.9: Maximum FTs for Schlicker

of minimum and maximum FTs are different, the locations of the points are closely matched on the two curves.

Due to the different natures of the two semantic similarity (Resnik & Schlicker) approaches and the two functional similarity (BMA & MAX) approaches, the distributions of the results of these approaches are quite different from each other (see Figure 5.10). Both functional similarity approaches using Resnik for semantic similarity cover a much smaller range of functional similarity values than the same approaches using Schlicker for semantic similarity. This is due to the fact that Resnik's approach is based solely on the most informative common ancestor (MICA) of two GO terms which has a lower information content than the query terms. In

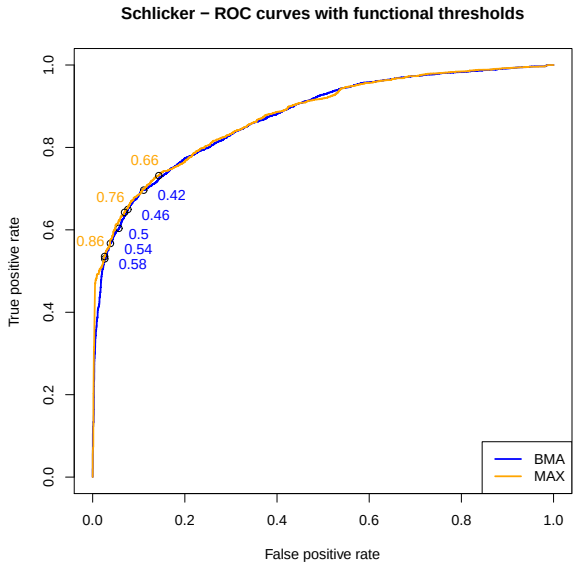

Figure 5.9: ROC curves showing the functional thresholds for Schlicker, both for BMA and MAX. Note that due to the nature of the step function used to display the sets of thresholds on the curve, the range for the MAX curve (yellow) ends at 0.86 rather than 0.88

order to bring results from Resnik's approach into the range [0,1], they need to be normalised by division with $maxIC$[1].

Schlicker's approach on the other hand is already normalised because its calculation includes the division of the IC_{MICA} by the sum of the information content of the two query terms. In addition, Schlicker's approach is weighted by multiplication with $1 - ln(p(MICA))$[2], increasing the higher similarities for more specific common ancestors and lowering the similarities for more generic common ancestors.

In terms of functional similarity patterns, the distributions for both semantic similarity approaches with BMA are more left-skewed than the two approaches with MAX. Similarities calculated using MAX would generally be expected to be higher than similarities calculated using BMA since MAX uses only the single most similar pair of GO terms for a given ontology while BMA uses the best match for each GO term, then averages them.

The percentage of functional similarity values in an all-against-all comparison of the Eisen dataset falling into the different ranges of "below minFT", "above maxFT" and "between minFT and maxFT" are shown in Table 5.10. As with the distribution of semantic similarity values, the majority of values lies below the minimum FT. The fraction of values between and above the FTs is however greater than for the STs, particularly for Schlicker.

[1] $maxIC = -ln(\frac{1}{N})$, where N is the total number of terms in the corpus
[2] $p(MICA)$ is the probability of occurrence of the most informative common ancestor

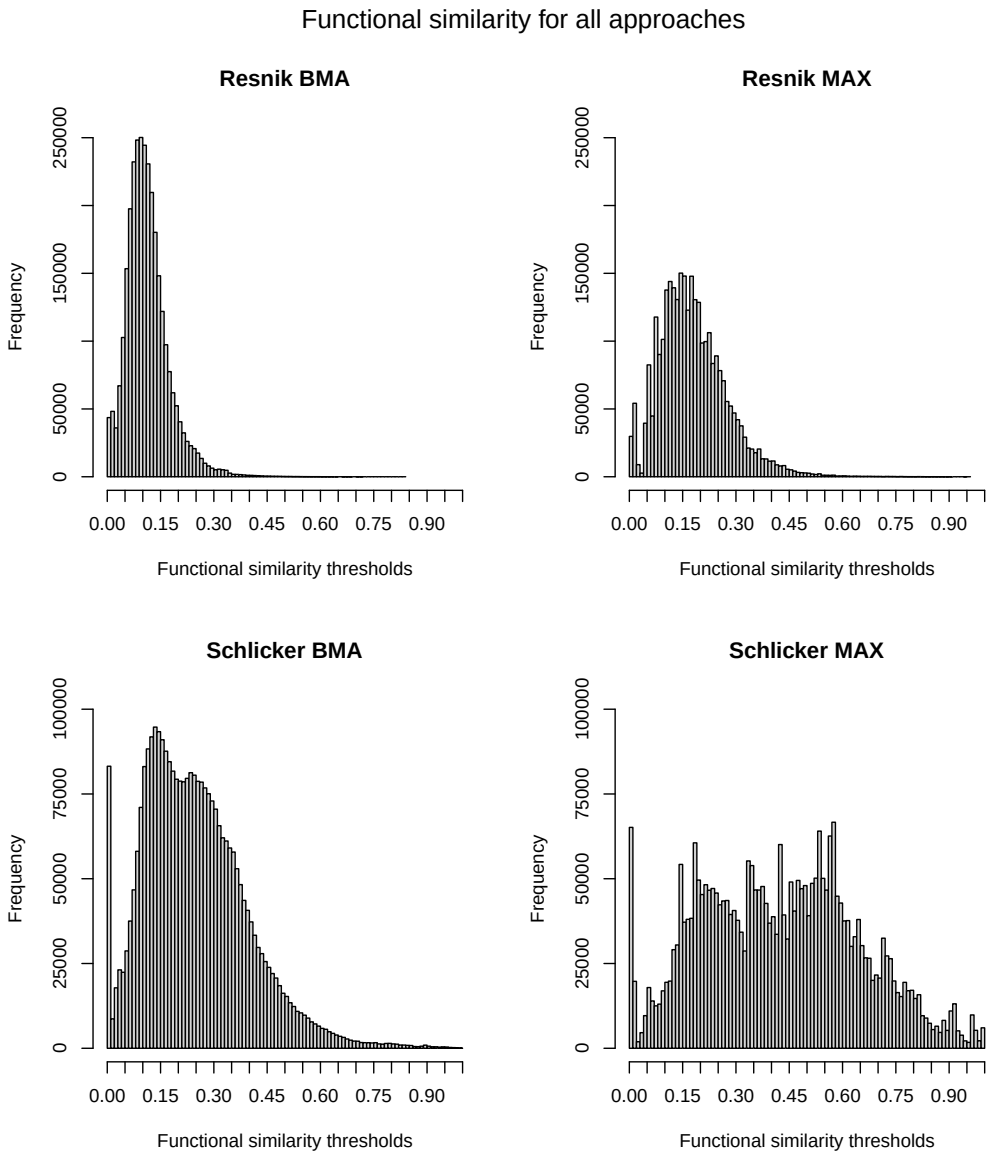

Figure 5.10: Functional similarity distributions for Resnik and Schlicker. Note that the scale of the Y-axes for Resnik and Schlicker are not the same.

	Resnik		Schlicker	
Range	BMA	MAX	BMA	MAX
similarity $<$ minST	85.93%	82.48%	88.38%	85.07%
similarity $>$ maxST	3.07%	2.27%	2.62%	2.43%
minST $\leq$ similarity $\leq$ maxST	11.00%	15.25%	8.99%	12.50%

Table 5.10: Percentage of gene product pairs within different ranges of functional similarity values, for both BMA and MAX in conjunction with Resnik and Schlicker. The percentage of gene product pairs with a similarity of 0 is not shown as it represents only a small fraction of the overall distribution.

5.3 Summary

In this chapter, semantic and functional thresholds for the different semantic and functional similarity approaches were derived using the true positive/true negative datasets described in Section 3.2.1. The same strategy was used for determining both types of thresholds, although the semantic thresholds had to be derived indirectly from a gene product dataset, as no benchmark dataset for semantic similarity between GO terms exists. Consistent with the distribution of similarity values, thresholds for Resnik are always slightly lower than thresholds derived for Schlicker.

The experimentally derived thresholds can now be used in the FuSiGroups algorithm in order to generate groups of functionally related genes. In the next three chapters, the results of these groupings with different parameters will be analysed, first from a high-level perspective, in terms of the general trends found in each set of groups (Chapter 6), then in greater detail, looking at some groups and smaller datasets in greater details (Chapters 7 and 8).

Chapter 6

Grouping trends

The purpose of the FuSiGroups algorithm is the generation of groups of functionally related gene products. Each group has a *definition* of one or more semantically similar GO terms. This definition characterises the functional aspect on which the gene products in the group are related. The level of similarity between the GO terms in the definition and between the gene products in the group are determined by the thresholds derived in the previous chapter. Although both types of thresholds cover a range of similarity values, the testing of the grouping algorithm will only be performed on combinations of minimum and maximum thresholds in order to avoid excessive repetition. Table 6.1 shows all the combinations of parameters for which the algorithm was run. As each ST-FT combination uniquely identifies a given experiment, this notation will be used from here on to refer to experiments, rather than the full name of the approaches, i.e. ST40-FT26 would refer to an experiment using the Resnik semantic similarity approach with MAX functional similarity, semantic threshold of 0.40 and functional threshold of 0.26.

In the subsequent analysis of grouping results, unless otherwise specified, any reference to number of groups will refer to *meaningful groups*, as opposed to *total number of groups* for a given combination of thresholds. As a reminder, a meaningful group is a group with four or more gene products, as defined in Definition 6 in Section 3.1.6.

The first part of the analysis, in this chapter, focusses on the high-level trends of the results such as number of groups generated, group sizes and definition sizes. Chapters 7 and 8 will then focus on the contents and definitions of a set of relevant examples in greater detail in order to evaluate the results generated by the FuSiGroups approach and compare them to both functional and expression clustering.

Semantic sim.	Functional sim.	ST	FT
Resnik	BMA	0.28	0.17
			0.25
		0.40	0.17
			0.25
	MAX	0.28	0.26
			0.41
		0.40	0.26
			0.41
Schlicker	BMA	0.93	0.42
			0.58
		0.95	0.42
			0.58
	MAX	0.93	0.66
			0.88
		0.95	0.66
			0.88

Table 6.1: Combinations of experimental parameters for which the FuSiGroups algorithm was run.

6.1 Number of groups

The first factor that should be analysed in a high-level context is the number of groups generated for a given set of parameters. Table 6.2 shows the total number of groups and the number of meaningful groups for each set of thresholds. It is noteworthy that while the total number of groups remains the same for a given semantic threshold, the number of meaningful groups varies with each functional threshold. This is because the total number of groups is exclusively determined by the group definition, which is only dependent on the ST. The number of meaningful groups on the other hand is defined by group size, i.e. group content, which is dependent on the FT. The higher the FT, the closer the genes in a group have to be related and therefore the fewer meaningful groups there are as more groups have too few genes to be considered meaningful.

One exception to the same total number of groups per ST occurs in the case of the maximum FT for Schlicker-MAX (FT = 0.88). In this particular case, for both minimum and maximum ST, the total number of groups is lower than the total number of groups found for the minimum FT (FT = 0.66). The 38 and 41 groups that disappear in the two cases are groups that at minimum FT have a size of 1 (data not shown). In each case, the single gene is one for which the functional similarity with itself is lower than the maximum FT. When the group content is

checked against the maximum completeness rule set by the functional threshold, the one gene is found to be in violation of the rule and is removed. The resulting empty group is discarded when the results are saved.

ST	No. groups	FT	No. "meaningful" groups
0.28	481	0.17	397
		0.25	387
0.40	740	0.17	564
		0.25	539
0.28	481	0.26	401
		0.41	401
0.40	740	0.26	572
		0.41	567
0.93	2564	0.42	972
		0.58	789
0.95	2693	0.42	963
		0.58	782
0.93	2564	0.66	1001
	*2526**	0.88	757
0.95	2693	0.66	999
	*2652**	0.88	749

Table 6.2: Total number of groups and number of meaningful groups for each combination of semantic and functional thresholds. The total number of groups is the same for a given semantic threshold, regardless of the associated functional threshold, as the number of groups is decided by the group definition, which is dependent only on the ST. Note that for maxFT for Schlicker-MAX (FT=0.88), the total number of groups is slightly lower than the normal total number of groups for that threshold (smaller numbers marked with *). The total number of groups is lower here than for the minimum FT because several groups at minFT, all of group size = 1, contained a gene whose similarity with itself was less than maxFT. The gene was therefore removed at maxFT for violating the maximum completeness rule, resulting in an empty group, which was discarded.

There is a very notable difference in the number of groups lost from total to meaningful groups between groups generated from Resnik's approach and groups generated using Schlicker's approach. For Resnik's approach, groups with four or more gene products represent roughly around 80% of total groups. For Schlicker's approach, they generally represent around 35% of total groups. In addition, the drop, for Resnik, is overall greater for the maximum ST (between 75% and 80% of total groups) than for the minimum ST (between 80% and 83% of total groups), with little or no effect observed for varying FT values. For Schlicker, the different STs have less of an effect than the difference between minimum and maximum FTs. In fact, the percentage of meaningful groups for the minimum FTs usually lies closer to 40%, while the percentage for the maximum FTs can be lower than 30% of total

groups.

In relation to Resnik's thresholds, these findings suggest that while a higher ST means a larger number of groups, these groups have a tighter definition and therefore fewer gene products will match the definition. This in turn leads to fewer groups with enough gene products to be counted as meaningful groups. For Schlicker, the small difference between the minimum and maximum STs is the most likely explanation for the very minor change in number of meaningful groups for different STs. The difference between minimum and maximum FT on the other hand is greater for Schlicker than for Resnik, matching the wider range of similarity values for Schlicker compared to Resnik (Figure 5.10). A higher FT is therefore more likely to have an effect on group size and, by extension, number of meaningful groups for Schlicker than for Resnik.

			Average size	
ST	FT	Max.size	all groups	meaningful groups
0.28	0.17	177	43.67	52.52
	0.25	112	20.76	25.35
0.40	0.17	170	22.41	28.85
	0.25	108	12.14	15.98
0.28	0.26	261	58.45	69.77
	0.41	75	22.56	26.71
0.40	0.26	175	33.31	42.58
	0.41	75	13.94	17.65
0.93	0.42	214	6.21	13.77
	0.58	121	4.28	10.33
0.95	0.42	211	5.70	13.09
	0.58	121	3.98	9.86
0.93	0.66	311	7.15	15.80
	0.88	191	4.63	11.83
0.95	0.66	306	6.44	14.68
	0.88	173	4.23	11.09

Table 6.3: Maximum and average group sizes for all threshold combinations. Average group sizes are calculated for both all groups and meaningful groups only.

The great difference in number of both total groups and meaningful groups between groupings based on Schlicker and groupings based on Resnik could be an indication that the semantic thresholds determined for Schlicker are too high. Based on the parameters defined for the determination of semantic and functional thresholds, it might have been possible to define a lower minimum ST for Schlicker. The accuracy curve in Figure 5.3 has a plateau around a cut-off value of 0.75 that appears to correspond to accuracy values that are fairly close to maximum accuracy. However,

a series of groupings based on randomly selected semantic thresholds for Schlicker revealed that the number of groups generated only drops below 1000 groups at an ST of about 0.45. In light of this, it was decided that even a less rigorously defined minimum ST would not considerably change the grouping results.

6.2 Group content

6.2.1 Group sizes

Since the number of meaningful groups goes down as the FT increases, it would make sense for the overall group size to also decrease as the FT increases, since fewer genes will have the necessary level of similarity. This trend is indeed observable from the data for a given approach (see Table 6.3), i.e. the maximum group size is lower for ST28-FT25 than it is for ST28-FT17. However, this does not extend across different functional similarity approaches, i.e. the largest group for ST28-FT26 (a set of thresholds associated with the MAX approach) is larger than the largest group for ST28-FT25 (associated with the BMA approach). The same is of course also true for average group sizes, when all groups are used to calculate the average group size and when only the meaningful groups are used.

While the maximum group sizes would in general not allow any conclusion with regard to which semantic similarity approach was used, there is a very clear difference between average group sizes for groups based on Resnik and groups based on Schlicker. If Resnik's approach is used, average group size is substantially bigger than if Schlicker's approach is used. This is the case for average group sizes across all groups and across only meaningful groups. The reason for this trend is that a larger proportion of groups based on Schlicker's approach have small group sizes compared to the groups based on Resnik's approach. This in turn suggests that either the thresholds derived for Schlicker create groups that are too tight, or that the thresholds derived for Resnik create groups that are not tight enough.

Both semantic and functional thresholds can affect the tightness of the groupings in this respect, as a too tight ST generates an excessively restrictive group definition while an insufficiently high ST can generate a too general group definition. The FT in turn affects the similarity of the genes within the group, determining how tightly related they are. Considering the percentage of pairs of GO terms and pairs of gene product that are greater than the respective thresholds (Tables 5.5 and 5.10), it is clear that Resnik's thresholds allow a greater percentage of pairs of both types to potentially be included in their respective part of the groups. Schlicker's

thresholds on the other hand allow a smaller percentage of pairs to potentially be used, particularly for the GO term pairs, where only around 0.2% of pairs have a similarity greater than either ST.

		BP		CC		MF	
ST	FT	total groups	meaningful	total groups	meaningful	total groups	meaningful
0.28	0.17	247	231	165	109	69	57
	0.25		227		106		54
0.40	0.17	347	306	194	124	199	134
	0.25		293		121		125
0.28	0.26	247	234	165	109	69	58
	0.41		234		109		58
0.40	0.26	347	310	194	124	199	138
	0.41		307		124		136
0.93	0.42	1130	496	437	228	997	248
	0.58		382		197		210
0.95	0.42	1204	487	450	230	1039	246
	0.58		374		200		208
0.93	0.66	1130	514	437	229	997	258
	0.88	1123*	357	429*	191	974*	209
0.95	0.66	1204	512	450	230	1039	257
	0.88	1195*	352	442*	193	1015*	204

Table 6.4: Total number of groups and number of meaningful groups for each ontological aspect for each threshold. The number of total groups for a given semantic threshold again generally remains the same at different functional thresholds, except for the numbers marked with *, in the case of the maximum FT for Schlicker-MAX. See Table 6.2 for a full explanation.

6.2.2 Groups by ontology

The same trends observed for the entire groupings can also be observed if the results are separated according to the GO's three sub-ontologies. Table 6.4 shows the total number of groups for each ontology and the number of meaningful groups. As with the full grouping, the total number of groups for each ontology is generally the same for a given ST, regardless of FT, except for the maximum FT for Schlicker-MAX. The number of meaningful groups for Resnik is always a greater percentage of the total groups at minimum ST than at maximum ST, while the number of meaningful groups for Schlicker is again more affected by the rise from minimum to maximum FT.

However, the specific numbers of total and meaningful groups vary quite considerably with the different ontologies, with the groups by no means being evenly distributed across the three ontologies. In all cases, the largest number of groups (both total and meaningful) belong to the BP ontology. For the other two ontologies, there is no similar overall trend. If Resnik's similarity measure is used, MF

has significantly fewer groups than CC at minimum ST while group numbers are about the same for the two ontologies at maximum ST, both in terms of total and meaningful groups. For Schlicker, there are far more MF groups than CC groups in terms of total number of groups. In terms of meaningful groups on the other hand, the two ontologies have very similar numbers of groups, i.e. there are far more MF groups with three or fewer gene products than there are CC groups.

The consistently high proportion of BP groups is a reflection of the proportion of BP terms in the total annotations of the Eisen dataset and the proportion of BP terms in the set of distinct terms in these annotations. The exact numbers of total annotations and distinct terms are given in Table 6.5. In fact, with Schlicker's measure, the number of total groups closely reflects the proportions of distinct GO terms from each ontology found in the annotation of the dataset. Since the number of total groups is primarily dependent on the ST, this may be a strong indicator that the STs determined for Schlicker are too high to generate good group definitions.

	No of annotations	No of distinct GO terms
BP	12611	1444
CC	8567	518
MF	8815	1139
Total	29993	3101

Table 6.5: Eisen annotations by ontology. Both total annotations (distinct gene-GO term tuples) and distinct GO terms are shown. Annotations found in GO release for 04-2009.

In the grouping process, each distinct GO term in the annotations of the dataset initially receives its own group to which related GO terms are added. Then any groups whose definition is a subset of another group's definition are removed in order to avoid duplication in definitions. Clearly, far fewer groups are affected by this removal process if Schlicker's measure is used than for Resnik. In terms of definition sizes, the largest group definition for any group generated using Schlicker consists of 5 GO terms for ST93 and 4 GO terms for ST95. For groups based on Resnik's similarity measure, the largest group definitions are 141 GO terms for ST28 and 57 GO terms for ST40. In addition, the number of group definitions of size 1 lie around 77% (minST) and 86% (maxST) of total groups respectively for Schlicker, whereas for Resnik, these groups represent 19% and 26% of total groups respectively. Group definitions are not subject to the same minimum size requirement as the group content and definitions of only one GO term can clearly show the functional concept on the basis of which the genes in the group are related. However, definitions of more than one GO term carry more information and an overly large proportion of groups with single-term definitions, as found for Schlicker, may not be desirable.

Comparing Tables 6.4 and 6.5, the number of total groups for each ontology for Resnik bears little resemblance with either the frequency or the distribution of annotations across the three ontologies. Although the total number of groups for BP is much higher than the total number of groups for either of the other two ontologies, it is much lower than the number of distinct GO terms found in the Eisen dataset annotations. For CC and MF, the total number of groups are much lower than for BP. Total number of groups for MF is either much lower than total number of groups for CC, in the case of the minimum ST, or about the same, in the case of the maximum ST.

The explanation for this behaviour lies mostly with the way Resnik's method works, rather than in the distribution of the annotations across the three ontologies, as it does with the behaviour of groups for Schlicker. As discussed in Section 3.1, Resnik's measure considers the location of the common ancestor of two terms within the hierarchy but not the distance between these terms and the ancestor. For each of the three ontologies, the longest path between the root and one of the terms present in the dataset annotation lies at 15 edges. The average depths for each ontology lie at 8.26 for BP, 10.18 for CC and 6.40 for MF, i.e. MF has overall the most shallow annotations while CC has overall the deepest. CC therefore has both the least number of terms and the deepest terms, making it more likely for two terms to be further apart, i.e. have a shallower ancestor and thus lower similarity between the terms. MF on the other hand has more terms and has overall less depth than CC, so that it is more likely that any two GO terms are more closely related and that the depth of their common ancestor is less shallow compared to the overall depth of the ontology. The resulting higher semantic similarity values mean that there are fewer groups compared to the number of total possible groups (or total number of GO terms) for MF, as more terms can be grouped together in a single definition. For CC on the other hand, lower semantic similarity values may result in more groups as terms are not sufficiently related to be grouped into the same definition at higher STs.

In terms of meaningful groups for each ontology, the same trends as discussed earlier for the full set of groups apply. If Resnik's approach is used, the number of groups of insufficient size is greater at maximum ST than at minimum ST, while the number of groups of insufficient size for Schlicker is greater at maximum FT than at minimum FT, with little effect from different STs.

Group sizes by ontology

No new trends can be determined from the maximum and average group sizes for each ontology, listed in Table 6.6. As with the average sizes for the full sets of groups, the average group sizes for both all and meaningful groups are considerably larger for groups derived using Resnik's measure than for Schlicker's measure. There is also no clear trend for one ontology consistently having larger groups sizes than another. The BP ontology has the greatest number of overall maximum group sizes for a given threshold, but there is no observable pattern in the thresholds where this is the case. It is more likely that BP has the largest group sizes because it also has the greatest number of groups, making it more likely that the largest group for a threshold will be a BP group.

		BP			CC			MF		
			Average size			Average size			Average size	
ST	FT	Max. size	all groups	meaningful	Max. size	all groups	meaningful	Max.size	all groups	meaningful
0.28	0.17	177	55.17	58.88	175	28.38	42.00	172	39.01	46.88
	0.25	99	25.55	27.64	108	14.99	22.27	112	17.41	21.76
0.40	0.17	91	27.21	30.63	129	17.89	26.94	170	18.46	26.57
	0.25	48	13.58	15.75	73	11.04	16.55	108	10.69	15.98
0.28	0.26	261	76.21	80.36	195	33.85	50.28	249	53.74	63.64
	0.41	75	28.08	29.56	73	15.95	23.18	75	18.62	21.86
0.40	0.26	151	45.42	50.65	175	21.52	32.60	174	23.69	33.42
	0.41	54	15.94	17.80	71	12.04	17.77	75	12.29	17.18
0.93	0.42	214	6.49	12.72	185	8.24	14.04	170	5.01	15.61
	0.58	120	4.27	9.43	121	5.68	10.31	121	3.68	11.98
0.95	0.42	211	5.81	12.01	185	7.90	13.63	170	4.61	14.71
	0.58	88	3.88	8.95	121	5.45	9.93	121	3.45	11.41
0.93	0.66	311	8.07	15.76	191	8.24	13.98	283	5.63	17.48
	0.88	177	4.63	11.15	171	5.72	10.56	191	4.16	14.17
0.95	0.66	306	7.14	14.61	191	7.89	13.59	174	5.01	15.80
	0.88	173	4.16	10.41	171	5.45	10.13	171	3.78	13.18

Table 6.6: Maximum and average group sizes for each ontological aspect for each threshold. The average group size is calculated across both all groups and meaningful groups only.

6.2.3 Number of genes

Until now, genes have only been discussed in terms of group size but not in terms of the number of genes that are grouped at least once for each set of thresholds. As mentioned in Section 3.1.6, not all genes from a dataset will necessarily be grouped by the FuSiGroups algorithm. This is particularly true in the case of larger datasets, which have a greater likelihood of containing genes that are not functionally related to any of the other genes in the dataset.

The fact that not all genes are grouped is not a flaw of the grouping process. If traditional hierarchical clustering were used to find related genes, either based on functional similarity or expression similarity, most trees would also contain genes

that are clustered alone unless the tree is cut at a very high level. This is indeed the case if the Eisen dataset is clustered using the original expression data and the parameters described in Eisen et al. [1998]. The resulting cluster tree is not 100% identical to the tree obtained by the original authors because even minor differences and improvements in the clustering algorithm can change the result of the clustering. The major trends however can be found and if the key clusters identified by Eisen et al. are considered, they are generally found at a level of between 0.3 and 0.5. At these levels, the number of genes clustered with at least one other gene and at least three other genes, the same size as our meaningful group size, are listed in Table 6.7.

Distance in tree	No of genes for s>1	No of genes for s>3
0.3	1109	589
0.4	1848	1154
0.5	2274	1723

Table 6.7: Number of clustered genes in clusters of size greater than 1 and size greater than 3, at three levels of distance in the cluster tree. Clusters of size 1 are never included as the number of genes including those clustered alone is always equal to the number of genes in the dataset.

Table 6.8 shows the number of distinct genes in all groups for each set of thresholds, the number in meaningful groups, the percentage of the total number of genes this represents and the difference in genes between all and meaningful groups. A comparison between the number of genes grouped at least once and the number of genes clustered with other genes in Table 6.7 shows that the grouping process generally includes more genes in the solution than the clustering process. This is particularly the case at minimum FTs, which obviously allow a higher number of genes to be grouped since the genes are not required to be as closely related as with maximum FTs. In Table 6.7, at a tree height of 0.3 for example, only 589 genes are found in clusters of more than three members whereas the number of genes in meaningful groups in Table 6.8 always exceeds 1500.

The "Difference" column in Table 6.8 reveals that the number of genes found only in groups of insufficient size is generally lower if Resnik's measure is used. Additionally, the loss of genes is greater if the BMA functional similarity approach is used, compared to the MAX approach. Aside from these observations, no further trends can be detected from this data.

ST	FT	Number of genes				
		All groups		Meaningful groups		Difference
0.28	0.17	2226	90.30%	2208	89.57%	18
	0.25	1586	64.34%	1502	60.93%	84
0.40	0.17	2342	95.01%	2318	94.04%	24
	0.25	1896	76.92%	1762	71.48%	134
0.28	0.26	2416	98.01%	2413	97.89%	3
	0.41	1659	67.30%	1614	65.48%	45
0.40	0.26	2442	99.07%	2441	99.03%	1
	0.41	2049	83.12%	1972	80.00%	77
0.93	0.42	2391	97.00%	2283	92.62%	108
	0.58	2247	91.16%	1771	71.85%	476
0.95	0.42	2395	97.16%	2284	92.66%	111
	0.58	2259	91.64%	1766	71.64%	493
0.93	0.66	2411	97.81%	2340	94.93%	71
	0.88	2118	85.92%	1661	67.38%	457
0.95	0.66	2412	97.85%	2345	95.13%	67
	0.88	2136	86.65%	1666	67.59%	470

Table 6.8: Number of genes grouped at least once for all groups and for meaningful groups. Percentage of total genes in dataset (2465) and difference between all groups and meaningful groups are also shown.

6.3 Group definitions

6.3.1 Definition size

Group definitions, i.e. the GO terms associated with each group, have already been briefly mentioned, in Section 6.2.2, but it is worth considering them in a little more detail. Table 6.9 shows the maximum and average definitions sizes for all sets of thresholds. From this table, the previously mentioned difference in definition sizes between groups based on Resnik's approach and groups based on Schlicker is immediately obvious, with the average group size for Schlicker never exceeding 1.5.

For groups based on Resnik's approach, the average group definition sizes suggest that while the majority of groups may not have definition sizes close to the maximum definition size, there should be a number of groups with larger definitions. As stated above, groups with a single GO term as their definition represent 19% (minST) or 26% (maxST) of all groups for Resnik, a much smaller proportion than for groups based on Schlicker.

Group definitions for maxST are invariably smaller than group definitions for minST, since fewer GO term pairs meet the similarity criteria to be grouped together at higher STs.

ST	Max.size	Average size all groups	FT	Average size meaningful groups
0.28	141	27.34	0.17	32.86
			0.25	33.64
0.40	57	9.39	0.17	11.92
			0.25	12.35
0.28	141	27.34	0.26	32.55
			0.41	
0.40	57	9.39	0.26	11.79
			0.41	11.84
0.93	5	1.27	0.42	1.50
			0.58	
0.95	4	1.16	0.42	1.32
			0.58	1.31
0.93	5	1.27	0.66	1.50
			0.88	1.49
0.95	4	1.16	0.66	1.32
			0.88	1.31

Table 6.9: Maximum and average group definition sizes for all threshold combinations. Average group sizes are calculated for both all groups and meaningful groups only. Since group definitions are only based on the semantic similarity between GO terms, not on the functional similarity between gene products, they are only affected by changes in the ST (same maximum and total average size for all FTs at a given ST). Average definition size for meaningful groups does however vary at different FTs as group content is dependent on the FT.

For Resnik's approach, the average definition size for meaningful groups for a given ST is usually slightly higher at maxFT than at minFT. This suggests that groups that contain not too strongly related genes do not have large definitions. Since these groups lose some of their content at higher FTs and no longer count as meaningful groups, the corresponding loss of small definitions from the average definition size would lead to an average that is slightly less skewed towards smaller definitions, even with a smaller maximum definition size. The same trend is not seen in Schlicker groups, although this may be due to the overall much smaller definitions rather than the true absence of the effect.

6.3.2 Group size vs. definition size

For both semantic similarity approaches, the average definition size is slightly higher if only meaningful groups are considered. This suggests that there might be a correlation between group size and group definition size. Figure 6.1 shows the correlation between the number of gene products in a group (group size) and the number of GO terms in the group's definition (definition size), for ST28-FT17. The scatter plot shows that there is some dependence between group and definition sizes if both variables are very small (approx. group size < 25, definition size < 12). This explains the apparent correlation seen in the increase in average definition size for meaningful groups. Overall however, there is no clear relationship between group and definition size, with some of the largest groups having very small definitions, while some the larger definitions belong to relatively small groups. The data certainly does not allow any automatic conclusion on definition size based on group size or vice versa. The log version of the scatter plot shows a slightly stronger linear relationship between group and definition size. However it also makes the spread of group sizes for the smallest definition sizes more obvious.

For space reasons, the correlation between group and definition size is only shown for ST28-FT17. The correlation coefficients for all threshold combinations, shown in Table 6.10, were calculated using the Pearson correlation coefficient and R's *cor()* function. The correlation coefficient for the data represented in Figure 6.1 is 0.553. It supports the conclusion already derived from the scatter plot that there is no true relationship between group size and definition size. Overall, the correlation coefficients for groups based on Resnik's approach follow a similar trend, with some stronger and some weaker than the coefficient for ST28-FT17. Even the highest coefficient (0.7, for ST40-FT17) does not reach the level generally considered to represent a strong relationship ($r \geq 0.8$). For Schlicker's approach, the correlation

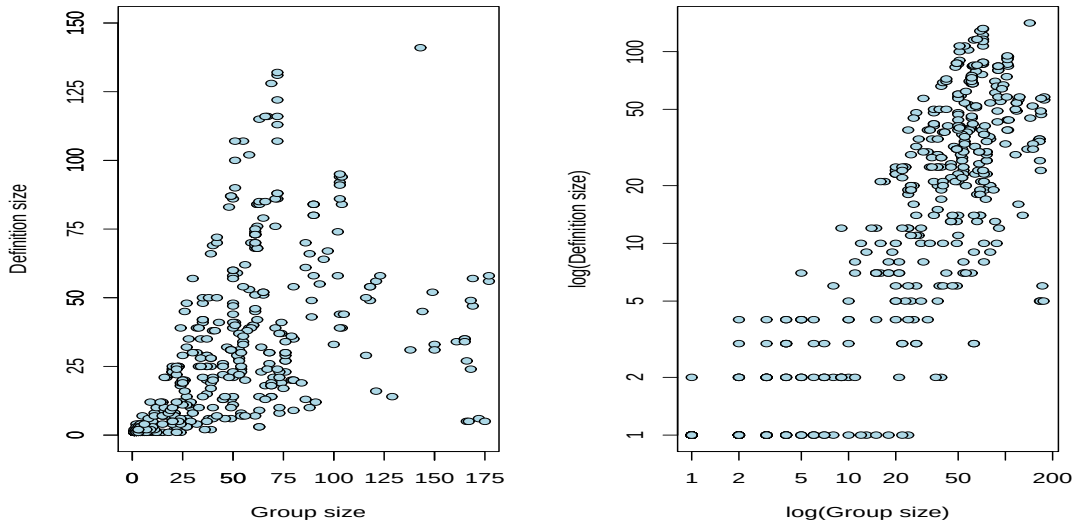

Figure 6.1: Correlation between group and definition size for ST28-FT17, for all groups. The left plot shows the direct correlations, while the right plot shows the correlation on a logarithmic scale.

coefficients are even lower than for Resnik. This is due to the very small definition sizes of these groups, which in no way reflect the range of corresponding group sizes (see Table 6.3).

6.3.3 Definitions by ontology

The trends discussed so far for all group definitions can also be found if definitions are considered by ontology (Table 6.11). Group definitions are much larger for Resnik groups than for Schlicker groups; group definitions are smaller at maxST than at minST; average definition size for meaningful groups for a given ST goes up marginally at maxFT compared to minFT for Resnik and average size for meaningful groups is slightly larger than average size for all groups, for both approaches.

In addition, with Resnik's approach, the average definitions sizes (both for all groups and also for meaningful groups only) for each ontology very roughly reflect the proportions of distinct GO terms of each ontology in the annotation. BP, which has the highest proportion of distinct GO terms, also has the largest average group definitions, while CC, with the lowest proportion of GO terms, also has the smallest average definitions and MF's definition sizes lie between the other two.

The same does not apply to maximum group sizes, where MF groups exceed BP groups at both minST and maxST, while CC always has the lowest maximum definition size of the three. Analysis of the groups with the largest definitions shows

ST	FT	r
0.28	0.17	0.55
	0.25	0.42
0.40	0.17	0.62
	0.25	0.53
0.28	0.26	0.55
	0.41	0.48
0.40	0.26	0.70
	0.41	0.51
0.93	0.42	0.17
	0.58	0.17
0.95	0.42	0.14
	0.58	0.14
0.93	0.66	0.16
	0.88	0.12
0.95	0.66	0.14
	0.88	0.10

Table 6.10: Correlation coefficients (r) for group size vs. definition size for each set of thresholds. The coefficients were calculated using Pearson correlation and R's *cor()* function.

		BP			CC			MF		
			Average size			Average size			Average size	
ST	FT	Max. size	all groups	meaningful	Max. size	all groups	meaningful	Max.size	all groups	meaningful
0.28	0.17	132	40.46	43.16	57	7.88	11.36	141	26.88	32.19
	0.25			43.88			11.63			33.81
0.40	0.17	50	12.85	14.39	28	4.30	6.10	57	8.32	11.67
	0.25			14.89			6.21			12.33
0.28	0.26	132	40.46	42.64	57	7.88	11.36	141	26.88	31.69
	0.41			42.64			11.36			31.69
0.40	0.26	50	12.85	14.24	28	4.30	6.10	57	8.32	11.41
	0.41			14.33			6.10			11.46
0.93	0.42	5	1.36	1.59	4	1.24	1.39	4	1.18	1.43
	0.58			1.59			1.39			1.44
0.95	0.42	4	1.22	1.41	4	1.15	1.25	3	1.10	1.22
	0.58			1.39			1.27			1.23
0.93	0.66	5	1.36	1.58	4	1.24	1.39	4	1.18	1.43
	0.88			1.61			1.37			1.41
0.95	0.66	4	1.22	1.40	4	1.15	1.25	3	1.10	1.23
	0.88			1.41			1.24			1.21

Table 6.11: Maximum and average group definition sizes for each ontological aspect for each threshold. The average group size is calculated across both all groups and meaningful groups only.

that at minST, there is only one MF group with a definition larger than the largest BP group. The greater maximum size of the largest MF group is therefore most likely a coincidence rather than an indicator of a trend. At maxST, there is an approximately equal representation of MF and BP among the groups with the largest definitions, which suggests that BP terms may be more susceptible to the higher ST, i.e. BP term pairs may, on average, have slightly lower semantic similarity. Neither of these trends are observable for groups based on Schlicker because of the overall smaller definitions, which mask any other potential effect.

6.3.4 Number of GO terms

In the same way as not all gene products are necessarily included in a grouping (see Table 6.8), not all GO terms that are annotated to the gene products in the dataset are used in group definitions. Table 6.12 shows how many out of the Eisen dataset's 3101 distinct GO terms are used at each set of thresholds. The GO terms that are missing in the "All groups" column represent terms for which similarity with themselves is lower than the ST for that grouping. As a result, these terms violate the maximum completeness rule of any definition they are in, even in a single-term definition and are therefore always removed.

The further loss of GO terms when only meaningful groups are considered is due to some GO terms only occurring in the definitions of groups considered as not meaningful due to the number of gene products associated with them. In this case, groups based on Schlicker's approach show a much greater loss in GO terms than groups based on Resnik. This finding is in line with the amount of actual groups lost as not meaningful for Schlicker compared to Resnik (see Table 6.2). As a result, fewer than half of the original annotations are found in meaningful groups for Schlicker, while even the worst loss for Resnik still only represents around 88% of the original terms. Although this reduction in descriptive richness may not automatically cause problems for Schlicker's groups, it is of course desirable to have as many of the annotated GO terms in the group definitions as possible.

6.3.5 Definition size vs. term depth

A final point that needs to be considered in relation to group definition size is whether there is any relationship between the number of terms in a definition and the depth of these terms. A strong correlation (positive or negative) between definition size and term depth would suggest bias in the data, with deep terms occurring

ST	FT	Number of GO terms	
		All groups	Meaningful groups
0.28	0.17	3067	2975
	0.25		2953
0.40	0.17	2976	2779
	0.25		2737
0.28	0.26	3067	2982
	0.41		2982
0.40	0.26	2976	2791
	0.41		2781
0.93	0.42	3067	1326
	0.58		1083
0.95	0.42	3058	1221
	0.58		990
0.93	0.66	3067	1363
	0.88		1044
0.95	0.66	3058	1263
	0.88		947

Table 6.12: Number of GO terms used at least once in the group definitions for all groups and for meaningful groups. The total number of GO terms in the annotation of the Eisen dataset is 3101. The total number of distinct GO terms in group definitions is less than 3101 because there are always a few GO terms whose similarity with themselves is less than the ST for that grouping.

primarily in either small definitions (negative correlation) or in large definitions (positive correlation).

ST	r
0.28	0.21
0.40	0.28
0.93	0.06
0.95	0.03

Table 6.13: Correlation coefficients (r) for definition size vs. average depth of GO terms in the definition, for each semantic threshold. The coefficients were calculated using Pearson correlation and R's *cor()* function. Correlation coefficients are independent of FT as the group definitions only depend on ST.

Table 6.13 shows the correlation coefficients (calculated as before in R, using Pearson's correlation) for definition size and average depth of the terms in the definition. The average depth was calculated using the maximum depth for each term (longest path between the term and the ontology root) in the definition. As a reminder, each term in the GO can be related to the root via multiple paths, which may traverse different numbers of nodes. Unless otherwise stated, any reference to "distance from root" of a GO term is to the maximum distance. For all the terms annotated to the Eisen dataset, the difference between minimum and maximum distance to the root ranges from 0 (about 39% of terms) to 11 (one case). The average difference lies at 1.98.

Based on the coefficients found for each ST, there is no obvious correlation between definition size and term depth and therefore no bias in the way the GO terms are grouped into definitions. In conjunction with the correlation coefficients found for group size vs. definition size (Table 6.10), the conclusion is that the underlying structure of the GO does not directly influence the semantic similarity between GO terms, the resulting functional similarity between gene products and the groups which are created using the two types of measures.

6.4 Summary

In this chapter, a number of overall trends of grouping results for different thresholds were considered. One recurring feature that stands out is the effect of Schlicker's semantic thresholds on the grouping results. The high thresholds and small difference between minimum and maximum ST led to a much larger number of groups than those obtained using Resnik's measure, much smaller average group sizes, a much greater proportion of groups of insufficient size and very small group definitions.

All of these elements bring into question the suitability of these semantic thresholds. As briefly discussed in Section 6.1, lowering the minimum ST for Schlicker was considered but initial tests showed little promise of improvement.

Although Schlicker's measure objectively addresses a drawback in Resnik's measure, it performed less well than the older measure in the present context. While using a different true-positive/true-negative dataset might have generated a different and better set of thresholds, this was not feasible within the scope of this project as it would have required a lot of time and very high levels of expert knowledge of all areas of molecular biology covered by the GO. The analysis of grouping results from here will therefore be limited to Resnik's approach.

While this chapter focussed on the groups generated by the FuSiGroups algorithm from a very high-level perspective, the next two chapters will focus on the actual content and definitions of some of the groups. In addition to the full Eisen dataset, subsets of the data will be analysed in detail.

Chapter 7

The complete Eisen dataset

Until now, the results generated by the FuSiGroups algorithm have only been analysed in very general terms such as the number of groups obtained for a given set of thresholds, group sizes and number of genes grouped. In this and the next chapter, the definition and content of groups will be analysed in order to determine whether the FuSiGroups algorithm does indeed meet its target functionality of grouping together gene products based on meaningful functional relationships, providing an objective view of such complex biological data containing valuable novel insights. Chapter 7 focusses on the full Eisen dataset, while Chapter 8 will provide detailed investigations into several smaller, less noisy datasets to address a number of specific questions.

At the end of Chapter 6, it was concluded that the semantic thresholds determined for Schlicker's approach are less suitable for use with FuSiGroups than those determined for Resnik. For this reason, all analysis in this chapter uses groups based on Resnik's semantic similarity approach.

In addition, in order to avoid repetition, it would be helpful to select only one combination of the ST and FT parameters on which to perform a more detailed analysis of groupings. It was shown previously that for Resnik, the BMA functional similarity approach performs better than the MAX functional similarity approach (Section 4.1, Figure 4.4). Considering this finding, the grouping results for the BMA approach will be used in this analysis. The main analysis will be performed on the grouping results using minimum ST and FT since these correspond to the highest accuracies in their respective datasets. Comparisons with results for maximum ST and FT will be made as necessary.

Unless stated otherwise, all groups have been created using the parameters listed in Table 7.1.

Variable	Value
Semantic similarity	Resnik
Functional similarity	BMA
Annotations	all annotations
Ancestor selection	MICA
Semantic threshold	0.28
Functional threshold	0.17

Table 7.1: FuSiGroups parameters for groups analysed in Chapter 7.

7.1 Largest groups and most common aspects

There are two angles from which a closer analysis of grouping results could be started, namely the largest groups or the most common functional aspects represented by the groups. In a smaller dataset, the most common functional aspects should be the most sensible starting point, as they are most likely to reveal immediate information about the groups. In a dataset the size of the Eisen dataset on the other hand, this does not necessarily hold true, as the most common functional aspects may be too generic to contain any useful information.

In Chapter 6, it was established that the grouping result for the parameters in Table 7.1 consists of 481 groups, 397 of which contain at least 4 gene products (Table 6.2) and that the largest group contains 177 gene products (Table 6.3). Figure 7.1 shows the distribution of group sizes for meaningful groups[1]. As they are not going to be considered in the analysis, the smaller group sizes are not included in the histogram in order to keep the size of the histogram's Y axis as readable as possible. The frequencies for groups of size 1, 2 and 3 are 35, 31 and 18, respectively.

Although the study of the distribution of group sizes would seem to be more appropriate in the previous chapter, this information was not considered until now as it was felt inappropriate and overly repetitive to perform this analysis for all sets of thresholds. It is included here in order to provide context for the largest groups, such as what fraction of the full set of groups they represent and how their sizes compare to the majority of the groups. From the histogram, it is clear that the majority of groups (almost 85% of meaningful groups) have sizes in the interval $[4, 80]$, although there are a couple of spikes in the number of groups at size 90 and 103, as well as a scattering of groups of sizes greater than 105 gene products.

Tables 7.2 and 7.3 show the most common group names, representing the most common functional aspects, and the largest groups, respectively. As a reminder,

[1] As a reminder, "meaningful groups" have previously been defined as groups which contain at least 4 genes.

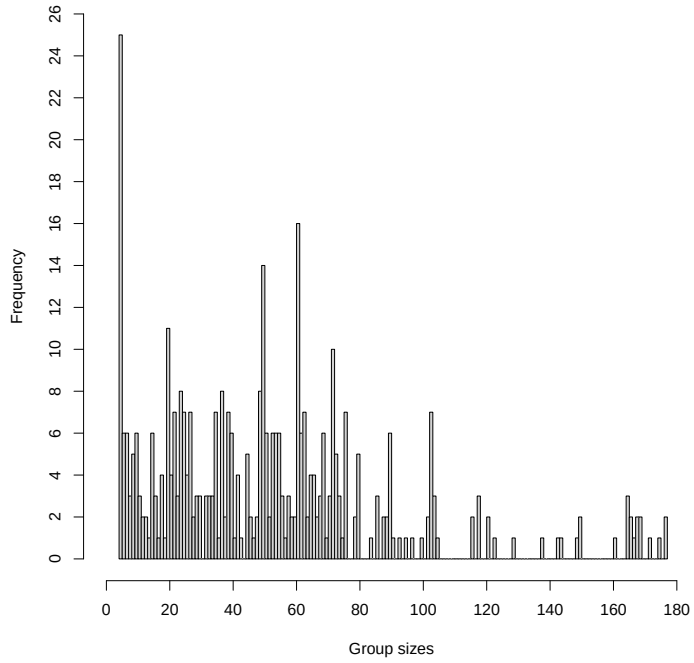

Figure 7.1: Distribution of group sizes for ST28-FT17, for meaningful groups. Groups of smaller size are not included in order to reduce the size of the histogram's Y axis. The frequencies for groups of size 1, 2 and 3 are 35, 31 and 18, respectively.

Name	Ontology	No. of groups	Group size		
			Maximum	Average	Minimum
biopolymer modification *(GO:0043412)*	BP	16	58	46.94	34
catabolic process *(GO:0009056)*	BP	15	62	61.00	59
organic acid metabolic process *(GO:0006082)*	BP	11	72	62.55	48
cellular localization *(GO:0051641)*	BP	11	104	97.09	64
endomembrane system *(GO:0012505)*	CC	8	80	73.63	69
nucleobase, nucleoside and nucleotide metabolic process *(GO:0055086)*	BP	7	38	35.29	34
cell cycle *(GO:0007049)*	BP	7	66	63.29	62
DNA metabolic process *(GO:0006259)*	BP	7	89	73.57	55
mitochondrial part *(GO:0044429)*	CC	7	88	75.43	66
response to stress *(GO:0006950)*	BP	6	90	89.67	88
nitrogen compound metabolic process *(GO:0006807)*	BP	6	72	71.67	71
carbohydrate metabolic process *(GO:0005975)*	BP	6	39	24.17	20
reproduction *(GO:0000003)*	BP	6	49	37.83	25
translation *(GO:0006412)*	BP	6	167	165.67	165
macromolecular complex subunit organization *(GO:0043933)*	BP	6	65	60.00	43
cytoskeleton *(GO:0005856)*	CC	6	55	53.50	52
negative regulation of biological process *(GO:0048519)*	BP	5	68	63.60	50
lipid metabolic process *(GO:0006629)*	BP	5	57	50.40	35

Table 7.2: Most common group names for ST28-FT17. Group names occuring a minimum of 5 times are shown, representing at total of 29.31% of all groups (or 35.52% of meaningful groups).

a group's name is the lowest common ancestor of all the GO terms in the group definition, where lowest refers to the maximum distance of the term from the root. If more than one ancestor term has the same maximum distance from the root, the first term in the list of equally deep ancestors is used. A comparison of the two tables shows that the only overlap between them is for the group name "translation (GO:0006412)". All groups with this name are found among the largest groups.

There are no molecular function groups in Table 7.2. This is not surprising as the functions of a set of proteins are likely to be more diverse than the set of processes these proteins are involved in, i.e. a number of different molecular functions make up a single biological process. A set of proteins that are part of the same biological process may therefore be subdivided into several distinctly named groups based on their functions because these functions differ enough for the overall functional similarity to be below the FT. The related BP- and CC-based groups however have the same or very similar names as the proteins are all part of the same process and act in the same cell part.

The explanation for the fact that only three out of the 18 most used group names are cellular component groups is slightly different. Although gene products grouped together because they are functionally similar are highly likely to be found in the same location, there are far fewer CC GO terms than BP terms, both in the number of distinct GO terms and in terms of annotations (see Table 6.5). It therefore follows that the majority of commonly used group names are of type BP rather than CC.

Overall, names in Table 7.2 reflect general cellular processes, such as metabolism (e.g. organic acid metabolic process, nitrogen compound metabolic process etc) and cell cycle. This is unsurprising considering the nature of the Eisen dataset. The genes in the dataset were selected based on the availability of functional annotations in 1998 [Eisen et al., 1998], not on the basis of any biological properties and therefore, they cover all aspects of the yeast genome. The experimental conditions on which Eisen et al. based their cluster analysis highlight genes involved in the affected processes but the full dataset is entirely unfiltered.

This effect is also observable in Table 7.3, among the largest groups obtained for ST28-FT17. The largest groups cover broad aspects of cell function, such as transcription and translation, and high-level locations such as cytosol and ribosome. All of these concepts cover a large number of genes, thus resulting in the largest groups.

This confirms the earlier assertion that considering the most common group names or largest groups may not be a useful approach for analysing a large dataset. In a smaller dataset, in which the genes may be related to a given theme, e.g. a

common pathway or set of pathways, this approach may reveal useful information. In a dataset like the Eisen dataset on the other hand, a more targeted approach, i.e. an analysis approach with a specific gene or function set in mind, would be more appropriate. This is not necessarily a drawback when approaching the data to address discrete biological hypotheses. Eisen et al. clustered genes from a set of gene expression studies involving diauxic shift, cell cycle, sporulation and temperature shock. Suitable starting points for the analysis of this dataset could therefore be genes of interest in one of these processes or functional aspects of these processes. This option will be addressed below.

An interesting observation is that many of the groups in Table 7.3 have the same name. There are for example four groups with the name "transcription, DNA-dependent (GO:0006351)" (marked with $*$ in Table 7.3), including the two largest groups. Between them, they contain 209 distinct genes of which 88 (42%) are found in all groups. 32 (15%) genes are unique to one of the groups, while a further 74 genes are found in three out of the four groups. The groups' definitions also have some overlap, although it is not quite as pronounced. 14 out of 61 distinct GO terms are present in all definitions while only 3 terms are unique to one definition.

This level of overlap is even more pronounced in the six groups with the name "translation" (marked with $\mp$), which have 168 distinct genes between them and all six groups contain 165 of these. Of the three genes not found in all groups, one is in two groups and two are unique to one group. From a group definition point of view, the situation is slightly different. Only one of 46 distinct GO terms is common to all six group definitions; this is GO:0006412, i.e. translation and 8 terms are unique to one of the definitions.

This trend of high levels of overlaps can be observed in any set of groups with the same group name. More broadly, most of the group names in Table 7.3 fall into two categories, namely transcription-related groups (14 groups) and translation-related groups (13 groups). Three groups (1474, 1478 and 1070) do not fit into either of these categories. All groups in either of the two categories have a considerable overlap in their gene content. Although no gene in the transcription-related groups is present in all 14 groups, 20 genes out of a total of 364 distinct genes are present in 13 groups and 154 genes (42%) are present in at least 8 groups. Only 75 genes are present in just one group.

The overlap is even stronger for the translation-related groups. Here, 26 genes out of a total of 288 occur in all 13 groups, while 164 genes (57%) are found in 7 or more groups, whereas there are only 10 genes that are unique to a single group. In the translation category, there are two sub-categories with even stronger overlap:

Group ID	Group name	Ontology	Group size
1193 ∗	transcription, DNA-dependent *(GO:0006351)*	BP	177
1350 ∗	transcription, DNA-dependent *(GO:0006351)*	BP	177
1367	ribosome *(GO:0005840)*	CC	175
1196	structural molecule activity *(GO:0005198)*	MF	172
1220	regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process *(GO:0019219)*	BP	169
1357 ±	nucleoplasm *(GO:0005654)*	CC	169
1365	regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolic process *(GO:0019219)*	BP	168
1391	transcription regulator activity *(GO:0030528)*	MF	168
1042 ∓	translation *(GO:0006412)*	BP	167
1036 ∓	translation *(GO:0006412)*	BP	166
1041 ∓	translation *(GO:0006412)*	BP	166
1095 ∓	translation *(GO:0006412)*	BP	165
1373 ∓	translation *(GO:0006412)*	BP	165
1375 ∓	translation *(GO:0006412)*	BP	165
1083 ∗	transcription, DNA-dependent *(GO:0006351)*	BP	161
1460 †	DNA binding *(GO:0003677)*	MF	150
1463 †	DNA binding *(GO:0003677)*	MF	150
1014 ±	nucleoplasm *(GO:0005654)*	CC	149
1073 ±	nucleoplasm *(GO:0005654)*	CC	144
1474	transporter activity *(GO:0005215)*	MF	143
1232 ±	nucleoplasm *(GO:0005654)*	CC	138
1371	cytosol *(GO:0005829)*	CC	129
1478 ‡	protein binding *(GO:0005515)*	MF	123
1070 ‡	protein binding *(GO:0005515)*	MF	121
1085 ∗	transcription, DNA-dependent *(GO:0006351)*	BP	121
1069 ◇	RNA processing *(GO:0006396)*	BP	118
1142 ◇	RNA processing *(GO:0006396)*	BP	118
1348 ◇	RNA processing *(GO:0006396)*	BP	118
1156	ribonucleoprotein complex biogenesis *(GO:0022613)*	BP	116
1290	chromosome *(GO:0005694)*	CC	116

Table 7.3: Largest groups for ST28-FT17. A cut-off of s $\geq$ 116 was chosen as there is a clearly visible gap in Figure 7.1 between this and the next-lowest group size. Groups with the same name are marked with a symbol for easier identification.

the 6 groups with name "translation" as well as groups 1367, 1196 and 1371 all share 113 of their 195 distinct genes, with only 7 genes unique to a single group. The remaining four groups (1069, 1142, 1348 and 1156) share 111 of their 123 genes and only 5 genes are found in only one group.

These findings suggest that the FuSiGroups algorithm may not be rigorous enough in avoiding duplication, as there is clearly a considerable level of overlap between groups, both in terms of content and definitions. In fact, the only step of the algorithm that really addresses duplication is the removal of groups whose definition is a subset of another group's definition. No similar step is applied to the group content. This is because the algorithm was originally designed based on the assumption that thresholds (semantic and functional) would be sufficient to create fairly discrete groups. Overlap in content was also expected, as related gene products are often related on multiple features, either in the same sub-ontology or in different ontologies, e.g. two gene products that are related based on their function are likely to also be related based on their location, or due to the process they take part in. It is therefore reasonable to expect a number of groups with roughly the same gene products but different definitions. The fact that there is also a lot of overlap among group definitions suggests that a more stringent grouping process may be required.

In order to visualise the extent of the overlaps, a matrix of groups against genes and one of groups against GO terms were created using Microsoft Excel. A sample screen shot of the groups/gene matrix is shown in Figure 7.2. Unfortunately, the matrix is too large to reproduce here. The screen shot does however demonstrate the extent of the overlap even on a small section of the matrix.

7.2 Supergroups

In order to address this overlap issue, an algorithm for creating *supergroups* was designed.

Definition 9. *Supergroup - a group that is created through the merging of two or more groups. A supergroup is not subject to the maximum-completeness rule.*

This algorithm merges groups with a high level of overlap into supergroups. This may lead to supergroups that violate the original maximum-completeness rule for group definitions or group content, as the GO terms or genes in the supergroup may no longer all have the required level of similarity with each other. In cases where a number of groups have a suitable level of overlap, this compromise was however deemed acceptable in order to reduce excessive overlap. A suitable level of overlap

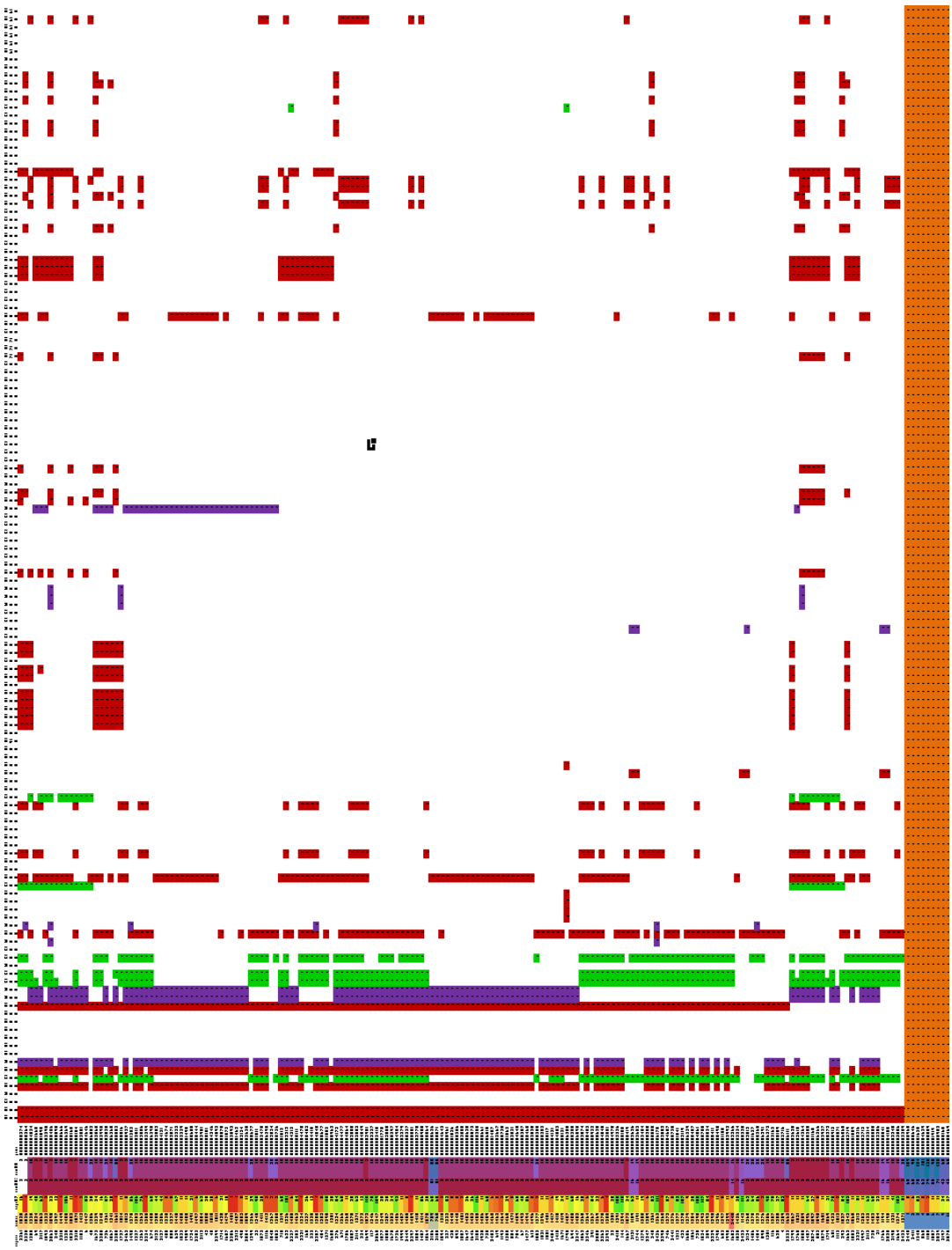

Figure 7.2: Screenshot of group alignment matrix for ST28-FT17. Note that the image has been rotated 90° anti-clockwise. The image shows 30% of the total width of the matrix and 8% of its height, at a zoom setting of 25%. The top three rows of the matrix list each group's ontology, size and ID number. The first five columns list the location of each gene in different hierarchical trees and the ID numbers of different clusters associated with these trees (see later). The actual gene IDs (SGD IDs) are in column 6. The matrix is colour-coded according to the GO ontologies for ease of interpretation: BP groups are red, CC green and MF purple. The solid orange rows represent genes that are not found in any groups.

was defined as a cosine similarity equal to or greater than 0.5 for both definition and content. Cosine similarity was discussed in the context of the phenotype dataset, in Section 3.2.1.

The similarity level of 0.5 was chosen by visual analysis of pairwise definition and content similarities between all pairs of groups that have any overlap in both categories (2535 distinct pairs). Most pairs of groups show either very high levels of overlap in both categories, or very low levels of overlap. Only a few groups have a high level of overlap in one category but not in the other, with a notable prevalence of pairs with high content overlap but low definition overlap (347 pairs) compared to pairs with high definition overlap but low content overlap (78 pairs). The option of using two distinct similarity levels for definition and content overlap was considered, but due to time constraints and the ad hoc nature of this additional algorithm, it was decided to proceed with a single threshold for both types of overlap.

The merging algorithm was designed as follows: first the overlap between all pairs of groups is calculated, both for their definitions and their content. Then, each group is matched with all the groups with which it has the required level of overlap. Immediately merging groups at this point would of course inevitably lead to duplicate supergroups, as a group A, which overlaps with another group B, would become a supergroup, but B, overlapping with A, would become a distinct, yet identical supergroup. For this reason, the algorithm first checks every set of matched groups against all other sets and removes duplicates, so that each set of identical groups is only merged into a supergroup once. Sets of matched groups are also checked to ensure that they are not subsets of others, and subsets are removed. Finally, prior to merging, the algorithm checks that the new supergroup would have a group content of at least four gene products in order to avoid generating non-meaningful supergroups.

Initial tests indicated that the supergroups had almost as much overlap in terms of their content as their original constituent groups. For this reason, the checking of sets of matched groups against each other was extended to consider the level of overlap between sets, and further merge closely related matched sets. The level of overlap was this time set to 0.8. Specifically, this meant that if there are two sets of related groups, $\{A, B, C, D, E\}$ and $\{A, B, C, E, F\}$, respectively, which have an overlap of 0.8 or more, they are merged into a single set.

initialise list *allGroups*
initialise list *matchedSets*
initialise list *mergedGroups*
FoR ALL $G \in allGroups$ DO
  FOR ALL $T \in allGroups - G$ DO
    calculate $overlap_def(G,T)$
    calculate $overlap_cont(G,T)$
    IF $overlap_{def}(G,T) \geq 0.5$ && $overlap_{cont}(G,T) \geq 0.5$ THEN
      add T to $list(groupsthatoverlapwithG)$
    END IF
  END FOR
  add $list(groupsthatoverlapwithG)$ to *matchedSets*
END FOR
FOR ALL $set_G \in matchedSets$ DO
  FOR ALL $set_T \in matchedSets - set_G$ DO
    IF$set_G == set_T \parallel set_T \subset set_G \parallel (set_G \cap set_T) \geq 0.8$ THEN
      FOR ALL $t \in set_T$ DO
        IF $t \notin set_G$ THEN
          add t to set_G
        END IF
        remove set_T from *matchedSets*
      END FOR
    END IF
  END FOR
END FOR
FOR ALL $set_G \in matchedSets$ DO
  merge all groups in set_G into $supergroup_G$
  IF $supergroup_Gcont \geq 4$ THEN
    add $supergroup_G$ to *mergedGroups*
  END IF
END FOR
RETURN *mergedGroups*

Table 7.4: Pseudocode for the supergroups algorithm

7.2.1 Pseudocode

7.2.2 Merging results

Of the 481 groups originally obtained for the parameters used in this chapter, 244 groups were merged into 54 supergroups, leaving 237 unmerged original groups. Out of the 244 merged groups the majority were merged into a supergroup once, with only 10 groups merged twice and no groups more than twice. The supergroups range in size from 23 to 235 gene products and in definition size from 4 to 161 GO terms.

The supergroups and unmerged original groups were visualised in a new colour-

coded matrix, similar to the one in Figure 7.2. While this visualisation showed that the supergroups had clearly alleviated the degree of overlap between groups, there still remained some overlap between supergroups, as well as between supergroups and unmerged original groups. Studying the cosine similarity between the original groups that were merged into the supergroups that have the most extreme levels of overlap showed that in almost all cases, those original groups that were not merged had very low levels of definition similarity but high levels of content similarity.

If more time had been available, further refinements of the merging algorithm, such as more extensive testing of different similarity levels, would have been useful. It was however decided that the improvement in overlap levels obtained from the algorithm in its present state was sufficient to proceed with further analysis and evaluation of the FuSiGroups algorithm.

7.3 Grouping vs. clustering

7.3.1 Expression clustering

The Eisen dataset was originally put together to test the use of cluster analysis in the discovery of genome-wide expression patterns. It would therefore be interesting to compare the clusters of genes with similar expression profiles to the groups of genes with high functional similarity, especially as correlation between high gene expression similarity and high functional similarity has been demonstrated in a number of studies [Wang et al., 2004; Sevilla et al., 2005].

In their analysis, Eisen et al. identified ten strong clusters of genes with very similar expression profiles. The paper only lists the content of nine of these (133 genes), the tenth cluster being too large (126 genes) to be displayed in full. Although this is of course the most interesting information in terms of the purpose of that paper, the present analysis would benefit from the complete cluster tree. The full clustering is available as supplementary materials to the paper but only in image (GIF) format and is therefore not machine-analysable. For this reason, it was decided to recalculate the clustering using the same parameters as those documented in the original Eisen paper.

There are three elements involved in the calculation of a cluster analysis, namely the choice of similarity measure, choice of clustering method and choice of linkage method. Eisen et al. used a variation of the Pearson correlation coefficient called uncentered Pearson's correlation to quantify the similarity between each pair of genes. Uncentered Pearson's correlation is the same as standard Pearson's cor-

relation, except that the mean of the observations for each gene is assumed to be 0. The clustering algorithm used by the authors was an agglomerative hierarchical clustering algorithm.

Linkage method

Unfortunately, there is some confusion in the literature as to which linkage method was used. In the Eisen paper itself, the authors state that they used an algorithm "[...] based closely on the average-linkage method of Sokal and Michener [...]" but then proceed to describe an approach that is closer to centroid linkage. The cited source, Sokal and Michener [1958], covers a number of linkage methods, including average and centroid linkage. It is commonly cited as the first description of *UPGMA* (Unweighted Pair-Group Method using arithmatic Averages) [Lance and Williams, 1967; Sneath and Sokal, 1973]. It is harder to find a definitive attribution for a first description of *UPGMC* (Unweighted Pair-Group Method using Centroids), although Lance and Williams [1967] do indeed ascribe it to Sokal and Michener [1958]. Sneath and Sokal [1973] only cite the 1958 paper for UPGMA, but do not provide a source for UPGMC.

More recently, at least two much-quoted reviews contradict each other with respect to the linkage method used by Eisen et al.. Jiang et al. [2004] state that "Eisen et al. [20] applied an agglomerative algorithm called UPGMA (Unweighted Pair Group Method with Arithmetic Mean) [...]". D'haeseleer [2005], who refers to the former for a survey of clustering methods used specifically with gene expression, states on the other hand that "Eisen *et al.*[5] applied hierarchical clustering (using uncentered correlation distance and centroid linkage)". These two contradictory statements added to the difficulty of establishing definitively which type of linkage was used to create the clusterings in the Eisen paper.

Clustering process

It was therefore decided to generate hierarchical clusterings for both average and centroid linkage. R's *hclust()* function [R Development Core Team, 2010] was used for this purpose. Although Eisen et al. developed the tool package of Cluster and Treeview, R was chosen as it provides a number of useful functionalities such as the cutting of dendrograms at different levels and exporting of the entire clustering by location of each gene in the tree and cluster. R is also used for all other statistical analyses in this work, so no adaptation of data formats is required. In addition, Cluster has gone through several revisions since the publication of the Eisen

data and only an exactly identical implementation of the clustering algorithm would produce the exact same results as in the paper, including the same tree ordering. Therefore any implementation of an agglomerative hierarchical clustering algorithm is appropriate here.

As R's *cor()* function [R Development Core Team, 2010] only implements standard Pearson's correlation but not the uncentered Pearson's correlation, the mean of the observations for each gene was set to 0 using R's *scale()* function [R Development Core Team, 2010]. A comparison of the correlation matrices obtained for the dataset with and without this transformation does however show that the difference in correlation is minimal, with the largest difference in the order of $2.2E^{-16}$, so the transformation may not have been essential. *hclust()* operates on a distance rather than a similarity matrix, so before passing the correlation matrix to the function, Pearson's correlation was transformed into Pearson's distance using $distance = 1 - correlation$.

When comparing the two dendrograms obtained for average and centroid-based clustering, it is immediately clear that the tree for centroid linkage is much "messier" than the tree for average linkage. This is due to a known issue in R: while monotone linkage methods such as average, single and complete linkage are guaranteed to produce dendrograms without crossing branches, this safe-guard does not exist for linkage approaches like centroid. In small datasets, this is generally not an issue but in a tree with 2465 nodes, this can result in an unreadable tree. When zooming into individual branches of the centroid-based tree, it was impossible to identify any but the nearest neighbours of a gene as the branches quickly become indistinguishable. In addition, the *cutree()* function, used to cut a dendrogram into a specified number of clusters or at a given height, was unable to recognise the *hclust()* object generated for centroid linkage as being in the appropriate format.

Cluster extraction

A clustering tree can be cut at different heights in order to give clusters of different size and strength. The height of the tree reflects the distance between the objects in the tree and varies for different clustering methods. The height of the tree for average clustering ranges from 0 to about 1.1 while the height for centroid clustering ranges from 0 to just over 0.5.

A particularly difficult question in all hierarchical cluster analysis is the choice of the right level at which to cut the tree as any dataset, no matter how random, will generate clustering solutions. There are no hard and fast rules on how to determine

the right tree height. In fact, the strongest clusters generally belong to a range of heights, with some clusters at an overall greater height being stronger than other clusters at a lower height. The Eisen paper does not list a specific tree height to which the ten strongest clusters correspond. In the recreated clustering of the dataset, the clusters identified by Eisen et al. range from $h < 0.2$ to $h = 0.6$ for average linkage and in the range of $0.1 < h < 0.3$ for centroid linkage. Cluster membership and tree height are harder to establish for the centroid-based tree as it is impossible to determine whether genes that are not adjacent in the tree might still be in the same cluster.

The usual approach for determining which are the best clusters is a mixture of statistical analysis, e.g. p-values for each cluster using bootstrapping, external validation against an existing classification, if one is available, and experience. Figure 7.3 shows the dendrogram for the Eisen dataset, with all clusters that have an approximately unbiased p-value (AU) greater than or equal to 95% highlighted in the red boxes. The AU values were computed by multi-scale bootstrap resampling, using R's *pvclust()* function [Suzuki and Shimodaira, 2009]. Clusters with an AU value of 95% or more are considered to be significant by both the *pvclust()* documentation and the Bioconductor manual [Girke, 2010] and therefore this assumption was also used here.

When the clusters with an AU value of 95% or more were compared for the two clusterings, 258 such clusters were identified for average linkage, compared to 157 for centroid linkage. In each case, roughly 68% of these clusters contained only two genes. Few clusters exceeded 3 to 5 genes, with the largest cluster for average containing 10 genes. Two centroid-derived clusters were larger than this largest cluster, at 12 and 19 genes respectively. Only Eisen cluster "H"[2] (8 histone genes) was completely identified in both clusterings. All other Eisen clusters were only present in part in the set of most significant clusters.

In the light of these findings, with neither type of clustering clearly closer to the clustering presented in the Eisen paper, it was decided to base further analysis on the average-linkage clustering. Although the description of the linkage process in the Eisen paper suggests that the authors most likely used centroid linkage, the difficulties with processing the centroid linkage dendrogram in R and the impossibility of extracting the full clustering at different tree heights further support this decision. Any clusterings referred to hereafter were therefore performed using average linkage.

[2]The ten groups identified in Eisen et al. [1998] are referred to in this work by the letter which identifies them in Figure 2 of the original paper.

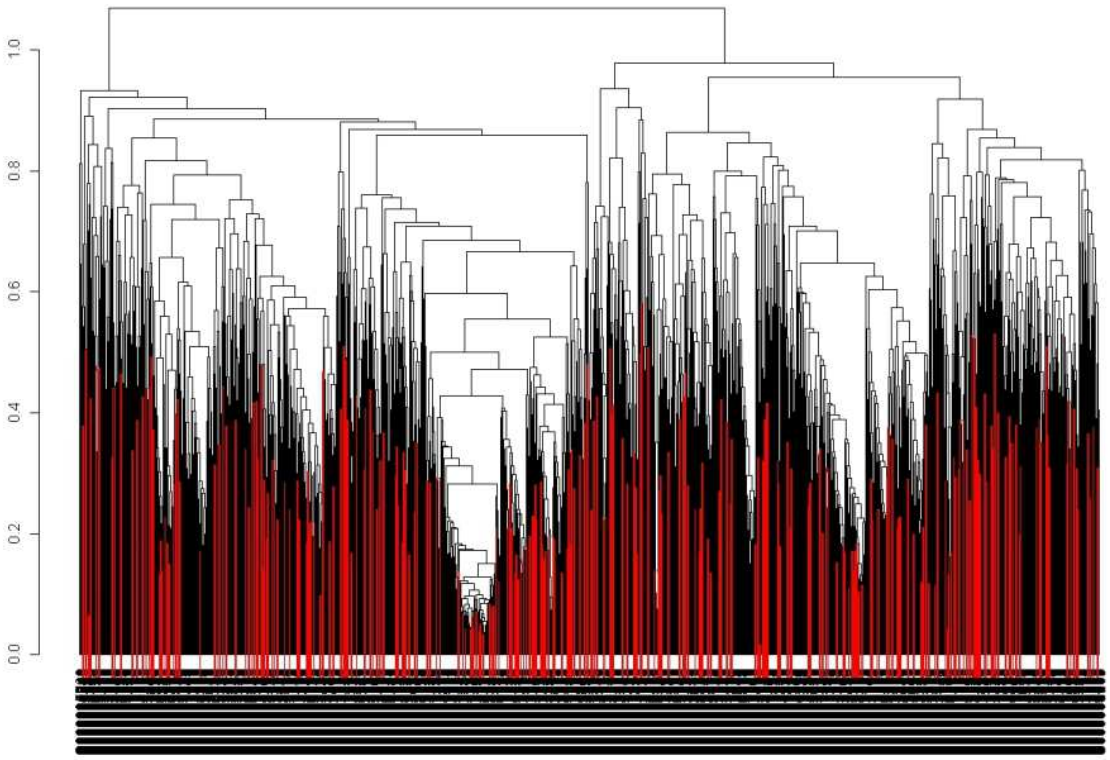

Figure 7.3: Dendrogram of clustered Eisen dataset, using average linkage. The red boxes represent the clusters with an AU value of 95% or greater.

7.3.2 Comparing expression and semantic clustering

The absence of a single "right" solution in cluster analysis poses a problem in the comparison of expression clusters and functional groups. In order to make an appropriate comparison, it would be necessary to have clusters based on a distance measure that corresponds to the FT used to generate the groups. Since the FuSiGroups algorithm is based on a completely different concept than hierarchical clustering, quality measures such as F-measure or mutual information are unsuitable as they require both datasets to be of the same type. For this reason, it was decided to insert an additional step in the analysis and cluster the gene products in the Eisen dataset using their functional distance as the distance measure. Functional distance (FD) is defined as the opposite of functional similarity, i.e. $FD = 1 - functional\ similarity$ [Couto et al., 2003]. This should then allow the identification of the expression tree height that most closely corresponds to FT = 0.17 ($FD = 1 - 0.17 = 0.83$).

Although this second clustering will be based on functional similarity, it will be referred to hereafter as semantic clustering rather than functional clustering in order to make a clear distinction from expression clustering, as this is also used to identify clusters of functionally similar genes [Eisen et al., 1998], albeit based on a different

similarity measure. The semantic clustering was performed with R's *hclust()* function and average linkage was again used. The distance matrix of functional distances that is passed to *hclust()* was generated from FuSiGroups and read into R rather than being generated from experimental data, as was the case for the expression dataset. As there is no data preceding the distance matrix for semantic clustering, it is not possible to perform multi-scale boostrapping using *pvclust()* as this function requires a dataset of observations, such as genes and samples, rather than a distance matrix.

Comparative approach

First, it was considered using a set of commonly used external quality measures (purity, F-measure, normalised entropy and mutual information) [Handl et al., 2005; Jakonienė et al., 2006] to evaluate how well semantic clusters compare to expression clusters and identify the expression threshold that most closely matches the semantic clusters for threshold 0.83. The thresholds for expression clustering ranged from 0 to 1.1 (the highest level of the tree exceeds 1, see Figure 7.3), with increments of 0.1. The thresholds for semantic clustering ranged from 0 to 1, also at increments of 0.1. In addition, the thresholds 0.75 and 0.83, corresponding to minFT and maxFT for Resnik, were used. The expression clusters were used as classes for the evaluation of the semantic clusters. The measures were also calculated for the reverse scenario, with the semantic clusters as classes for the expression clusters.

Of the four measures, purity, F-measure and mutual information should be maximised, while normalised entropy should be minimised. A good result in all four categories indicates an excellent match between classes and clusters. If the expression clusters are used as classes, the matches are near perfect at very low thresholds (0-0.3) in both datasets. This is unsurprising as there are very few clustered genes at these levels. The majority of genes are on their own or, at most, clustered in pairs. The match between semantic and expression clusters is therefore almost perfect. The same occurs if the semantic clusters are used as classes for the expression clusters.

Up to a semantic clustering threshold of 0.5, purity has a near-perfect score with any level of expression clusters, if these are the classes. This is because purity reflects the average precision of the clusters with respect to their best matching classes. For Resnik-BMA, very few gene pairs have a functional similarity greater than 0.5 (see Figure 5.10), so up to a functional distance of 0.5, very few genes are clustered together. Each cluster is therefore going to very closely match a cluster in

the expression set. In the reverse situation, with the semantic clusters as classes, the same effect occurs for the F-measure, which measures the accuracy of classes with respect to their best matching groups. Similar deceptively good scores are obtained for different measures at very low thresholds, as well as the maximum thresholds for each approach, where all genes are clustered together into a single cluster, which therefore represents a perfect match for at least one of the measure. This is due to the limitations of the measures. At intermediate thresholds for both sets of clusters, the clustered genes vary widely, thus giving fairly poor scores.

As there is clearly no tree height in the expression dendrogram that gives clusters that match semantic clusters at a threshold of 0.83 particularly well, it was decided to choose the expression threshold at which the number of clusters and the largest cluster sizes were most similar to the semantic clusters. At a threshold of 0.83, the semantic dendrogram consists of 80 clusters, of which the largest contains 410 genes. At the next lowest threshold, 0.8, this rises to 137 clusters with a maximum size of 223. The closest match in numbers and size of clusters in the expression dataset is at threshold 0.7, where there are 119 clusters, with a maximum size of 394 genes. For comparison, at a threshold of 0.6, there are 269 clusters with a maximum size of 354 and at 0.8, there are 37 clusters with a maximum size of 559 genes. Using the criteria of cluster numbers and size, clusters at level 0.7 are clearly the closest match to semantic clusters at 0.83.

Cluster matching

In order to compare the semantic and expression clusters, the locations of each gene in the respective dendrograms and the corresponding cluster IDs were aligned for expression clusters at threshold 0.7 and semantic clusters at threshold of 0.83. The alignments were included with the coloured matrix of genes and groups shown in Figure 7.2. They are the five columns that were previously undescribed. The first column in the figure contains the location of each gene in the expression dendrogram, the second column the location in the semantic dendrogram. The third column holds the expression cluster IDs, the fourth column the semantic cluster IDs and the fifth column the semantic cluster IDs for threshold 0.8. This last clustering was included to compare the change in cluster numbers between the two thresholds.

At a very high level, it is immediately clear that there is very little consistent overlap between the two types of clusterings. The genes in the largest expression clusters are spread out all over the semantic dendrogram and vice versa. Although there is some overlap between them, it is not consistent. There are cases where

several genes from one cluster match a single cluster of the other type, but they are generally not in adjacent locations in the tree. This finding is consistent with studies of the correlation between expression and functional similarity, which found only very low levels of correlation if the data was considered on a pair-by-pair basis [Wang et al., 2004; Sevilla et al., 2005; Xu et al., 2008]. High correlations were only obtained if similarities were averaged across set intervals, as discussed in Section 2.4.

The same holds true when comparing clusters and groups. Sorting the visualisation matrix by expression tree location breaks up most of the groups, with the coloured cells representing a gene's group membership spread out across the length of the gene list. Blocks of consecutively located genes are only found in a few of the largest groups. These blocks however do not contain all the genes from these groups and there are sections of individual or pairs of genes far removed from the central block or blocks.

The overlap between semantic clusters and groups is better. When sorting the matrix by semantic tree location, many groups show a great deal of overlap with semantic clusters, with only individual genes responsible for gaps in otherwise solid blocks. Nonetheless, even in this alignment, there are genes that, while being grouped together, are far apart in the clustering dendrogram.

As group validation based on existing classifications is difficult, it was decided to focus on individual sets of genes to test the power of the FuSiGroups algorithm. For this purpose, three sets of genes were selected, namely genes involved in the proteasome, ribosomal genes and a collection of genes belonging to two different sets of metabolic pathways.

7.4 Summary

In this chapter, the results obtained from the FuSiGroups algorithm with the parameters in Table 7.1 for the full Eisen dataset were presented and analysed. It was found that the largest groups and most commonly represented functional aspects corresponded to general cellular processes, as would be expected in a dataset like the Eisen dataset, which contains genes from across the entire yeast genome. The level of overlap observed between many of the groups lead to the introduction of the concept of supergroups, in order to reduce duplication across groups. Groups with sufficient overlap in both definition and content were merged into supergroups, which considerably reduced the level of overlap.

A comparison of the grouping results to expression and semantic clustering was also performed. Overlap between both types of clustering, and between either type

of clustering and grouping was found to be limited and inconsistent, suggesting that the existing classifications would be a poor choice of benchmark against which to evaluate FuSiGroups groups.

In the next chapter, three smaller datasets will be used to evaluate the biological relevance of FuSiGroups groups.

Chapter 8

Biological evaluation

From Chapter 7, it is clear that a large dataset like the Eisen dataset, grouped using FuSiGroups, creates an equally large, hard to analyse set of grouping results. In order to evaluate the biological potential of the FuSiGroups algorithm, three smaller, less noisy sub-datasets of the Eisen dataset were selected and their grouping results were analysed separately in order to address specific questions about FuSiGroups' functionality, such as

- Is FuSiGroups able to identify the main functional aspects of a dataset?
- Does FuSiGroups produce biologically relevant groups?
- Is FuSiGroups able to identify gene products that are functionally unrelated to the majority of the gene products in a dataset?
- To what extent does functional grouping reflect other forms of biological relatedness?
  - Do the groups reflect the structure of gene expression clusters?
  - Do the groups identify distinct biological pathways?

In addition to these questions, this evaluation also helps to identify any issues with the FuSiGroups algorithm, as well as any other related issues that are not directly addressed by any of the above questions. For each of the three datasets, the process of selecting the genes and how it relates to the evaluation questions will be described. Then a summary of the grouping results and how they address the various questions is given, with a discussion of specific examples to illustrate the findings.

Parameters for all groupings, unless explicitly stated otherwise, will be the same as those used in Chapter 7, listed in Table 7.1.

8.1 Proteasome

8.1.1 Gene selection

The proteasome, or 26S proteasome, is a multisubunit enzyme complex that is highly conserved among all eukaryotic species. Its function is to digest unneeded or damaged proteins tagged with the regulatory protein Ubiquitin. As can be seen in Figure 8.1, the full proteasome consists of a symmetrical core complex, the 20S core particle, which contains the proteolytic active sites, and a 19S regulatory cap particle on each end. For further details on the structure and function of the proteasome, see Coux et al. [1996]; Wolf and Hilt [2004]; Feldmann [2005].

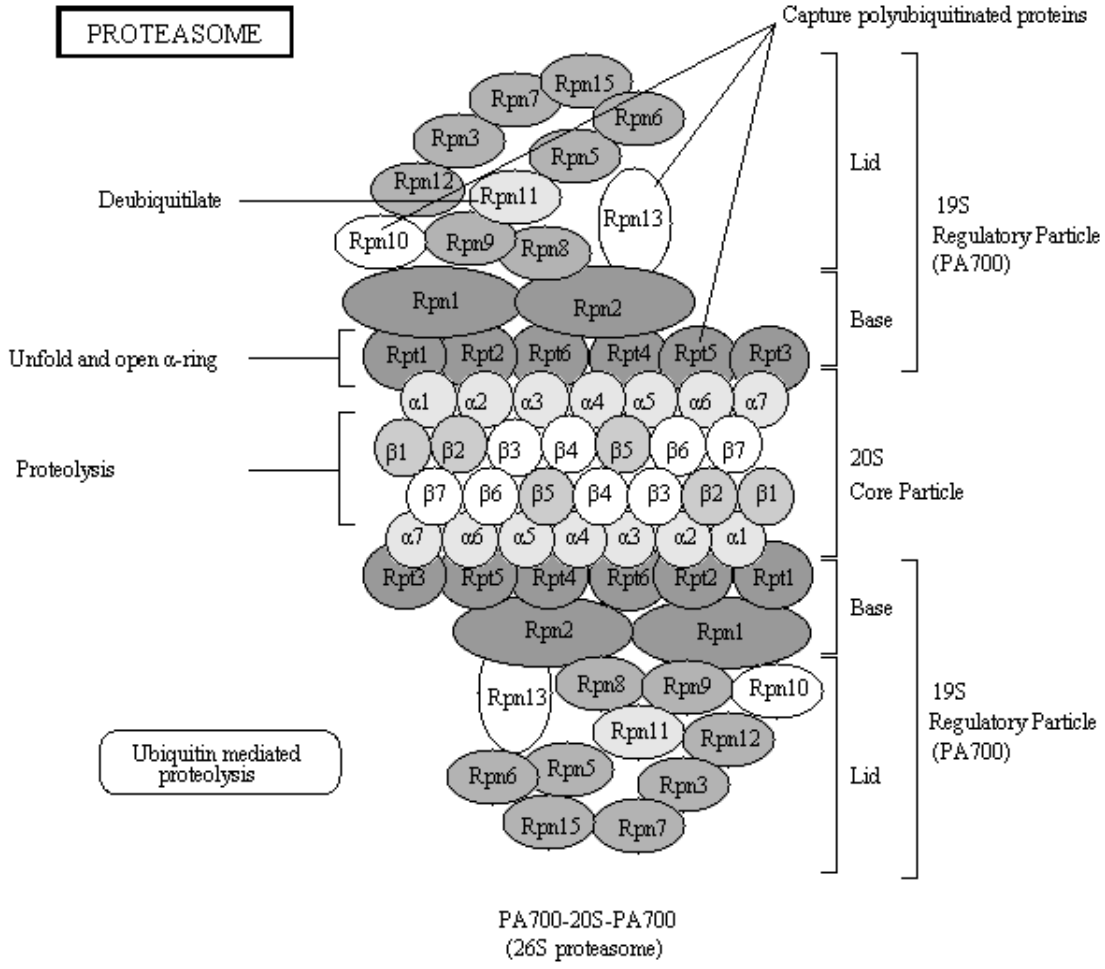

Figure 8.1: Diagram of the 26S proteasome. The image is taken from KEGG pathway map sce03050 [Kanehisa and Goto, 2000].

The pathway resource KEGG [Kanehisa and Goto, 2000] lists 35 genes as part of the proteasome. 32 of these genes are present in the Eisen dataset. One of the ten strong clusters identified by Eisen et al., cluster "C" (Figure 8.2, red box), consists of 27 genes involved in protein degradation. 26 of the 27 genes are found among proteasome genes listed in KEGG. The one exception is UFD1, a protein that forms part of a complex involved in the recognition of polyubiquitinated proteins for presentation to the 26S proteasome [Braun et al., 2002]. A further four proteasome genes can be found immediately adjacent to cluster C but were not included in the cluster (Figure 8.2, green boxes). The authors do not state how the clusters were identified and it is notable that the two genes at the top of the cluster, RPN5 and RPN8, are part of a different branch of the dendrogram than the core of the cluster, yet the three other genes of that branch, which are not part of the proteasome, were not included. Had that full branch been included, it would also have been possible to include the first two genes below the cluster, which are proteasome genes.

The clustering performed as part of this analysis resulted in a similar but not identical proteasome-based cluster. Depending on the level at which the tree is partitioned, 23 to 25 of Eisen cluster C's 27 genes are included in the recalculated cluster. RPN5 and RPN8 are not included within the cluster as they are located in a completely different part of the tree but between two and four of the proteasome genes not in C are in the replicated cluster. Several non-proteasome genes found adjacent to cluster C are also in the replicated clusters, as well as up to four non-proteasome genes not found near C.

For the sake of simplicity, the tree cutting thresholds were chosen at intervals of 0.1. In this case, the smallest cluster that included all of the cluster C genes except for RPN5 and RPN8 was found at a tree height of 0.5. This cluster includes 25 cluster C genes, 4 further proteasome genes, 5 non-proteasome genes found near cluster C and 4 other genes that are neither part of the proteasome nor found near cluster C. It was decided to include all of these genes in the proteasome sub-dataset. Also included were the 4 other proteasome genes found in the Eisen dataset but not within the recreated cluster (RPN1, RPN5, RPN8, RPT2), leading to a total of 42 genes.

The reason for including all the proteasome genes found in the Eisen dataset was to test FuSiGroups' ability to group together biologically related genes. If RPN1, RPN5, RPN8 and RPT2 are found to be grouped with the other proteasome genes, this would show an improvement of functional grouping over expression and semantic clustering, both of which separate these genes from the rest of the proteasome genes. Similarly, the inclusion in the dataset of non-proteasome genes clustered

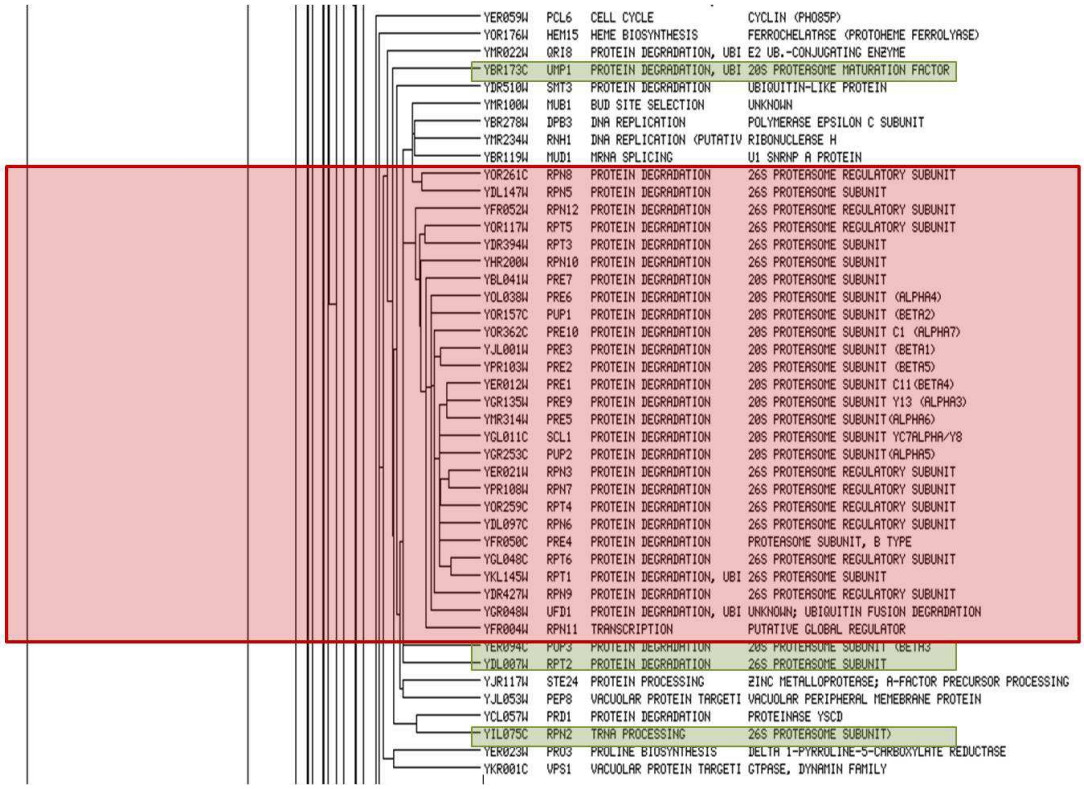

Figure 8.2: Eisen cluster C. The image was produced from the supplementary materials images provided with the paper. The heatmap located between the dendrogram and the gene list was removed. The red box shows the cluster identified by the authors. The green boxes indicate genes that are part of the proteasome but were not included in the cluster. The gene list is in reverse compared to the list published in the paper, as is the case with all the images, i.e. the first image corresponds to the bottom of the paper figure and the last image corresponds to the top. It should also be noted that in the paper, the last gene in this figure (first gene in the published figure), RPN11, has a different description in the paper than it does in the full image.

with proteasome genes in a gene expression context tests the algorithm's ability to identify functionally unrelated genes by not grouping them.

The full list of genes, ordered by their location in the expression dendrogram, is given in Table 8.1. Also included in this table is the location of each gene in the semantic dendrogram. While the ordering of the genes within the dendrogram is completely different to the ordering in the expression dendrogram, most of the proteasome genes (28 out of 32) are clustered in the range of 1882 to 1909. The four proteasome genes not found in this range are also clustered together, in location 858 to 861. All non-proteasome genes, including UFD1, are found at unrelated locations in the dendrogram. This may be an indication that semantic clustering is slightly better for identifying functionally related genes than expression clustering, or at least for differentiating between genes from a common pathway and genes not in that pathway.

The 42 genes of the proteasome dataset are annotated with 117 distinct GO terms. This translates into a total of 468 gene-GO term pairs, as each gene has multiple annotations and each GO term can be annotated to multiple genes.

8.1.2 Grouping

In the FuSiGroups grouping of the full dataset, the 42 genes listed in Table 8.1 are never all grouped together. There is also no group that contains only genes from this subset. 3 genes, UMP1, PRO3 and GSH2, are not found in any group at all. In the full grouping, there are exactly 100 groups (98 meaningful groups[1]) containing at least one of the subset genes. If supergroups are considered, this changes to 14 supergroups and 15 unmerged groups (13 meaningful groups). The groups containing the most genes from the subset are supergroups 112 and 149, named "proteasome complex" and supergroup 114, named "catabolic process". The subset genes make up 18 out of 28, 18 out of 27, and 22 out of 64 total genes, respectively. Of the remaining groups, the best matches are group 1243, named "proteolysis", with 22 out of 63 genes and group 1296, called "peptidase activity", with 27 out of 74 total genes.

If the subset of 42 genes is run separately on the FuSiGroups prototype, the result consists of a total of 49 distinct groups or 5 supergroups and 38 unmerged groups. This is a significant dimensional reduction from the original 117 GO terms and 468 individual annotations. In terms of meaningful groups, there are 17 groups

[1] As a reminder, a "meaningful group" is defined as a group containing at least 4 gene products, as stated in Section 3.1.6 and consistent with the minimum cluster size defined by Huang et al. [2007]

Expr. tree	Sem. tree	SGD ID	Gene name	Full name
1598	*2359*	*S000003568*	*BET4* ‡	*Alpha subunit of Type II geranylgeranyltransferase required for vesicular transport between the endoplasmic reticulum and the Golgi*
1599	*1649*	*S000004494*	*RAD52* ‡	*Protein that stimulates strand exchange by facilitating Rad51p binding to single-stranded DNA*
1600	*2143*	*S000000825*	*PRO3* †	*Delta 1-pyrroline-5-carboxylate reductase, catalyzes the last step in proline biosynthesis*
1601	*1909*	*S000000562*	*PRD1* †	*Zinc metalloendopeptidase, found in the cytoplasm and intermembrane space of mitochondria*
1602	1901	S000001337	RPN2 ∓†	Subunit of the 26S proteasome, substrate of the N-acetyltransferase Nat1p
1603	*1848*	*S000002918*	*SMT3* †	*Ubiquitin-like protein of the SUMO family, conjugated to lysine residues of target proteins*
1604	*1718*	*S000005409*	*GSH2* ‡	*Glutathione synthetase, catalyzes the ATP-dependent synthesis of glutathione (GSH) from gamma-glutamylcysteine and glycine*
1605	*1823*	*S000004624*	*UBC7 (as QRI8)* †	*Ubiquitin conjugating enzyme, involved in the ER-associated protein degradation pathway*
1606	1890	S000005889	PRE10	Alpha 7 subunit of the 20S proteasome
1607	*91*	*S000003589*	*PEP8* †	*Vacuolar protein sorting protein that forms part of the multimeric membrane-associated retromer complex along with Vps35p, Vps29p, Vps17p, and Vps5p*
1608	*1918*	*S000003878*	*STE24* †	*Highly conserved zinc metalloprotease that functions in two steps of a-factor maturation, C-terminal CAAX proteolysis and the first step of N-terminal proteolytic processing*
1609	1744	S000000377	UMP1 ∓†	Short-lived chaperone required for correct maturation of the 20S proteasome
1610	1892	S000000137	PRE7	Beta 6 subunit of the 20S proteasome
1611	1889	S000004557	PRE8 ∓‡	Alpha 2 subunit of the 20S proteasome
1612	1906	S000002802	RPT3	One of six ATPases of the 19S regulatory particle of the 26S proteasome involved in the degradation of ubiquitinated substrates
1613	1887	S000005683	PUP1	Beta 2 subunit of the 20S proteasome
1614	1905	S000001628	RPT1	One of six ATPases of the 19S regulatory particle of the 26S proteasome involved in the degradation of ubiquitinated substrates
1615	1908	S000003016	RPT6	One of six ATPases of the 19S regulatory particle of the 26S proteasome involved in the degradation of ubiquitinated substrates
1616	1898	S000001243	RPN10	Non-ATPase base subunit of the 19S regulatory particle (RP) of the 26S proteasome
1617	1885	S000003538	PRE3	Beta 1 subunit of the 20S proteasome, responsible for cleavage after acidic residues in peptides
1618	1882	S000006307	PRE2	Beta 5 subunit of the 20S proteasome, responsible for the chymotryptic activity of the proteasome
1619	1903	S000005785	RPT4	One of six ATPases of the 19S regulatory particle of the 26S proteasome involved in the degradation of ubiquitinated substrates
1620	861	S000002255	RPN6	Essential, non-ATPase regulatory subunit of the 26S proteasome lid required for the assembly and activity of the 26S proteasome
1621	1902	S000001946	PRE4	Beta 7 subunit of the 20S proteasome
1622	1899	S000001948	RPN12	Subunit of the 19S regulatory particle of the 26S proteasome lid
1623	860	S000002835	RPN9	Non-ATPase regulatory subunit of the 26S proteasome, has similarity to putative proteasomal subunits in other species
1624	1900	S000002979	SCL1	Alpha 1 subunit of the 20S proteasome involved in the degradation of ubiquitinated substrates
1625	1888	S000005398	PRE6	Alpha 4 subunit of the 20S proteasome
1626	1904	S000005643	RPT5	One of six ATPases of the 19S regulatory particle of the 26S proteasome involved in the degradation of ubiquitinated substrates
1627	553	S000003280	UFD1 *	Protein that interacts with Cdc48p and Npl4p, involved in recognition of polyubiquitinated proteins and their presentation to the 26S proteasome for degradation
1628	1895	S000000823	RPN3	Essential, non-ATPase regulatory subunit of the 26S proteasome lid, similar to the p58 subunit of the human 26S proteasome
1629	859	S000006312	RPN7	Essential, non-ATPase regulatory subunit of the 26S proteasome, similar to another S. cerevisiae regulatory subunit, Rpn5p, as well as to mammalian proteasome subunits
1630	1893	S000000896	PUP3 ∓†	Beta 3 subunit of the 20S proteasome involved in ubiquitin-dependent catabolism
1631	1897	S000001900	RPN11	Metalloprotease subunit of the 19S regulatory particle of the 26S proteasome lid
1632	1884	S000003485	PUP2	Alpha 5 subunit of the 20S proteasome involved in ubiquitin-dependent catabolism
1633	1891	S000004931	PRE5	Alpha 6 subunit of the 20S proteasome
1634	1883	S000000814	PRE1	Beta 4 subunit of the 20S proteasome
1635	1886	S000003367	PRE9	Alpha 3 subunit of the 20S proteasome, the only nonessential 20S subunit
1977	1894	S000005787	RPN8	Essential, non-ATPase regulatory subunit of the 26S proteasome
1978	1907	S000002165	RPT2 ∓†	One of six ATPases of the 19S regulatory particle of the 26S proteasome involved in the degradation of ubiquitinated substrates
1979	858	S000002306	RPN5	Essential, non-ATPase regulatory subunit of the 26S proteasome lid, similar to mammalian p55 subunit and to another S. cerevisiae regulatory subunit, Rpn7p
2319	1896	S000001069	RPN1 ∓‡	Non-ATPase base subunit of the 19S regulatory particle of the 26S proteasome

Table 8.1: The proteasome subset. The first column shows the location of each gene in the expression tree, the second column its location in the semantic tree. Genes in *italics* are not found in Eisen cluster C; ∗ - gene in cluster C but not in proteasome; ∓ - gene in proteasome but not in cluster C; † - gene near cluster C in Figure 8.2; ‡ - gene not near cluster C in Figure 8.2.

meeting minimum size requirement among the total 49 groups and 8 among the 38 unmerged groups, which further reduces the complexity of the analysis without loss of information. Every gene in the dataset is grouped at least twice, although six of the genes are only found in groups with fewer than four genes. These genes are GSH2, BET4, RAD52, SMT3, PRO3 and PEP8, i.e. only genes that are neither part of the proteasome nor code for proteins with any other known proteolytic activity. This is already a strong indicator that FuSiGroups is capable of identifying genes that have no functional relationship with the majority of the dataset.

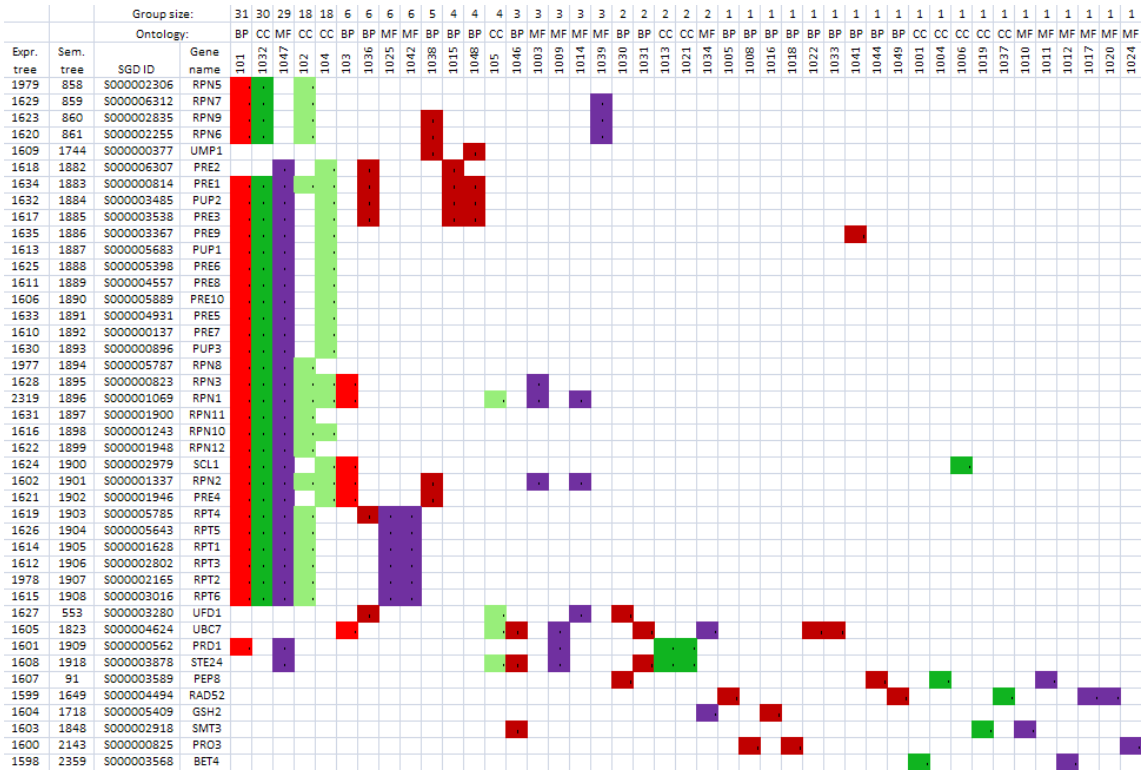

Figure 8.3: Supergroups and unmerged groups for the proteasome dataset, ordered left to right by size and top to bottom by three categories, then by the location of the genes in the semantic dendrogram for each category. The three categories are genes coding 1. proteins included in the proteasome according to KEGG, 2. other proteolytic enzymes, 3. proteins without known proteolytic properties. BP groups are represented in red, CC groups in green and MF groups in purple. Supergroups are in a lighter tone of the same colour as the corresponding type of unmerged groups. There are no MF supergroups. All group sizes are included in order to demonstrate that some of the genes are never grouped with any other gene. The first row shows the group size, the second row the ontological type of the group and the third row lists the group's ID number. The first column lists the location of each gene in the expression tree, the second column its location in the semantic tree. The third column shows the gene's SGD ID and fourth column the gene name.

Figure 8.3 shows all the supergroups and unmerged groups, including groups with fewer than four genes. These groups were included so that the afore-mentioned six genes, located in the bottom six rows, can be included without their respective

rows being marked as blank. The smaller groups were however excluded from the list of group names in Table 8.2.

Group ID	Group name	Ontology	Group size
101	protein metabolic process *(GO:0019538)*	BP	31
1032	cytosol *(GO:0005829)*	CC	30
1047	peptidase activity *(GO:0008233)*	MF	29
102	proteasome complex *(GO:0000502)*	CC	18
104	proteasome complex *(GO:0000502)*	CC	18
103	regulation of protein metabolic process *(GO:0051246)*	BP	6
1036	cellular process *(GO:0009987)*	BP	6
1025	nucleoside-triphosphatase activity *(GO:0017111)*	MF	6
1042	ATP binding *(GO:0005524)*	MF	6
1038	cellular macromolecular complex assembly *(GO:0034622)*	BP	5
1015	reproduction *(GO:0000003)*	BP	4
1048	response to stress *(GO:0006950)*	BP	4
105	cell part *(GO:0044464)*	CC	4

Table 8.2: Names of all the meaningful groups and supergroups shown in Figure 8.3, sorted by decreasing size. Three-digit group IDs indicate supergroups, four-digit IDs normal groups.

Although some of the group names in Table 8.2 are fairly high-level, such as "protein metabolic process" or "cell part", most of the names in the list are representative of the functional aspects expected to be associated with proteasome genes. Only one group name, reproduction (group 1015), stands out as having no obvious connection to proteasome function. The reason for the presence of this group will be discussed in the next section.

From the analysis of content and definitions of the 13 groups listed in Table 8.2, a number of recurring themes can be identified. A summary of all the groups is given in Table 8.3, providing details about whether each group is biologically relevant in the context of the proteasome dataset, whether it is affected by any annotation issues and whether any potential algorithmic refinements can be identified from it. A full analysis of each individual group can found in Appendix B.1.

Annotation issues

The analysis of the groups shows that overall, the FuSiGroups algorithm performs well and produces groups of functionally similar genes representing the major functional aspects of the proteasome. The fact that the 32 genes associated with the proteasome according to KEGG are never all found in the same group together is due to variations in their individual annotations. These either lead to exclusion of some genes due to insufficient levels of functional similarity with other genes or to exclusion due to the absence of any appropriate definition term in a gene's anno-

Group ID	Biologically relevant?	Annotation issues	Potential algorithmic improvements
101	Yes	∗PRE2	Refinement of definition based on group content
1032	Yes	∗PRE2, UMP1	
1047	Yes	∗RPN5, RPN6, RPN7, RPN9	
102	Yes		
104	Yes		
103	Yes	PRE4, SCL1	
1036	*No*	*PRE1, PRE2, PRE3, PUP2, RPT4, UFD1*	
1025	Yes		
1042	Yes		
1038	Yes		Refinement of definition based on group content
105	Yes	RPN1	Refinement of definition based on group content
1015	*No*	*PRE1, PRE2, PRE3, PUP2*	
1048	Yes	PRE1, PRE3, PUP2	Refinement of definition based on group content

Table 8.3: Summary of all meaningful groups for the proteasome dataset, with respect to the key issues identified in the analysis. Entries in the "Annotation issues" column preceded by ∗ indicate that the issues in question led to the absence of these genes from the group when they could reasonably have been expected to be included in the group. Groups in *italics* were found to be biologically irrelevant in the context of the proteasome datset.

tations. This is one of the drawbacks of functional similarity itself, rather than an issue with the FuSiGroups algorithm.

More specifically, a number of annotation issues were identified, ranging from the complete absence of functional annotation in certain categories for some genes to obviously incorrect annotations in others. The most pertinent examples of this will be discussed here.

Two of the thirteen groups in Table 8.3 were found to not be biologically relevant with respect to the proteasome dataset. One of these groups, group 1015, has already been noted for its unexpected group name in Table 8.2. The reason for this group's lack of relevance is the same as for group 1036 and as this group has more affected genes, this one will be discussed here.

Group 1036's definition is a collection of GO terms which, although semantically similar, are biologically quite diverse. Two of the terms refer to sporulation, one to cell differentiation and one to cell death. None of these processes are directly associated with proteasome function. Their presence in the proteasome dataset is the result of a number of dubious annotations of the genes in the group.

The most notable case of this, both in this group and in the sub-dataset is PRE2, which is annotated with five BP terms that are inconsistent with proteasome function. Four of these are reproduction-related terms despite there being no direct evidence of PRE2's involvement in reproductive processes. All five annotations are qualified with the "RCA" (inferred from Reviewed Computational Analysis) evidence code and all are no longer present in the latest version (2011-01) of the GO. Four of the five annotations are unique to PRE2 in the proteasome dataset, which strongly affects PRE2's functional similarity with other genes, leading to its exclusion from groups such as 102 and 1032, where its presence would have been biologically appropriate.

The same problem affects the other five genes in group 1036, although to a lesser extent. PRE1, PRE3 and PUP2 are all annotated with one of PRE2's four reproduction-related terms, ascospore formation. In this case, the associations are qualified as "TAS" (Traceable Author Statement), an evidence code often falsely regarded as a guarantee for high-quality annotation. The associated reference, Hochstrasser [1996], does however not provide any evidence for the involvement of these genes in ascospore formation. It only states "Required for sporulation [...]" as a comment in a list of proteasome genes, without reference or further discussion of this statement. In the latest version of the GO, this annotation has been withdrawn and replaced with the term's parent sporulation resulting in formation of a cellular spore, evidence code RCA, referenced with Huttonhower and Troyanskaya [2009].

For genes PUP2, RPT4 and UFD1, the potentially questionable annotation is the term cell death. The associations are qualified with the RCA evidence code based on the same source as PRE2's RCA annotations, Huttonhower and Troyanskaya [2009], from which the majority of RCA annotations in SGD are derived, and all three are absent from the latest version of GO.

While group 1036 is the best example of the effect of potentially questionable annotations, other groups are affected by these examples. They include group 1015, which is also rendered biologically not relevant in the context of the proteasome through these annotations. Other groups are also affected, although to a lesser extent.

Although the most striking, dubious annotations are not the only annotation issue identified during the analysis of the proteasome groups. A number of genes, UMP1, RPN5, RPN6, RPN7 and RPN9 suffer from poor annotation which leads to their exclusion from a number of groups, either because none of their annotation terms match the definition of a key group or even because it incorrectly lowers their functional similarity to some of the other genes in the dataset. The latter applies to UMP1, which does not have any MF annotation despite being relatively well studied [Li et al., 2007]. The four RPN genes listed here suffer from the former issue. Even though their MF annotation is as poor as that for UMP1, with no annotation for RPN5 and only a very high-level term, structural molecule activity (GO:0005198), for the other three, the rest of their annotations match the majority of the other proteasome subunits very closely. This means that their functional similarity is sufficiently high for them to be included in non-MF groups but they do not have any suitable annotations to qualify them for inclusion in MF groups.

Potential algorithmic improvements

With respect to the grouping algorithm, a number of groups have a definition that is not entirely representative of the group's content, i.e. not all GO terms in the definition are found in the annotations of the genes in the group. The reason for this is that while the terms in the definition may be sufficiently semantically similar to be grouped together, the same does not apply to all the genes they annotate. This may lead to group names that are more general than they need to be for the genes in the group. A minor modification of the grouping algorithm, checking the group definition against the annotations of the gene products in the group after the group content has been finalised and removing any unrepresented terms from the definition, would improve the grouping results.

The group that best illustrates this situation is group 1038, cellular macromolecular complex assembly (GO:0034622). Two of this group's three definition terms are not associated with any of the genes in the group. In the proteasome dataset, they are in fact only annotated to RAD52, which is not involved with proteasome function and therefore has low functional similarity with the majority of the genes in the dataset. If this group's definition was re-checked against the content's annotations, the two terms annotated to RAD52 would be removed, leaving the term proteasome assembly (GO:00432480), which perfectly captures the functional aspect represented by this group.

Group overlap

Although it was not encountered in this set of results, there is a possibility of inappropriate overlap between groups, particularly supergroups. In fact, there are only two groups of the same functional type that are entirely identical, namely MF group 1025, nucleoside-triphosphatase activity (GO:0017111) and MF group 1042, ATP binding (GO:0005524). The overlap between these two groups is not only appropriate but a perfect example of distinct but related functional aspects of a set of genes. All six genes in the two groups are ATPase subunits found in the 19S regulatory particle of the proteasome. ATPase activity is a form of nucleoside-triphosphatase activity and both these GO terms are present in group 1025's definition. In addition, ATPase activity necessitates the binding of ATP, which is why group 1042's content is identical to group 1025's. At the same time, the merger of the two groups into a supergroup would have been inappropriate as the concepts represented by each group are completely different from an ontological perspective, as reflected by the fact that their only common ancestor is the root of the MF ontology.

Overall, the proteasome dataset actually provides a good illustration of the different scenarios that can lead to the creation of supergroups. Figure 8.4 shows the five supergroups created for the proteasome dataset and the groups they originate from. As the proteasome dataset is fairly small, all the supergroups are derived from the merger of two groups and all original groups are merged no more than once. It is of course possible to merge more than two groups or to merge a group more than once, as described in Section 7.2.

Some groups, such as groups 1043 and 1045, which form supergroup 104, have identical content but have slightly different definitions. In the present case, the two definitions differ in one of their three respective terms, which means they meet the merging criteria.

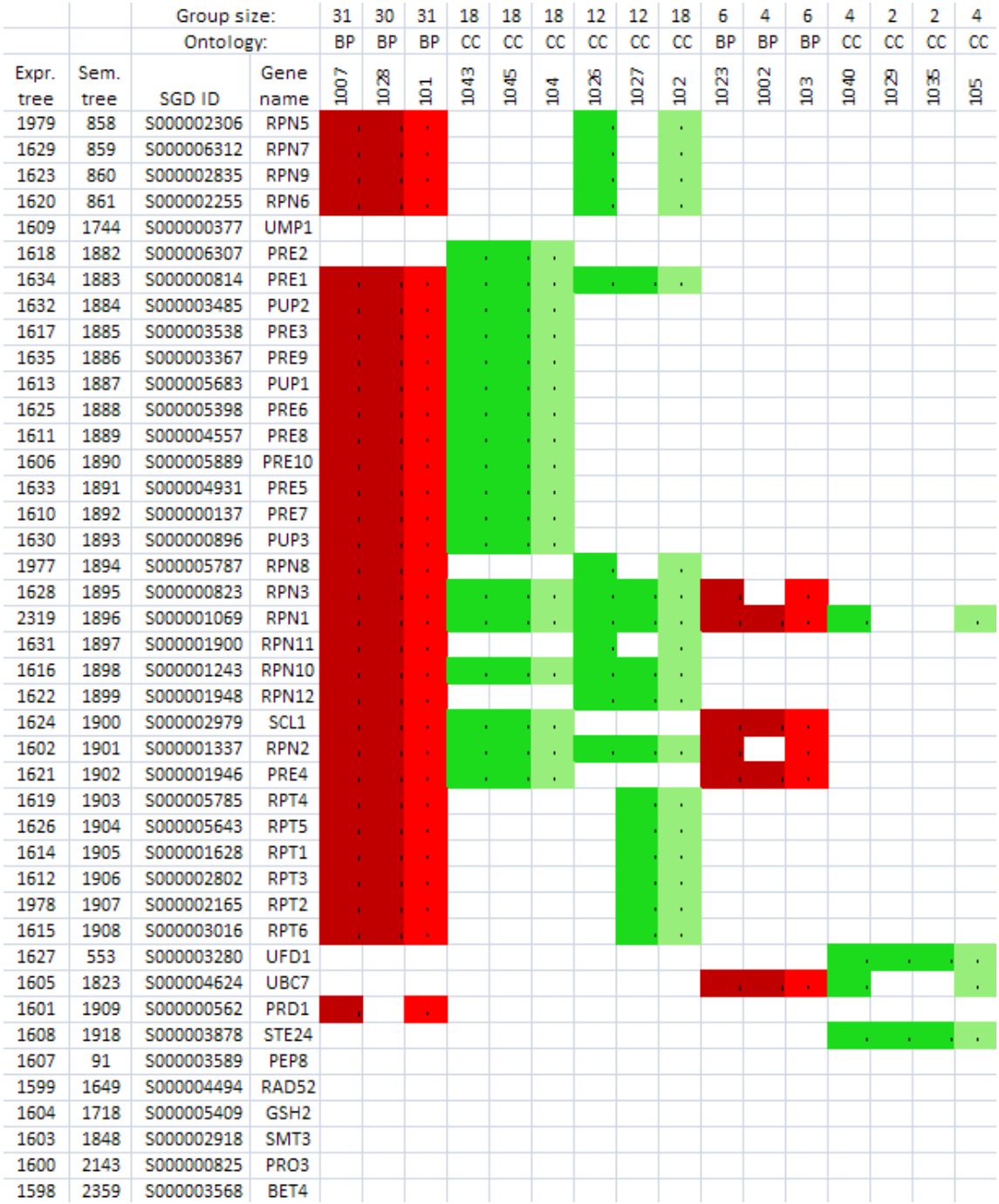

Figure 8.4: Illustration of original groups that were merged into supergroups. Conventions and ordering the same as in Figure 8.3.

The most different groups that were merged are 1026 and 1027, which result in supergroup 102. They have a content overlap of six out of their respective twelve terms, while their definition overlap is two terms out of three, with a shared group name. The level of difference in group content and the fact that the non-overlapping definition terms are "proteasome regulatory particle, lid subcomplex" and "proteasome regulatory particle, base subcomplex", two terms referring to different parts of the proteasome, might suggest that a merger is biologically inappropriate for these two groups. The two overlapping terms however are less location-specific, so in the light of the full definition, the merger appears appropriate. As long as an algorithm like the merging process has to be based on predefined criteria, borderline cases such as this, which can only really be resolved using human judgement, will occur. It may therefore be of interest to consider the original two groups separately as well.

The remaining three sets of pre- and post-merging groups are all groups in which content-wise, one of the merged groups is a subset of the other. In each case, there is also some difference between the definitions, but with a term overlap of at least 50%.

8.2 Ribosomal genes

8.2.1 Gene selection

In the previous section, the ability of the FuSiGroups algorithm to group together genes from the same pathway and exclude unrelated genes has been demonstrated. Usually, one would not expect the type of dataset that requires grouping to be as well-defined as the proteasome dataset. For this reason, a more diverse dataset was chosen to test whether the resulting groups could identify the original expression clusters in the dataset.

In the Eisen paper, one cluster was not defined in the same way as the other nine. Cluster "I", the largest of the ten clusters, contains 112 ribosomal protein genes, seven translation initiation or elongation factors, three tRNA synthetases, and three genes of apparently unrelated function. Due to its size, the individual genes in the cluster are not listed and it is marked simply with the functional term "protein synthesis". As the paper does not provide exact details on how the clusters were selected, it is difficult to determine the exact content of cluster I even from the supplementary figures, particularly in the light of the cluster boundary issues discussed in relation to Figure 8.2. There is however little doubt that this protein synthesis cluster is a very strong cluster, with many strongly co-expressed genes.

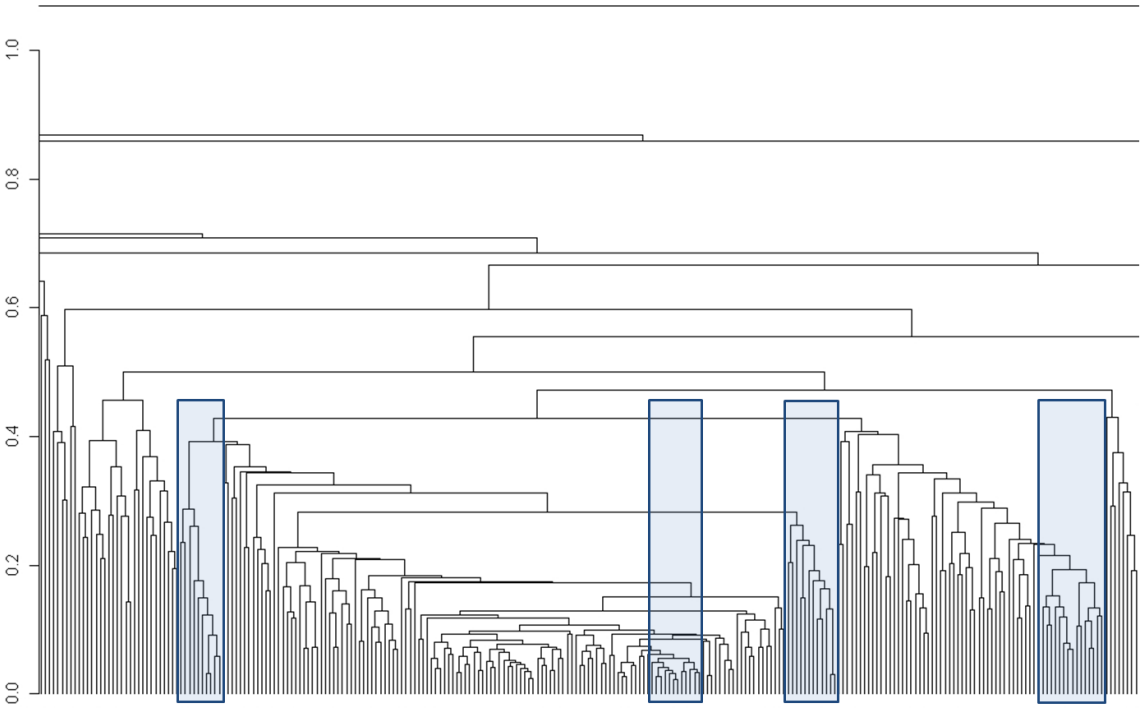

Figure 8.5: Expression dendrogram locations 840 to 1100. The blue boxes show the clusters which make up the ribosome dataset. Gene names were excluded from the figure as they are unreadable at this resolution.

When searching our re-clustered dendrogram for genes containing the term "ribosom"[2] in their name, the most continuous set of locations in which these genes are found lies between locations 840 and 1100. There are two smaller fairly coherent blocks, but with mitochondrial ribosomal proteins. The largest block is therefore most likely to contain Eisen cluster I. As it was virtually impossible to determine exactly which genes were included in cluster I and as the size of cluster I would make detailed analysis of the grouping laborious, it was decided to select several different clusters from across the entire region and gather them into a single dataset. This also allows the question of how closely FuSiGroups reflects other forms of biological relatedness to be addressed with respect to gene expression clusters. In total, four clusters were selected, with a total of 49 genes. Figure 8.5 shows the distribution of the clusters across the region of the tree.

Ribosomes are hugely complex cellular components that translate mRNA into proteins. Eukaryotic ribosomes are composed of roughly two-thirds ribosomal RNA (rRNA) and one third protein. The protein component is divided into the small 40S ribosomal subunit, which contains about 33 proteins and the large 60S ribosomal

[2]This spelling is deliberate as a search for this term will include gene names containing the word "ribosomal" as well "ribosome"

subunit containing 49 different proteins. The small subunit contains one rRNA, the 18S rRNA, while the large subunit is based around three rRNAs, the 5S rRNA, the 28S rRNA and the 5.8rRNA.[Alberts et al., 2002]

The KEGG pathway resource associates 159 distinct entities with the ribosome pathway in yeast. 15 of these refer to rRNAs, 4 to mitochondrial ribosomal proteins. None of the rRNAs are included in the Eisen dataset. Of the 144 proteins in the list, 128 are present in the Eisen dataset. 3 of these are mitochondrial ribosomal proteins, the remaining 125 are normal ribosomal proteins. 122 of the 125 ribosomal proteins can be found in the identified expression tree range. The ribosome dataset selected for further analysis contains 21 of these, including 20 ribosomal proteins and one protein essential for ribosomal large subunit biogenesis (RLP24).

In addition to these 21 genes, the ribosome dataset contains 28 other genes, making a total of 49 genes. Five of these are translation initiation or elongation factors, nine are in some way involved in ribosome assembly, eight have some kind of RNA synthesis activity and six genes are not obviously related to any of these processes. The full list of genes is given in Table 8.4.

There are 147 distinct GO terms associated with the 49 genes in the ribosome subset, forming 521 distinct gene-GO term annotation pairs.

8.2.2 Grouping

As with the proteasome dataset, the genes from the ribosome dataset are never all grouped together in the full grouping, nor are there any groups that only contain genes from the dataset, except for groups of size 1. In total, ribosome dataset genes are found in 72 distinct groups, or 10 supergroups and 28 unmerged groups. In both cases, all but one group are meaningful groups. Three genes, SQT1, PSE1 and YNL247W, do not occur in any of these groups. The largest set of ribosome genes grouped together are 21 or 22 genes in groups such as “translation” or “ribosome”, although these genes represent only around 10% of the total group content.

When grouped separately, the ribosome dataset is grouped into 59 groups, or 4 supergroups and 45 unmerged groups. As with the proteasome dataset, this is a considerable dimensional reduction compared to the 147 GO terms and 521 annotations of the dataset. 26 of the total 59 groups are meaningful groups, and the same applies to 13 of the 45 unmerged groups. Four genes, SHM1, PSE1, SAH1 and IMD2, are not grouped into any meaningful groups. Figure 8.6 shows all supergroups and unmerged groups for the ribosome dataset, including non-meaningful groups, while Table 8.5 lists all the meaningful supergroups and unmerged groups,

Expr. tree	Sem. tree	SGD ID	Gene name	Full name
863	814	S000000322	TEF2 ∓	Translational elongation factor EF-1 alpha
864	2133	S000000467	SHM1 ‡	Mitochondrial serine hydroxymethyltransferase, converts serine to glycine plus 5,10 methylenetetrahydrofolate
865	943	S000001663	RPL17A *	Protein component of the large (60S) ribosomal subunit, nearly identical to Rpl17Bp and has similarity to E. coli L22 and rat L17 ribosomal proteins
866	983	S000000252	RPS11B *	Protein component of the small (40S) ribosomal subunit
867	926	S000000993	RPL14B *	Protein component of the large (60S) ribosomal subunit, nearly identical to Rpl14Ap and has similarity to rat L14 ribosomal protein
868	923	S000000183	RPL23A *	Protein component of the large (60S) ribosomal subunit, identical to Rpl23Bp and has similarity to E. coli L14 and rat L23 ribosomal proteins
869	1005	S000000393	RPS9B *	Protein component of the small (40S) ribosomal subunit
870	941	S000000395	RPL21A *	Protein component of the large (60S) ribosomal subunit, nearly identical to Rpl21Bp and has similarity to rat L21 ribosomal protein
871	986	S000001025	RPL8A *	Ribosomal protein L4 of the large (60S) ribosomal subunit, nearly identical to Rpl8Bp and has similarity to rat L7a ribosomal protein
872	987	S000002858	RPS18A *	Protein component of the small (40S) ribosomal subunit
974	962	S000004317	RPL38 *	Protein component of the large (60S) ribosomal subunit, has similarity to rat L38 ribosomal protein
975	974	S000003317	RPL11B *	Protein component of the large (60S) ribosomal subunit, nearly identical to Rpl11Ap
976	975	S000006306	RPL11A *	Protein component of the large (60S) ribosomal subunit, nearly identical to Rpl11Bp
977	995	S000003157	RPS26A *	Protein component of the small (40S) ribosomal subunit
978	1022	S000002999	RPL24A *	Ribosomal protein L30 of the large (60S) ribosomal subunit, nearly identical to Rpl24Bp and has similarity to rat L24 ribosomal protein
979	1012	S000004433	RPS1A *	Ribosomal protein 10 (rp10) of the small (40S) subunit
980	1014	S000004528	RPS1B *	Ribosomal protein 10 (rp10) of the small (40S) subunit
981	970	S000004065	RPL10 *	Protein component of the large (60S) ribosomal subunit, responsible for joining the 40S and 60S subunits
982	992	S000005122	RPS3 *	Protein component of the small (40S) ribosomal subunit, has apurinic/apyrimidinic (AP) endonuclease activity
983	990	S000005246	RPS19B *	Protein component of the small (40S) ribosomal subunit, required for assembly and maturation of pre-40 S particles
984	996	S000000933	RPS26B *	Protein component of the small (40S) ribosomal subunit
985	931	S000001314	RPL34B *	Protein component of the large (60S) ribosomal subunit, nearly identical to Rpl34Ap and has similarity to rat L34 ribosomal protein
1006	88	S000004925	PSE1 ‡	Karyopherin/importin that interacts with the nuclear pore complex
1007	1763	S000005191	YNL247W †	Cysteinyl-tRNA synthetase
1008	2021	S000000845	SAH1 ‡	S-adenosyl-L-homocysteine hydrolase, catabolizes S-adenosyl-L-homocysteine which is formed after donation of the activated methyl group of S-adenosyl-L-methionine (AdoMet) to an acceptor
1009	1782	S000000325	GRS1 †	Cytoplasmic and mitochondrial glycyl-tRNA synthase that ligates glycine to the cognate anticodon bearing tRNA
1010	535	S000001106	SSZ1 ±	Hsp70 protein that interacts with Zuo1p (a DnaJ homolog) to form a ribosome-associated complex that binds the ribosome via the Zuo1p subunit
1011	819	S000004239	YEF3 ∓	Translational elongation factor 3, stimulates the binding of aminoacyl-tRNA (AA-tRNA) to ribosomes by releasing EF-1 alpha from the ribosomal complex
1012	1795	S000004830	GUA1 ‡	GMP synthase, an enzyme that catalyzes the second step in the biosynthesis of GMP from inosine 5'-phosphate (IMP)
1013	813	S000001564	TEF4 ∓	Translation elongation factor EF-1 gamma
1014	450	S000004284	GSP1 †	Ran GTPase, GTP binding protein (mammalian Ranp homolog) involved in the maintenance of nuclear organization, RNA processing and transport
1015	2091	S000001259	IMD2 ‡	Inosine monophosphate dehydrogenase, catalyzes the first step of GMP biosynthesis, expression is induced by mycophenolic acid resulting in resistance to the drug, expression is repressed by nutrient limitation
1016	824	S000001767	TIF1 ∓	Translation initiation factor eIF4A, identical to Tif2p
1017	823	S000003674	TIF2 ∓	Translation initiation factor eIF4A, identical to Tif1p
1066	1079	S000000451	ENP1 ±†	Protein associated with U3 and U14 snoRNAs, required for pre-rRNA processing and 40S ribosomal subunit synthesis
1067	1102	S000001451	SQT1 ±	Essential protein involved in a late step of 60S ribosomal subunit assembly or modification
1068	1093	S000001732	DBP7 ±	Putative ATP-dependent RNA helicase of the DEAD-box family involved in ribosomal biogenesis
1069	2225	S000000135	URA7 ‡	Major CTP synthase isozyme (see also URA8), catalyzes the ATP-dependent transfer of the amide nitrogen from glutamine to UTP, forming CTP, the final step in de novo biosynthesis of pyrimidines

Continued on next page

Continued from previous page				
Expr. tree	Sem. tree	SGD ID	Gene name	Full name
1070	1204	S000003088	PRP43 †	RNA helicase in the DEAH-box family, functions in both RNA polymerase I and polymerase II transcript metabolism, involved in release of the lariat-intron from the spliceosome
1071	1284	S000005192	RPA49 †	RNA polymerase I subunit A49
1072	297	S000001213	NMD3 ±	Protein involved in nuclear export of the large ribosomal subunit
1073	1075	S000003391	NSR1 ±†	Nucleolar protein that binds nuclear localization sequences, required for pre-rRNA processing and ribosome biogenesis
1074	1081	S000004187	NOP56 †	Essential evolutionarily-conserved nucleolar protein component of the box C/D snoRNP complexes that direct 2'-O-methylation of pre-rRNA during its maturation
1075	1085	S000005837	NOP58 †	Protein involved in pre-rRNA processing, 18S rRNA synthesis, and snoRNA synthesis
1076	1105	S000000023	MAK16 ±	Essential nuclear protein, constituent of 66S pre-ribosomal particles
1077	1104	S000001492	MRT4 ±	Protein involved in mRNA turnover and ribosome assembly, localizes to the nucleolus
1078	1076	S000003999	RLP24 *±	Essential protein with similarity to Rpl24Ap and Rpl24Bp, associated with pre-60S ribosomal subunits and required for ribosomal large subunit biogenesis
1079	1082	S000003934	SOF1 ±	Essential protein required for biogenesis of 40S (small) ribosomal subunit
1080	1273	S000005057	RPC19 †	RNA polymerase subunit, common to RNA polymerases I and III

Table 8.4: The ribosome subset. $*$ - gene in KEGG ribosome pathway; $\mp$ translation initation or elongation factors; $\pm$ - involved with ribosome assembly or function but not ribosomal protein; † - RNA synthesis activity (inc. RNA polymerase); ‡ - not obviously related to any of the other functions.

ordered by decreasing group size.

Two things immediately stand out in Figure 8.6. First of all, the three largest groups are all identical in content. The names of two of these groups, in Table 8.5, are also identical. There is also in general more duplication between groups than in the previous dataset. This is the result of the greater diversity of the ribosome dataset. The similarity in gene content of different groups in fact highlights the different functional aspects shared by a set of genes. A very good example of this is supergroups 102 and 104, which share 19 of their 19 and 20 respective genes. Although group 102's name is too generic to give any indication of the detailed functional aspect it represents, this group reflects the RNA aspects of ribosome biogenesis, while group 104 reflects the protein aspects of the same process. Another good example of this kind includes groups 1036 and 1050, which are identical in content and represent two distinct but related molecular functions.

Secondly, there is little obvious separation of group content according to the clusters in the expression tree. The first few groups do not contain any genes from the right-most cluster in Figure 8.5 (locations 1066 to 1080) and a few other groups do not contain any genes from other clusters but there is no clear trend of groups reflecting the original clusters. There is also no indication that genes from one particular cluster are always or never grouped with genes from another cluster. This finding is consistent with other work that suggests that while there is a high level of correlation between functional similarity and gene expression similarity if the

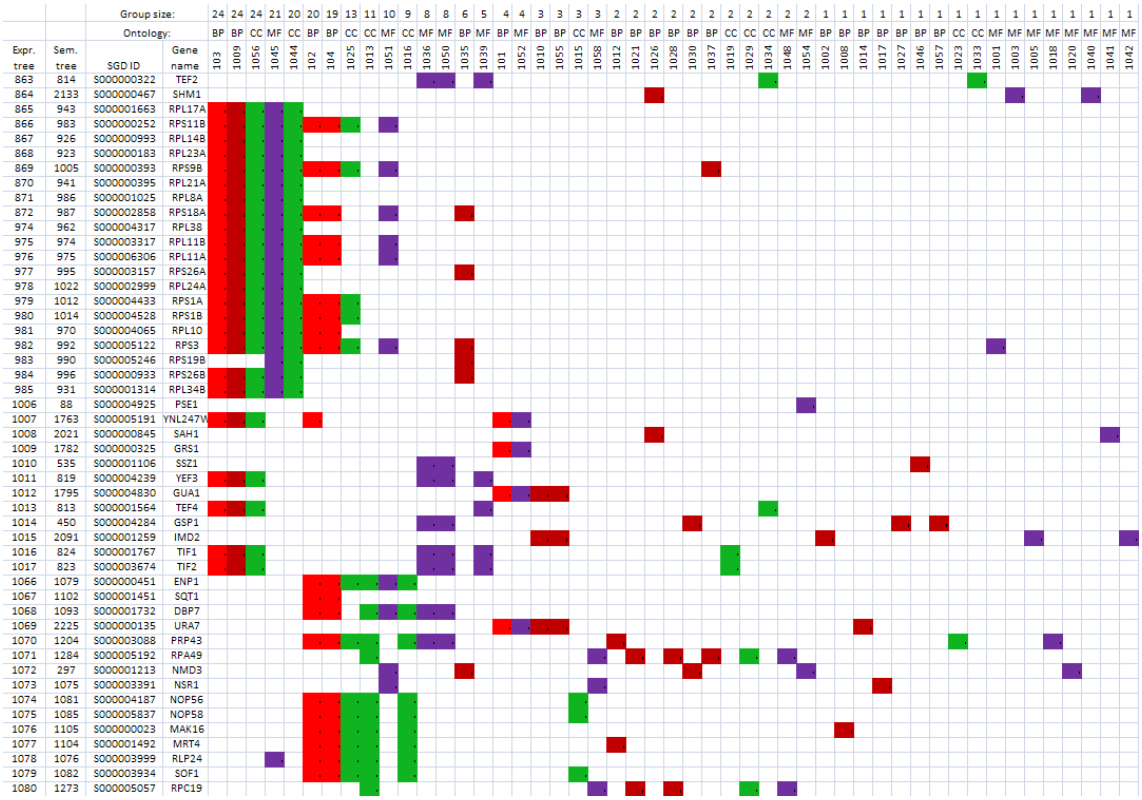

Figure 8.6: Supergroups and unmerged groups for the ribosome dataset, ordered left to right by size and top to bottom by their location in the expression tree. Conventions are the same as in Figure 8.3.

two concepts are considered across intervals of similarity, there is little consistent correlation if the comparison is done on a gene pair by gene pair basis.

Other datasources are however more closely reflected in some of the groups. All the genes from this subdataset that are associated with the ribosome in KEGG are grouped together in group 1045, structural constituent of ribosome, and all but one, whose annotation does not include any of the definition terms, are found in 1044, cytosol.

Group ID	Group name	Ontology	Group size
103	translation *(GO:0006412)*	BP	24
1009	translation *(GO:0006412)*	BP	24
1056	ribosome *(GO:0005840)*	CC	24
1045	structural constituent of ribosome *(GO:0003735)*	MF	21
1044	cytosol *(GO:0005829)*	CC	20
102	cellular process *(GO:0009987)*	BP	20
104	ribosome biogenesis *(GO:0042254)*	BP	19
1025	preribosome *(GO:0030684)*	CC	13
1013	nucleolus *(GO:0005730)*	CC	11
1051	RNA binding *(GO:0003723)*	MF	10
1016	nucleolus *(GO:0005730)*	CC	9
1036	purine ribonucleotide binding *(GO:0032555)*	MF	8
1050	hydrolase activity, acting on acid anhydrides *(GO:0016817)*	MF	8
1035	intracellular transport *(GO:0046907)*	BP	6
1039	translation factor activity, nucleic acid binding *(GO:0008135)*	MF	5
101	amino acid metabolic process *(GO:0006520)*	BP	4
1052	ligase activity *(GO:0016874)*	MF	4

Table 8.5: Names of all the meaningful groups and supergroups shown in Figure 8.6, sorted by decreasing size. Three-digit group IDs indicate supergroups, four-digit IDs normal groups.

Based on Table 8.5, group names vary from very high-level categories, such as cellular process, to quite detailed terms, like nucleolus. There are no group names that are obviously inconsistent with ribosomal functions. This initial conclusion is confirmed by detailed analysis of the grouping results, summarised in Table 8.6. While none of the groups contain unexpected genes, a number of groups have genes missing that could have been reasonably expected to be in the group, because these genes have insufficiently high functional similarity with some or all of the genes in the group. The reason for this is that the ribosome dataset is quite functionally diverse, leading to lower overall levels of functional similarity.

The functional diversity of the ribosome dataset also translates into fairly diverse group definitions. This in turn leads to a slightly larger proportion of groups (6 out of 17) whose definition does not fully represent the group content (see Table 8.6), compared to the proteasome dataset (4 out of 13). In some cases, this even raises the question whether the semantic threshold might be too low and that a slightly higher ST might be appropriate. A particularly good example of this is supergroup

Group ID	Biologically relevant?	Annotation issues	Potential algorithmic improvements
103	Yes		Refinement of definition based on group content
1009	Yes		
1056	Yes		
1045	Yes		
1044	Yes		Refinement of definition based on group content
102	Yes		Refinement of definition based on group content
104	Yes		Refinement of definition based on group content
1025	Yes		
1013	Yes		
1051	Yes		
1016	Yes	RPA49, RPC19	
1036	Yes		
1050	Yes		
1035	Yes		Refinement of definition based on group content
1039	Yes		
101	Yes		Refinement of definition based on group content
1052	Yes		

Table 8.6: Summary of all meaningful groups for the ribosome dataset. All groups were found to be biologically relevant in the context of the ribosome dataset. The same potential algorithmic improvements already identified in the previous dataset was found again. The ribosome groups are affected by notably fewer annotation issues than the proteasome dataset.

104, named ribosome biogenesis, which has 16 definition terms that cover a range of diverse aspects of ribosome biogenesis. However, while using the maximum ST would lead to a larger number of groups with higher specificity, almost two thirds of the groups covering any of the aspects of ribosome biogenesis contain fewer than 4 genes and would therefore be excluded from the analysis.

No further potential algorithmic improvements were identified with respect to the ribosome dataset. A full analysis of each group in Table 8.5 can be found in Appendix B.2.

Annotation issues

Unlike the proteasome dataset, the ribosome dataset has no major annotation issues. The only minor issue in the ribosome dataset was identified from the exclusion of RPA49 and RPC19 from group 1016, which is identical in content to group 1013 except for the two genes in question. The two groups also share the same name, which is also the one term common in their respective two-term definitions. RPA49 and RPC19 are included in group 1013 because they are annotated with that group's distinct definition term, DNA-directed RNA polymerase I complex, although neither gene is annotated with the term nucleolus. This is the reason for their exclusion from group 1016, in which DNA-directed RNA polymerase I complex has been replaced with the term box C/D snoRNP complex. In the latest version of the GO (2011-01), the two genes in question have been annotated with the term nucleolus, under the IEA evidence code. If this version of the GO was used, they would therefore be included in the group. In fact, this would qualify groups 1013 and 1016 for merging into a supergroup.

8.3 Pathway identification

8.3.1 Gene selection

In previous sections, the capacity of the FuSiGroups algorithm to reflect the functional similarity of genes with similar expression profiles was tested. This section is going to focus on a different form of classification, namely genes that are considered to be part of a common pathway.

For this purpose, two "superpathways" were selected from SGD's pathways database[3]. They were the "superpathway of TCA cycle and glyoxylate cycle" and

[3]Downloaded from the SGD's FTP at http://downloads.yeastgenome.org/literature_curation/ on 01/06/2010

"phosphatidic acid and phospholipid biosynthesis". The former includes, as the name suggests, the TCA cycle and the glyoxylate cycle, while the latter includes phosphatidic acid biosynthesis, phospholipid biosynthesis (Kennedy pathway) and phospholipid biosynthesis.

Choosing two sets of metabolic pathways allows study of the sensitivity of FuSiGroups to distinguish between genes from related pathways and from different sets. The citric acid or TCA cycle and the glyoxylate cycle are two highly similar pathways which have a number of reactions and enzymes in common but take place in different parts of the cell [Berg et al., 2002; Regev-Rudzki et al., 2005]. The TCA cycle, found in all cells that operate under aerobic conditions, consists of a series of biochemical reactions that play a crucial part in the process of turning fuel molecules into energy. In eukaryotic cells these reactions take place in the matrix of the mitochondria. The glyoxylate cycle fulfils a similar role, but is only found in certain bacteria, plants and fungi. It bypasses the two decarboxylation steps of the TCA cycle and uses an additional acetyl CoA molecule per cycle. The reactions do not take place in the mitochondria but in the cytosol. Figure 8.7 shows the full superpathway.

Due to the considerable overlap between the two pathways, it can be expected that FuSiGroups will generate BP and possibly MF groups containing genes from both pathways. CC groups on the other hand should only contain genes from one pathway, including those shared between the two, but not genes from both pathways that are unique to one of the pathways.

The three component pathways of the phosphatidic acid and phospholipid biosynthesis superpathway are involved in the synthesis of phospholipids, a major component of cell membranes [Berg et al., 2002; Carman and Henry, 1989]. Related pathways that are not included under this superpathway in the SGD database are phospholipid biosynthesis II (Kennedy pathway), CDP-diacylglycerol biosynthesis and phosphatidylinositol biosynthesis. These pathways are however associated with the superpathway in question in the online biochemcial pathways resource on the SGD website, while the actual phosphatidic acid biosynthesis pathway is in fact excluded from the superpathway despite its inclusion in the superpathway name (see Figure 8.8). There is no apparent reason for the difference between the two resources, so it was decided to adopt the classification used in SGD's downloadable literature curated database, not only because this resource is machine processable, but also because it is more up-to-date than the online pathway, which has still not been manually curated at the time of writing.

Although the phosphatidic acid and phospholipid biosynthesis superpathway is split into multiple subpathways, these pathways are not truly stand-alone pathways,

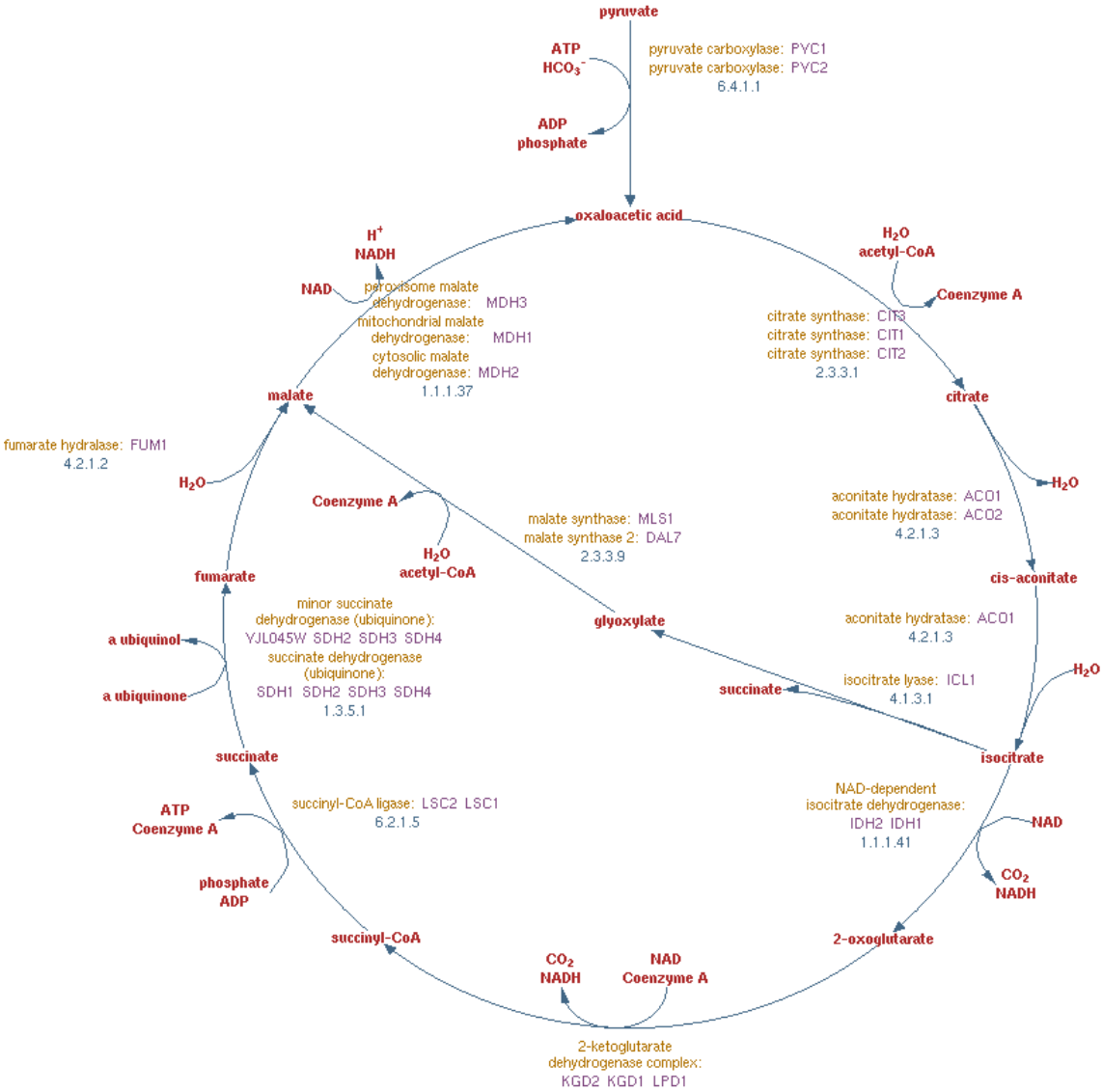

Figure 8.7: Superpathway of TCA cycle and glyoxylate cycle, taken from the SGD biochemcial pathways resource.

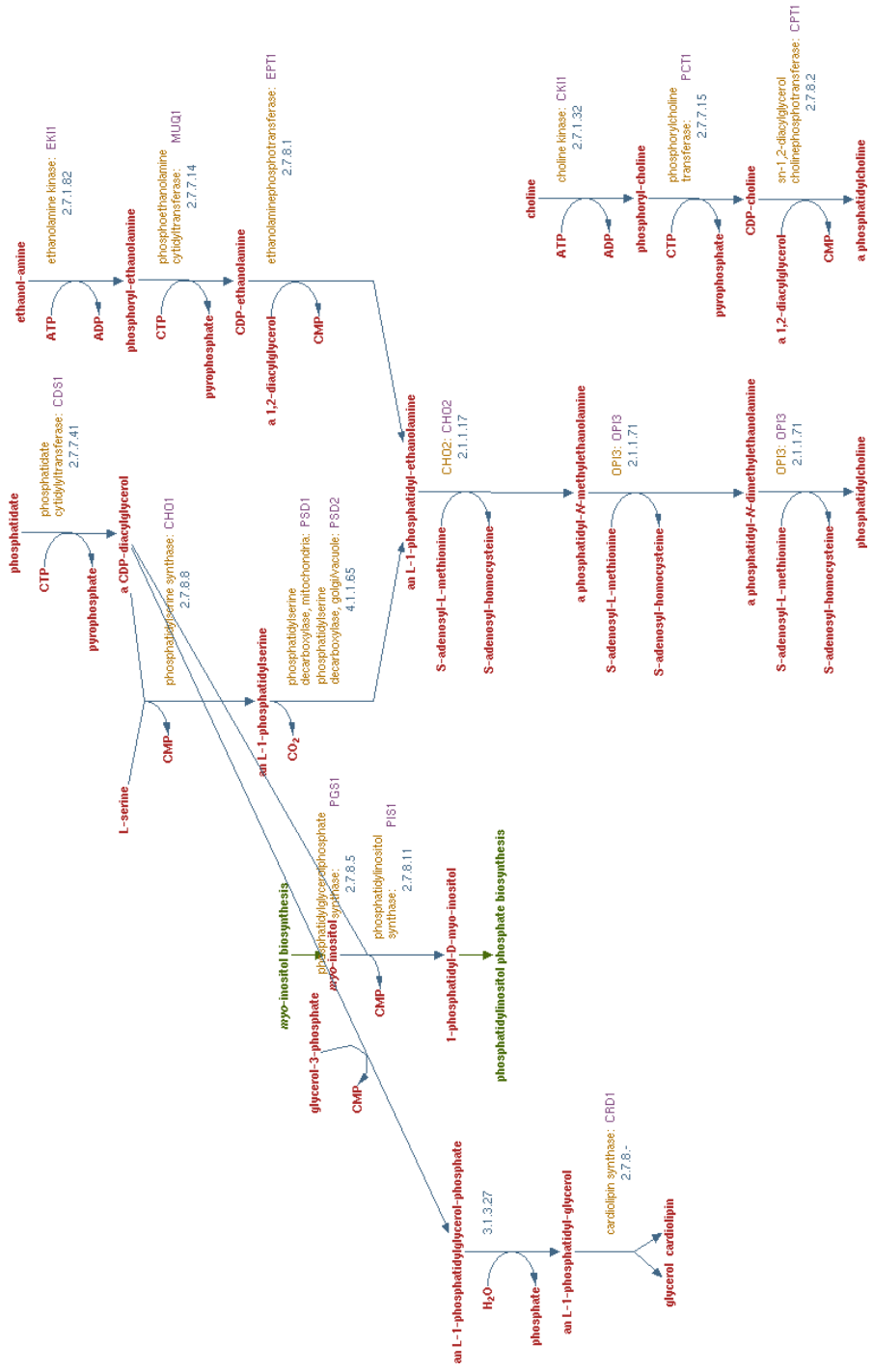

Figure 8.8: Superpathway of phosphatidic acid and phospholipid biosynthesis, taken from the SGD biochemcial pathways resource. It should be noted that this superpathway representation includes some subpathways not specifically listed for this superpathway in the downloadable SGD pathway database. These include phosphatidylinositol biosynthesis, phospholipid biosynthesis II (Kennedy pathway) and CDP-diacylglycerol biosynthesis, while phosphatidic acid biosynthesis is in fact covered by the superpathway of phosphatidate biosynthesis, with only the final product of the pathway, phosphatidate, included here as the starting point to the superpathway. The reason for these differences is that the two resources, despite both being provided by SGD, are created in different way. It should also be noted that unlike the pathway in Figure 8.7, this pathway has not yet undergone manual curation.

nor are they even as different as the TCA and glyoxylate cycle, which occur in different parts of the cell. Rather, the phospholipid subpathways are a series of biochemical reactions which feed into each other or run in parallel, creating the same product from different substrates. For this reason, it is expected that most groups will not differentiate between the subpathways.

There is no documented direct link between the two superpathways, i.e. no part of either superpathway feeds into the other. Indirectly, the pathways are linked within the complex network of biochemical reactions in the cell, with by-products of one pathway feeding into other pathways. This should however not have any noise effect on the grouping and it is expected that there should be no or very few groups containing genes from both sets of pathways. The ontological branch most likely to generate groups with overlap between the superpathways is CC, due to the fact that in yeast phospholipid biosynthesis occurs not only in the endoplasmic reticulum, as it does in mammals, but also in mitochondria [Cobon et al., 1974; Kuchler et al., 1986], the organelle in which the TCA cycle takes place.

All the genes from the full Eisen dataset associated with the two superpathways were selected for inclusion in the pathways dataset. The genes, 34 in total, are listed in Table 8.7. The genes represent 10 of 11 genes listed by SGD for the glyoxylate cycle, 21 of 22 genes for the TCA cycle, 1 of 6 genes for phosphatidic acid biosynthesis, 2 of 3 for phospholipid biosynthesis (Kennedy pathway) and 7 out of 7 for phospholipid biosynthesis. 7 genes are shared between the TCA cycle and glyoxylate cycle.

In KEGG, these genes can be found in three pathway listings, namely the TCA cycle (sce00020), Glyoxylate and dicarboxylate metabolism (sce00630) and Glycerophospholipid metabolism (sce00564). The allocation of genes to pathways is done slightly differently in KEGG, so that these three pathways contain genes found in the Eisen dataset that were not included in the pathway dataset here. The discrepancies are listed in Table 8.8. KEGG also includes phospholipid biosynthesis II (Kennedy pathway) and phosphatidylinositol biosynthesis in the greater glycerophospholipid metabolism.

The pathways dataset has 456 distinct annotations (gene-GO term pairs), with 169 distinct GO terms.

In the selection of the datasets discussed in Section 8.1 and Section 8.2, the genes' location in the expression cluster dendrogram was an important part of the selection process. The genes in the pathways dataset are fairly evenly spread out across both the expression dendrogram and the semantic cluster dendrogram, as the locations listed in Table 8.7 show. There are a few sets of two to five genes in each

Expr. tree	Sem. tree	SGD ID	Gene name	Full name
8	2295	S000000598	CIT2 ±∓	Citrate synthase, catalyzes the condensation of acetyl coenzyme A and oxaloacetate to form citrate, peroxisomal isozyme involved in glyoxylate cycle
186	2011	S000003476	LSC2 ∓	Beta subunit of succinyl-CoA ligase, which is a mitochondrial enzyme of the TCA cycle that catalyzes the nucleotide-dependent conversion of succinyl-CoA to succinate
201	2129	S000001876	LPD1 ∓	Dihydrolipoamide dehydrogenase, the lipoamide dehydrogenase component (E3) of the pyruvate dehydrogenase and 2-oxoglutarate dehydrogenase multi-enzyme complexes
206	2162	S000001631	SDH1 ∓	Flavoprotein subunit of succinate dehydrogenase (Sdh1p, Sdh2p, Sdh3p, Sdh4p), which couples the oxidation of succinate to the transfer of electrons to ubiquinone
333	478	S000002236	MDH3 ±∓	Peroxisomal malate dehydrogenase, catalyzes interconversion of malate and oxaloacetate
406	2120	S000005486	MDH2 ±∓	Cytoplasmic malate dehydrogenase, one of three isozymes that catalyze interconversion of malate and oxaloacetate
409	2075	S000006183	FUM1 ∓	Fumarase, converts fumaric acid to L-malic acid in the TCA cycle
410	2112	S000004982	IDH1 ∓	Subunit of mitochondrial NAD(+)-dependent isocitrate dehydrogenase, which catalyzes the oxidation of isocitrate to alpha-ketoglutarate in the TCA cycle
411	2113	S000005662	IDH2 ∓	Subunit of mitochondrial NAD(+)-dependent isocitrate dehydrogenase, which catalyzes the oxidation of isocitrate to alpha-ketoglutarate in the TCA cycle
442	2297	S000005061	MLS1 ±	Malate synthase, enzyme of the glyoxylate cycle, involved in utilization of non-fermentable carbon sources
443	2063	S000000867	ICL1 ±	Isocitrate lyase, catalyzes the formation of succinate and glyoxylate from isocitrate, a key reaction of the glyoxylate cycle
663	2298	S000001470	DAL7 ±	Malate synthase, role in allantoin degradation unknown
1224	2010	S000005668	LSC1 ∓	Alpha subunit of succinyl-CoA ligase, which is a mitochondrial enzyme of the TCA cycle that catalyzes the nucleotide-dependent conversion of succinyl-CoA to succinate
1232	2296	S000005284	CIT1 ±∓	Citrate synthase, catalyzes the condensation of acetyl coenzyme A and oxaloacetate to form citrate
1245	2130	S000001387	KGD1 ∓	Component of the mitochondrial alpha-ketoglutarate dehydrogenase complex, which catalyzes a key step in the tricarboxylic acid (TCA) cycle, the oxidative decarboxylation of alpha-ketoglutarate to form succinyl-CoA
1262	2087	S000004295	ACO1 ±∓	Aconitase, required for the tricarboxylic acid (TCA) cycle and also independently required for mitochondrial genome maintenance
1267	2159	S000002585	SDH4 ∓	Membrane anchor subunit of succinate dehydrogenase (Sdh1p, Sdh2p, Sdh3p, Sdh4p), which couples the oxidation of succinate to the transfer of electrons to ubiquinone
1269	2119	S000001568	MDH1 ±∓	Mitochondrial malate dehydrogenase, catalyzes interconversion of malate and oxaloacetate
1270	2161	S000003964	SDH2 ∓	Iron-sulfur protein subunit of succinate dehydrogenase (Sdh1p, Sdh2p, Sdh3p, Sdh4p), which couples the oxidation of succinate to the transfer of electrons to ubiquinone
1273	2160	S000001624	SDH3 ∓	Cytochrome b subunit of succinate dehydrogenase (Sdh1p, Sdh2p, Sdh3p, Sdh4p), which couples the oxidation of succinate to the transfer of electrons to ubiquinone
1752	2288	S000002555	KGD2 ∓	Dihydrolipoyl transsuccinylase, component of the mitochondrial alpha-ketoglutarate dehydrogenase complex, which catalyzes the oxidative decarboxylation of alpha-ketoglutarate to succinyl-CoA in the TCA cycle
2049	2294	S000006205	CIT3 ±∓	Dual specificity mitochondrial citrate and methylcitrate synthase
2264	1760	S000000422	PYC2 ∓	Pyruvate carboxylase isoform, cytoplasmic enzyme that converts pyruvate to oxaloacetate
2265	1761	S000003030	PYC1 ∓	Pyruvate carboxylase isoform, cytoplasmic enzyme that converts pyruvate to oxaloacetate
134	2397	S000003389	CHO2 †	Phosphatidylethanolamine methyltransferase (PEMT), catalyzes the first step in the conversion of phosphatidylethanolamine to phosphatidylcholine during the methylation pathway of phosphatidylcholine biosynthesis
426	2390	S000002301	CRD1 †	Cardiolipin synthase
432	2396	S000003834	OPI3 †	Phospholipid methyltransferase (methylene-fatty-acyl-phospholipid synthase), catalyzes the last two steps in phosphatidylcholine biosynthesis
620	2354	S000000510	PGS1 †	Phosphatidylglycerolphosphate synthase, catalyzes the synthesis of phosphatidylglycerolphosphate from CDP-diacylglycerol and sn-glycerol 3-phosphate in the first committed and rate-limiting step of cardiolipin biosynthesis
1098	2065	S000003402	PSD2 †	Phosphatidylserine decarboxylase of the Golgi and vacuolar membranes, converts phosphatidylserine to phosphatidylethanolamine
1099	2391	S000000828	CHO1 †	Phosphatidylserine synthase, functions in phospholipid biosynthesis
1512	2064	S000005113	PSD1 †	Phosphatidylserine decarboxylase of the mitochondrial inner membrane, converts phosphatidylserine to phosphatidylethanolamine

Continued on next page

Continued from previous page				
Expr. tree	Sem. tree	SGD ID	Gene name	Full name
1858	412	S000003434	PCT1 ‡	Cholinephosphate cytidylyltransferase, also known as CTP:phosphocholine cytidylyltransferase, rate-determining enzyme of the CDP-choline pathway for phosphatidylcholine synthesis, inhibited by Sec14p, activated upon lipid-binding
2088	2394	S000005074	CPT1 ‡	Cholinephosphotransferase, required for phosphatidylcholine biosynthesis and for inositol-dependent regulation of EPT1 transcription
2119	2380	S000002210	SLC1 *	1-acyl-sn-gylcerol-3-phosphate acyltransferase, catalyzes the acylation of lysophosphatidic acid to form phosphatidic acid, a key intermediate in lipid metabolism

Table 8.7: The pathways subset. ± - glyoxylate cycle; ∓ - TCA cycle, aerobic respiration; † - phospholipid biosynthesis; ‡ - phospholipid biosynthesis (Kennedy pathway); ∗ - phosphatidic acid biosynthesis

Genename	SGD pathway	KEGG pathway
ERG10	mevalonate pathway	Glyoxylate and dicarboxylate metabolism (sce00630)
IDP1	superpathway of glutamate biosynthesis	TCA cycle (sce00020)
IDP2	superpathway of glutamate biosynthesis	TCA cycle (sce00020)
IDP3	superpathway of glutamate biosynthesis	TCA cycle (sce00020)
LAT1	pyruvate dehydrogenase complex	TCA cycle (sce00020)
PCK1	gluconeogenesis	TCA cycle (sce00020)
PDA1	pyruvate dehydrogenase complex	TCA cycle (sce00020)
PDB1	pyruvate dehydrogenase complex	TCA cycle (sce00020)
EPT1	phospholipid biosynthesis II (Kennedy pathway)	Glycerophospholipid metabolism (sce00564)
MUQ1	phospholipid biosynthesis II (Kennedy pathway)	Glycerophospholipid metabolism (sce00564)
PIS1	phosphatidylinositol biosynthesis	Glycerophospholipid metabolism (sce00564)
PLB1	*not associated with a pathway in SGD*	Glycerophospholipid metabolism (sce00564)
OPI3	phospholipid biosynthesis	*not associated with a pathway in KEGG*

Table 8.8: Discrepancies between gene allocations to different pathways in SGD and KEGG, for genes that are found in the Eisen dataset.

tree that are located in adjacent or very close positions but there is no big cluster nor is there a single particularly coherent subpathway. Tree locations will therefore be included in this dataset for reference only.

8.3.2 Grouping

In the grouping of the full Eisen dataset, the genes of the pathways dataset are never all grouped together, although this is of course to be expected due to the nature of the dataset. The largest grouped set consists of 15 TCA cycle genes but this represents only around a third of the group's full content. There are also no groups which contain only genes from the pathways subset, except for two non-meaningful CC groups, oxoglutarate dehydrogenase complex and tricarboxylic acid cycle enzyme complex. Three genes, ICL1, PGS1 and PCT1 are never grouped. Overall, genes from the pathways dataset are found in 65 original groups, 61 of which contain four or more genes, or 8 supergroups and 20 unmerged groups, 16 of which meet minimum content requirements.

When grouped separately, the pathways dataset produces the 6 supergroups and 47 unmerged groups, of which 18 contain four or more genes, shown in Figure 8.9.

This is a considerable reduction in volume from 169 distinct GO terms and 456 annotations. Prior to merging, there were 60 groups, 31 of which were meaningful. From Figure 8.9, it is immediately clear that with the exception of two groups, there is no overlap between the genes from the TCA cycle and glyoxylate superpathway and the genes from the phospholipids pathways. The two exceptions are CC supergroup 103 and MF group 1033. On closer analysis, both of these groups are revealed to be biologically appropriate and the genes in the group do indeed all share the cellular location or molecular function reflected by the respective groups.

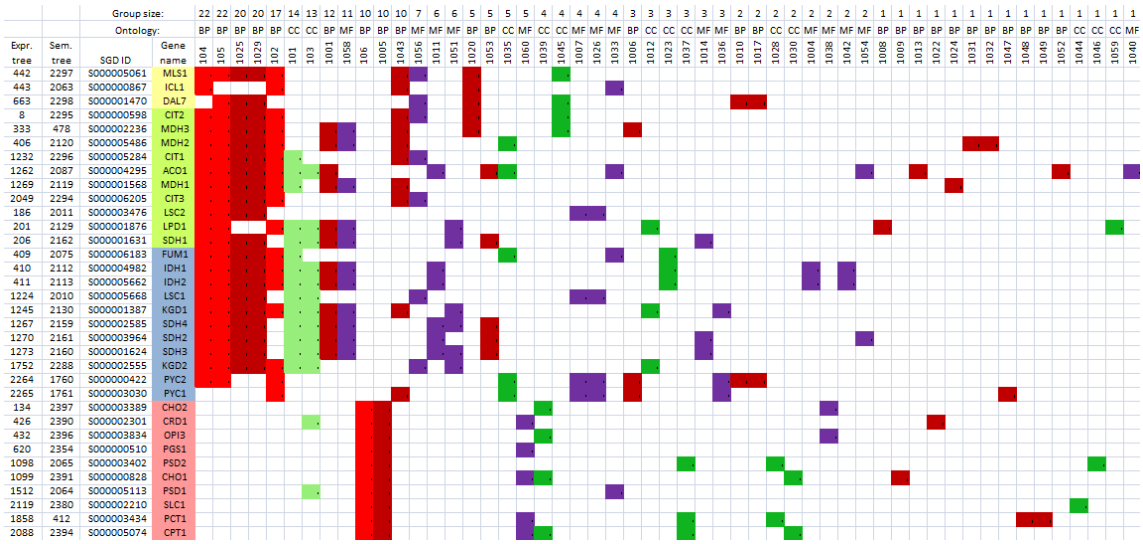

Figure 8.9: Supergroups and unmerged groups for the pathways dataset, ordered left to right by size and top to bottom by the pathway, then by their location in the expression tree, with the following pathway order: glyoxylate cycle genes, genes active in both glyoxylate and TCA cycle, TCA cycle genes, phospholipid biosynthesis genes, phosphatidic acid biosynthesis genes and Kennedy pathway genes. In order to illustrate the mapping of the genes in the groups to their original pathways (shown in Figures 8.7 and 8.8), the gene names of glyoxylate cycle specific genes are given in yellow, those of TCA cycle specific genes in blue and those of genes shared by the two pathways are in green, while the genes from the phospholipids pathways are coloured in red. Other colour conventions are the same as in Figure 8.3.

As in the previous two datasets, the largest groups in Figure 8.9 are clearly very similar. This is confirmed through the group names in Table 8.9, which shows that the two largest supergroups do indeed have the same fairly generic name. They are BP supergroups 104, and 105. Supergroup 102, although slightly smaller, also shares this generic name.

The potential algorithmic improvement identified in Sections 8.1 and 8.2 is also relevant for the pathway groups, as summarised in Table 8.10 and is as prevalent as in the other two dataset. No new potential algorithmic improvements were identified as part of this analysis.

None of the group names in Table 8.9 are obviously inconsistent with the func-

Group ID	Group name	Ontology	Group size
104	cellular metabolic process *(GO:0044248)*	BP	22
105	cellular metabolic process *(GO:0044237)*	BP	22
1025	generation of precursor metabolites and energy *(GO:0006091)*	BP	20
1029	coenzyme metabolic process *(GO:0006732)*	BP	20
102	cellular metabolic process *(GO:0044237)*	BP	17
101	mitochondrial part *(GO:0044429)*	CC	14
103	intracellular membrane-bounded organelle *(GO:0043231)*	CC	13
1001	oxidation reduction *(GO:0055114)*	BP	12
1058	oxidoreductase activity *(GO:0016491)*	MF	11
106	lipid metabolic process *(GO:0006629)*	BP	10
1005	cellular lipid metabolic process *(GO:0044255)*	BP	10
1043	carbohydrate metabolic process *(GO:0005975)*	BP	10
1056	transferase activity, transferring acyl groups *(GO:0016746)*	MF	7
1011	transition metal ion binding *(GO:0046914)*	MF	6
1051	cofactor binding *(GO:0048037)*	MF	6
1020	cellular aldehyde metabolic process *(GO:0006081)*	BP	5
1035	cytosol *(GO:0005829)*	CC	5
1053	phosphorus metabolic process *(GO:0006793)*	BP	5
1060	transferase activity, transferring phosphorus-containing groups *(GO:0016772)*	MF	5
1007	purine nucleotide binding *(GO:0017076)*	MF	4
1026	ligase activity *(GO:0016874)*	MF	4
1033	lyase activity *(GO:0016829)*	MF	4
1039	endoplasmic reticulum *(GO:0005783)*	CC	4
1045	microbody *(GO:0042579)*	CC	4

Table 8.9: Names of all the meaningful groups and supergroups shown in Figure 8.9, sorted by decreasing size. Three-digit group IDs indicate supergroups, four-digit IDs normal groups.

tional aspects expected to be associated with the genes in the dataset. Some names, such as cellular metabolic process, are too high-level to allow a definite conclusion to be drawn about the biological relevance of the terms in the group definition. A detailed analysis of each group, available in Appendix B.3, confirmed that all the groups are biologically relevant in the context of the pathways dataset.

A few groups, notably supergroups 104, 105 and 102, did however have such diverse definitions that, although they were biologically correct, they provided only limited insight into the functional aspects shared by their content genes. The same does not apply to all large groups, as the definitions of groups 1025 and 1029 are much less diverse and more insightful. If the maximum semantic threshold is used, the groups with very diverse definitions are not found and the largest groups are roughly the same as groups 1025 and 1029 in this analysis.

In terms of the stated goal for the pathways dataset, namely testing FuSiGroups' ability to distinguish between different pathways, the results analysis is positive. Aside from two valid groups which contain genes from both superpathways, all groups contain genes from only one of the two superpathways shown in Figures 8.7 and 8.8. In order to illustrate the mapping of the genes in the groups back to the original pathways, the gene names in Figure 8.9 are colour-coded based on their pathways membership, such as yellow for glyoxylate cycle-specific genes and blue

Group ID	Biologically relevant?	Annotation issues	Potential algorithmic improvements
104	Yes		Refinement of definition based on group content
105	Yes		
1025	Yes	MLS1, DAL7	Refinement of definition based on group content
1029	Yes	MLS1, DAL7	Refinement of definition based on group content
102	Yes		
101	Yes		Refinement of definition based on group content
103	Yes		Refinement of definition based on group content
1001	Yes	ACO1	
1058	Yes		
106	Yes		Refinement of definition based on group content
1005	Yes		
1043	Yes		
1056	Yes		Refinement of definition based on group content
1011	Yes		Refinement of definition based on group content
1051	Yes		
1020	Yes	∗ACO1, MDH2, MDH1, CIT1, CIT3	
1035	Yes		
1053	Yes	ACO1	
1060	Yes		
1007	Yes		Refinement of definition based on group content
1026	Yes		
1033	Yes		
1039	Yes		
1045	Yes		

Table 8.10: Summary of the analysis of all meaningful groups in the pathways dataset. As in previous cases, ∗ indicates that the annotation issue in question led to the exclusion of the genes from a group. No new potential algorithmic refinements were identified for this dataset.

for TCA cycle-specific genes.

The groups that contain only genes from the phosphatidic acid and phospholipid biosynthesis pathway do not distinguish between the individual subpathways. This is unsurprising as it was established earlier that these subpathways are not so much stand-alone pathways as different elements of a large pathway, as can be seen in Figure 8.8.

The TCA cycle and glyoxylate cycle on the other hand are two distinct pathways, not least because of the different cellular locations that they take place in. In terms of the sequence of reactions in the pathway, as can be seen in Figure 8.7, TCA cycle is represented by the full outer circle, while the glyoxylate cycle consists of the upper part of the circle and the reactions represented in the line going across the circle, thus bypassing the lower part of the circle.

Among the pathways groups, there are two groups, group 1020 and group 1045, which contain only genes associated with the glyoxylate cycle. These include three and two genes that are unique to the glyoxylate cycle and two genes that are common to both pathways. The names of these groups, as well as their full definitions, identify them as groups that should indeed only contain glyoxylate cycle-specific genes. The same also applies to groups like group 105 and group 102, which contain only genes from the TCA cycle.

There are no groups that contain all the genes from either the glyoxylate cycle or the TCA cycle, nor are there any groups that contain all the genes from the superpathway. The explanation for this is that some of the genes, particularly ICL1, PYC1 and PYC2, have insufficiently high functional similarity with some of the other genes in the superpathway. It should be noted that from an ontological point of view, it would be impossible for the terms tricarboxylic acid cycle and glyoxylate cycle to appear in the same group definition as these terms are part of different branches of the BP ontology, despite referring to extremely similar pathways.

Annotation issues

There are four genes in the pathways dataset for which annotation issues were identified during the results analysis. The most obvious one of these is ACO1, for which several inappropriate annotations were found in relation to several groups. These include an annotation with the BP term oxidation reduction, qualified with the RCA evidence code, despite the fact that aconitase activity is not a redox reaction [Beinert et al., 1996], as well as the term phosphorus metabolic process, another RCA annotation, which is also inconsistent with aconitase activity. Both of these

annotations are no longer found in the latest version of the Gene Ontology. In addition to these inappropriate annotations, ACO1 also lacks annotation with the term glyoxylate cycle, despite evidence that it is involved in this process. This lack of annotation led to the exclusion of ACO1 from group 1020. The same problem was found for MDH2. Neither gene has acquired this annotation in more recent versions of the GO.

The absence of annotation with the term glyoxylate cycle was also found for genes MDH1, CIT1 and CIT3. These are listed as being shared between the TCA cycle and the glyoxylate cycle. Upon closer examination however, it becomes clear that these three genes are in fact TCA cycle-specific. In this case, the absence of the annotation term was therefore appropriate and it is the association of these genes with the glyoxylate cycle in KEGG and SGD that might be considered questionable.

The two glyoxylate cycle-specific genes MLS1 and DAL7 were both found to be annotated with the term tricarboxylic acid cycle under the IEA evidence code. This annotation, which is most likely the result of "guilt by association" with one of the genes shared between the two pathways, is almost certainly inappropriate but it is still present in the latest version of the GO. It is also the only reason for the inclusion of these two genes in groups 1025 and 1029. In this case, an inappropriate annotation led to an appropriate inclusion of two genes in a group.

8.4 Summary

In this chapter, the results obtained from the FuSiGroups algorithm for three smaller sub-datasets of the Eisen dataset were presented and analysed. Specifically, the ability of FuSiGroups to address a number of specific scenarios was considered. For all three datasets, FuSiGroups correctly identified the main functional aspects of the dataset and apart from a couple of minor exceptions, all groups were found to be biologically relevant. Results for both the proteasome and ribosome datasets showed that the algorithm is able to identify genes that are functionally unrelated to the bulk of the dataset by not including them in any meaningful groups. FuSiGroups groups were found to be consistent with pathway subdivisions, but the algorithm did not consistently reflect the clusters obtained from gene expression analysis of the same genes.

A number of limitations to the FuSiGroups algorithm in particular and semantic and functional similarity in general were identified. These included the accuracy of the semantic threshold, the need to revisit each group definition after group content allocation in order to ensure that the definition fully represents the content genes

and the ability of semantic and functional similarity to accurately capture complex biological relationships.

In the next chapter, the implications of the results presented in this and the four previous chapters will be discussed. An outlook on possible refinements and additions to this work in general and FuSiGroups in particular will also be given.

Chapter 9

Discussion & Conclusion

In Chapters 4 through 8, the results from the different parts of this project were analysed. In this chapter, the meaning and implications of these results will be discussed, both in relation to the project and in the context of the wider research area of functional annotation similarity.

9.1 Semantic and functional similarity approaches

One of the goals of this project was to compare a number of semantic and functional similarity approaches, as well as the associated parameters of ontological score combination, ancestor choice and type of annotation data, in order to establish which ones performed the best. The test dataset was an aggregate dataset consisting of gene product pairs with known similarity (or dissimilarity) in either gene expression, protein interaction or phenotype.

9.1.1 Semantic similarity

Three information content-based semantic similarity measures, by Resnik [1995]; Lin [1998]; Schlicker et al. [2006], and one hybrid measure, making use of both the nodes and the edges of the GO graph, by Wang et al. [2007], were selected. Resnik's and Schlicker's approaches performed the best, with such marginal differences between their performances that it is difficult to establish a clear "best" approach. Lin's and Wang's approaches generally perform less well, with Wang clearly showing the overall worst performance.

9.1.2 Functional similarity

The BMA [Sevilla et al., 2005] and MAX [Couto et al., 2005] approaches were chosen for the comparison of functional similarity approaches. Neither approach clearly outperforms the other. BMA appears to perform marginally better overall but the difference between the two is so minor that no clear conclusion can be drawn

9.1.3 Other parameters

Individual vs. combined scores

The performance of the aggregate rFunSim [Schlicker et al., 2007b] score was compared to that of the individual scores for the three GO ontologies. The reason for this comparison was that a lot of studies use only individual ontological scores rather than comparing gene products based on a single score including all three ontologies. rFunSim was found to outperform individual ontology scores in the majority of cases. The greatest improvement in performance was found compared to the MF ontology, where AUC indices were generally much higher for rFunSim. The largest number of cases where rFunSim performed worse than an ontological score related to CC, although the difference in AUC was always minimal. The most likely explanation for this is that CC is the smallest of the three ontologies and it is therefore easier to obtain a high level of similarity for CC than a high level of similarity overall.

Ancestor choice

The effect of using Couto et al.'s GraSM algorithm for disjunctive ancestor choice instead of the more common most informative common ancestor (MICA) was also considered. Although GraSM might objectively be expected to provide an improvement over MICA as it makes better use of the ontological structure, it was actually found to generally perform worse than the single ancestor approach.

Annotation

Finally, the performance of the different approaches was compared for all annotation data and annotation data excluding electronic annotation. In this analysis, using all annotations led to an overall better performance.

9.1.4 Recommendations

Based on the performance analysis of the different approaches and parameters, the Resnik and Schlicker semantic similarity approaches were selected to be used in FuSiGroups. For functional similarity, both BMA and MAX were selected, while the choices for annotation and ancestor were full annotation and MICA, respectively. The combined rFunSim score was selected over individual ontological scores.

It should however be noted that while these choices were clearly the most appropriate in relation to this work, different choices may be more appropriate under certain circumstances. One example of this is the use of functional annotation. While the better performance of full annotation over non-electronic annotation is not surprising in a yeast dataset, where electronic annotation is fairly high-quality and very abundant, the same may not be true for other species, particularly for a species which has little electronic annotation. It is therefore essential to have a good understanding of the annotation sources for the data being analysed.

Depending on the type of work, it may also be more suitable to use an individual ontological score, despite the significantly better performance of rFunSim in this comparison. A particularly relevant example would be a study of subcellular locations of gene products, in which case limited functional similarity to the CC score would be more appropriate than the combined score.

9.2 FuSiGroups

The second big aspect of this project was the development of a grouping algorithm based on the semantic similarity between GO terms and the functional similarity between the gene products they annotate. The analysis and evaluation of this algorithm was covered in Chapters 5 to 8. First, the experimental parameters for the algorithm, the semantic and functional thresholds, were determined. Then the results of the algorithm for a selection of different variables, such as different semantic and functional similarity measures, were analysed on a high level to establish whether any interesting patterns could be found and the groups for one set of variables were compared to other forms of clustering. Finally, the algorithm's performance was evaluated on three smaller datasets to establish whether it generates biologically relevant results.

9.2.1 Grouping trends

An entire chapter of this thesis, Chapter 6, was devoted to the study of the overall trends of the grouping results for the two semantic and two functional similarity approaches previously found to perform the best out of all the approaches under consideration, at their empirical minimum and maximum semantic and functional thresholds. While this may seem like an unnecessary step before the analysis of the actual groups, considering overall trends such as the number of groups produced by a given set of parameters or the sizes of the group definitions is in fact an essential part of evaluating the performance of the algorithm. This is particularly true as analysing the individual groups for each set of parameters would be extremely laborious and repetitive.

By analysing a number of factors, including the number of total and meaningful groups, the maximum and average group sizes and the maximum and average group definition sizes, it was possible to establish that the semantic thresholds derived for Schlicker's approach were most likely inappropriately high. Indicators for this were the much larger number of groups generated for Schlicker compared to Resnik and the associated much smaller group definition sizes. Due to the nature of the algorithm, where group definitions that are subsets of other definitions are removed, a large number of groups with very small definitions indicates that very few groups have been discarded, which in turn suggests that the similarity threshold responsible for group definitions only allows very few GO terms to be grouped together. Through the study of the grouping trends, it was therefore possible to eliminate Schlicker's approach from the more detailed analysis and focus on Resnik's approach, which was most likely to generate useful results.

Another factor that could be established from the analysis of grouping trends was the lack of correlation between group sizes and definition sizes, as well as between definition sizes and the ontological depth of the definition terms. A strong correlation in either category would have been an indicator of bias in the algorithm. In the former case, it would have indicated that the number of GO terms in a definition affects the number of genes in the group, which would mean that groups with many genes can only exist due to functionally diverse definitions, rather than because these genes are genuinely related based on a focussed functional aspect. A correlation between definition size and ontological depth would have suggested a strong bias in the semantic similarity approach in question, with deeper (for a positive correlation) or shallower (for a negative correlation) terms scoring disproportionately high similarity scores. This in turn would mean that more terms of that ontological depth

would be grouped together, leading to larger definitions. Since no such correlation was found, the absence of bias in the semantic similarity approaches was confirmed.

9.2.2 Grouping vs. clustering

A large part of the analysis was initially targeted to be a comparison of expression clusters, semantic clusters and functional groups. It soon turned out however that there was little consistent overlap between these three types of groupings. External cluster validation techniques performed poorly across a range of clustering thresholds, for both expression and semantic clusters.

Both the full Eisen dataset and the smaller datasets showed little consistency between FuSiGroups groups and clusters of either type. One exception to this is the proteasome dataset, in which the genes are mostly clustered together in both the expression and the semantic tree. The expression clustering is obviously very consistent here as that is how the dataset was selected, but the semantic clustering is also surprisingly good, although not quite as good as the grouping. The same also applies to the expression clustering in the ribosome dataset, as the ribosome dataset was selected from a number of distinct but closely related expression clusters. The semantic clustering of this dataset, as well as both types of clustering in the pathways dataset, are on the other hand pretty much "all over the place". Only small sets of two to five genes were found in the same area of a clustering tree and there was little consistency across the two trees, so a set of genes clustered together in one tree would not be clustered in the other tree.

These findings are consistent with studies of the correlation between functional and expression similarity [Sevilla et al., 2005; Wang et al., 2004; Xu et al., 2008], which generally find correlation between the two types of similarity only if similarities are averaged across intervals, but not if pair by pair correlation is used. As all forms of grouping used in this study are based on the similarity between pairs of gene products, it is not entirely unsurprising to find that clustering approaches based on the different measures do not compare well.

9.2.3 Grouping results

FuSiGroups does not perform particularly well with very large datasets as they contain lots of noise. As a result, groups may not actually be particularly functionally coherent. This is clearly illustrated in the fact that the genes from each of the three smaller evaluation datasets always show poor grouping results in the full dataset despite clearly generating good functional groups if considered individually. In terms

of scale, test datasets ranged from 2465 genes for the large dataset to 34, 42 and 49 genes for the small datasets. Analysis of one or more intermediate size datasets (100-500 genes) would have been useful to study at what point the effect of noise begins to affect the quality of the groups but this was not possible due to time constraints.

The three smaller datasets, discussed in Chapter 8, were chosen to address a number of questions, including FuSiGroups' ability of identify the main functional aspects of a dataset, generate biologically relevant groups, identify unrelated "noise" genes in a dataset and how groups compare to other forms of biological classifications such as expression clusters or pathways. Based on the analysis of the grouping results for the three datasets, all of these questions were successfully addressed.

For each dataset, the key functional aspects were identified (Tables 8.2, 8.5, 8.9) correctly. Only in the proteasome dataset were any functional aspects not related to the central part of the dataset translated into a group. The reason for these inappropriate groups was a set of poor annotations. In a few cases, the group name, defined as the lowest common ancestor of all the terms in a group definition, was very high level and did not provide any obvious insight as to the functional aspect represented by that group, but if the entire group definition was taken into consideration, this issue could be resolved. Additionally, all groups, aside from the few proteasome groups already mentioned, were found to be biologically relevant. This means that it is appropriate, based on current understanding from published research, for the genes in the group to be grouped together under the functional aspect reflected by the group definition.

Both the proteasome and the ribosome datasets contained a number of genes that were not directly related to the majority of genes in the dataset. In both cases, FuSiGroups successfully "identified" these genes in the sense that they were only grouped in groups with fewer than four genes, so-called non-meaningful groups.

FuSiGroups also managed to successfully separate the two superpathways in the pathways dataset, with only two groups containing genes from both sets of pathways. In both cases, the "cross-over" groups were found to reflect a function or location which is indeed applicable to genes from both superpathways. Within the TCA cycle and glyoxylate cycle superpathway, there was some separation of genes by individual pathway, as well as a number of groups containing genes from both patwhays. From the analysis of the groups, it was found that in many cases, the annotation of the genes unique to one of the two pathways was too similar to that of the genes common to both to produce a clear differentiation between the two pathways. No differentiation was found between the three sub-pathways of the phosphatidic acid

and phospholipid biosynthesis superpathway. This result can be explained by the close relationship between these subpathways.

Very little successful differentiation was found between the four clusters that make up the ribosome dataset. The majority of the groups contained genes from multiple clusters, with only two of the analysed groups containing genes from just one cluster, although not all the genes from that cluster. The most successful differentiation was between the closest three clusters and the fourth, less related cluster, although again, not all genes from the closest three clusters were grouped together. As correlation between functional and expression similarity was found to only be significant for averaged similarity of one type across intervals of the other in other studies, such as Wang et al. [2004] and Sevilla et al. [2005], this result is not unexpected. It is also consistent with the comparison of grouping and clustering for the full Eisen dataset, discussed in Chapter 7 and Section 9.2.2 above. It might have been of interest to study the performance of FuSiGroups on a fourth dataset, consisting of gene products from two or three clusters from distinct sections of the expression tree, but this was impossible due to time constraints.

Based on these findings, the ideal use for FuSiGroups would be on smaller datasets, such as individual clusters of genes from high-throughput experiments. From these, it could highlight the main functions common to most of the genes and eliminate genes that are functionally unrelated to the rest of the cluster.

Another feature that was repeatedly found in the detailed analysis was inappropriate annotations of gene products, particularly annotations with the RCA evidence code. This does however require detailed analysis of groups, so FuSiGroups would not be appropriate for deliberate large-scale investigations of wrong annotations. It could however help to identify poor annotations of individual datasets in an ad hoc fashion.

The analysis of group definitions showed that in some cases, definitions can be very diverse, too diverse in fact to identify a clear functional aspect from them. This brings into question the appropriateness of the value for Resnik's minimum semantic threshold, which was used for most of the evaluation. Comparison to the results at Resnik's maximum semantic threshold showed that the higher threshold generally reduced the diversity of the definition, although in some cases, the narrower group definitions led to more non-meaningful groups. From the analysis of grouping trends in Chapter 6, it was shown that Schlicker's semantic thresholds were probably too high as they led to a much higher number of groups than Resnik's measure, as well as much smaller group definitions at an average size of less than two GO terms.

From these two aspects of the analysis, it is possible to conclude that the ap-

proach for determining semantic thresholds may not be as successful as the equivalent approach for the functional thresholds. Much lower variation in the effect of the functional thresholds of Resnik and Schlicker was found in the sense that there was much less difference in average group sizes between Resnik and Schlicker than there was in average definition size. From the detailed groups analysis, no cases were identified that would suggest that the FT might be inappropriate.

9.2.4 FuSiGroups compared to other approaches

The key difference between FuSiGroups and other approaches is that it makes use of both the similarity between annotation terms and the similarity between gene products. The closest comparable tool currently available is DAVID [Huang et al., 2007], which does however have a number of crucial differences. First of all, DAVID's fuzzy functional clustering algorithm, although very similar to FuSiGroups, works only in one direction. Either gene products are clustered based on their functional similarity or annotation terms are clustered based on their similarity.

In the case of gene functional clustering, the annotations common to a set of functionally similar groups are presented all together, ranked using enrichment analysis, and not separated by categories. In FuSiGroups on the other hand, the GO terms in a given group of gene products are also all related. Each group represents one functional aspect, so there may be multiple groups with the same or similar gene products, reflecting multiple functional aspects these genes share. Several examples of this were found in the three evaluation datasets in Chapter 8.

For DAVID's functional annotation clustering, the same overall situation applies. In a given cluster of similar annotation terms, all gene products in the dataset associated with any of the clustered terms are returned, regardless of whether they are functionally similar. FuSiGroups on the other hand ensures that all the gene products associated with a given group definition are also functionally similar.

FuSiGroups effectively represents the two dimensions of term and gene product similarity at the same time. It could be argued that this is only relevant for a specific sub-set of the applications of DAVID as it might be desirable to only consider the similarity in one direction. There are however a number of existing approaches, not least DAVID, that fulfil this purpose.

Additionally, DAVID makes use of multiple forms of functional annotation, not just GO. Although this may at first glance seem to be an advantage compared to only using GO, since more information is provided, there appears to be no study evaluating whether it is appropriate to use a large number of different data sources

and treat them equally. A particular concern here is data circularity as some types of annotations are derived from others. While the selection of the data sources in DAVID is a user choice and can therefore be adjusted to address the needs of a given analysis, an understanding of the detailed connections between different types of data sources may not be part of the user's expertise. Using only the GO, rather than a wide variety of annotation sources, in a functional analysis is therefore not necessarily a disadvantage, until a more detailed analysis of the interaction and interdependence of different sources, and the effect of these on functional similarity, have been carried out.

9.2.5 Analysis pathway

In Section 7.1, it was mentioned that there are two angles from which an analysis of FuSiGroups results could be started, namely from the largest groups or the most common group names, i.e. the most commonly represented functional aspects. The following is the analysis pathway used for the analysis in Chapter 8 and which was established as the most efficient approach to analyse the grouping results. It should be noted that this is not the only possible approach.

For each dataset, the coloured group-gene matrix, such as that in Figure 8.3, was created first, and the matrix sorted according to the most appropriate parameters. Left to right, the most appropriate sort is considered to be decreasing group size. Top to bottom, the sorting criteria were more varied. If the dataset is derived from one or more clusters, it might be appropriate to sort the genes by their location in the original cluster tree. This was the approach taken for the ribosome dataset. In the pathways dataset, the genes were sorted according to their membership in different pathways, while in the proteasome dataset, the genes were sorted first by one of three categories (KEGG proteasome genes, other proteolytic enzymes and unrelated genes), then by semantic dendrogram location within each category. The nature of the dataset should dictate the most appropriate parameter, or a number of options can be tested.

The group-gene matrix shows the overall coherence of the groups, as well as the level of duplication between the groups, if any. From the matrix and the list of group names, groups of interest can be identified for further detailed analysis.

For each group of interest, the first analysis step involves looking at the group definition. This can reveal whether a group might be too functionally diverse to be truly useful, but it can also provide a better insight into the functional aspect represented by the group than the group name alone does. Analysing the group

definition also includes considering how many of the group's genes are annotated to each term. Through this, the relationship between the definition terms and the group genes is established.

Next, the genes themselves are considered. It is established if any of them are annotated with more than one definition term. If this is the case for multiple genes, it may indicate a particularly strong functional similarity between the genes. When considering the genes, some level of understanding of these genes is required to judge whether the grouping of these genes is surprising or not. If there are genes whose grouping is unexpected, in particular in relation to the functional aspect of the group, it is advisable to consider the type of relationship between the genes and the GO terms in terms of evidence codes. From these, the source of the annotation can be derived and studied to discover whether the annotation and, by extension, the inclusion of this gene in the group, is appropriate.

The third step in the analysis involves finding any genes that could have been included in the group based on their annotations but were not. This is a necessary step as there are cases when genes are too functionally diverse to have appropriate levels of functional similarity to be grouped together by FuSiGroups, yet it may still be interesting to know about these genes.

It could be argued that this part of the pathways demonstrates that FuSiGroups is not an improvement over DAVID, where similar annotation terms are clustered together and all genes associated with these annotations are also associated with the cluster. It is then up to the user to determine which genes are of interest and which are not. FuSiGroups however has the advantage of first grouping related terms and genes and it is only in the course of the analysis that further genes associated with the definition terms but not included in the group are considered, rather than having to perform the exclusion step manually, based on judgement. Instead of an *a priori* information overload, additional information can be sought out if this is deemed appropriate during the analysis.

9.3 Future work

While the work described in this thesis addresses the research question posed in Section 1.1 and demonstrates the feasibility and appropriateness of functional grouping, it raises a number of additional questions that it would be interesting to address.

9.3.1 Algorithmic work

First of all, one potential algorithmic refinement was identified from the evaluation in Chapter 8. Currently, certain group definitions contain terms that are not found in the annotations of the genes in that group. This situation can be relatively easily addressed by adding an additional step to the grouping algorithm that double-checks the GO terms in the group definition against the annotations of the genes in the group after group content allocation. GO terms not found in the annotations of the group content can then be removed, increasing the functional coherence of the group definition.

This refinement may indirectly reduce the occurrence of high-level group names as many groups found to be affected by this situation also had very high-level group names. The removal of irrelevant definition terms may result in a more specific group name if the removed terms were less semantically similar to the other definition terms than these are to each other. Additionally, this change in group definitions may also affect the number of supergroups as it might reduce the overlap between some group definitions below the merging threshold. Preliminary incorporation of the refinement into the algorithm showed that it has no major effect on the results but does indeed lead to more specific group names.

A second algorithmic modification that was not identified during the results analysis but that might nonetheless be of interest is the use of three semantic thresholds, one for each aspect of the GO, instead of a single one. Since the semantic thresholds for both Resnik and Schlicker were identified as limitations to the optimal function of FuSiGroups, using three different thresholds is a potential option for addressing this issue.

9.3.2 Data sources

There are several additional aspects that it would be interesting to test the FuSiGroups algorithm on, in particular after the modifications proposed in Section 9.3.1. Of particular interest would be using a newer version of the GO, as it was found that most annotation issues identified in the evaluation had been addressed since the release of GO used in this work (2009-04). One or more datasets of intermediate size, between 100 and 500 gene products could address the question at which dataset size the noise from unrelated gene products and generic, high-level annotations becomes too strong to draw meaningful conclusions from the grouping results.

As all testing of the FuSiGroups algorithm so far was done using a well-studied yeast dataset, or subsets thereof, it would also be of interest to study a dataset

from a different species in order to establish whether this has a substantial effect on performance. Ideally, datasets from at least one well-studied species and one less well-studied one should be used to also assess if a less comprehensive level of functional annotation is a considerable drawback. In addition to testing the FuSiGroups algorithm itself on different species, it would also be useful to study the effect these different species have on the performance of the different semantic and functional similarity approaches, as well as on the semantic and functional thresholds. Similar results to those obtained here, particularly for another well-studied organism such as *C. elegans* or mouse, would demonstrate the robustness of these measures and their indifference to the variable nature of a dataset. This is an important consideration as information content semantic similarity measures, such as the majority of the measures used in this work, are dependent on annotation frequencies in the corpus underlying a given analysis.

Different ontologies

The recent proliferation of biological ontologies raises the question whether it might be of interest to include other ontologies in the FuSiGroups analysis. This would first of all require the identification of appropriate ontologies and a study of the way they are used in annotation. Measures such as Resnik's and Schlicker's may only be suitable for use with multiple ontologies if the annotation can be treated in the same way as GO annotation. Extensions or modifications of existing approaches might be necessary to accommodate the use of multiple ontologies. Interesting starting points for this might be the work by Bodenreider et al. [2005] and Huang et al. [2007], as both these approaches allow the use of multiple data sources.

9.3.3 Semantic and functional similarity

In Section 2.2, a wider range of semantic similarity approaches than those studied in this thesis were discussed. It would be interesting to extend the comparison of the performance of different semantic similarity approaches to include more node-based approaches. This applies particularly to the recent approach by Herrmann et al. [2009], published after the selection of approaches to be included in this work had been completed, and which defines a form of information content that is independent of annotation frequencies, the biggest barrier to efficient cross-species or cross-data source functional similarity comparisons.

Equally, it might be useful to include certain of the edge-based semantic similarity approaches in the comparison, to see how they compare in particular to Wang's

hybrid measure, which performed badly compared to the IC-based measures, but also to the node-based measures in general.

Finally, only pair-wise functional similarity measures that use the semantic similarity between GO terms were considered here due to the nature of the FuSiGroups algorithm. For the part of this work that compares the performance of different approaches however, it might be interesting to include also some of the group-wise functional similarity approaches discussed in Section 2.3.

9.3.4 Benchmark dataset

The semantic thresholds were identified as one of the limitations of the FuSiGroups algorithm, since issues were identified with these thresholds for both Resnik and Schlicker. As no similar issues were found to affect the respective functional thresholds, it was concluded that while the overall approach, using true positive and true negative datasets, was appropriate, the specific approach used for the semantic similarity, namely considering the similarity for MAX for each ontological aspect, did not work as well as expected. The evaluation of semantic similarity measures is a known issue in the field [Pesquita et al., 2009], as there is no objective benchmark against which the similarity between GO terms can be evaluated. Designing and testing such a benchmark dataset, a laborious task deemed too time-consuming to be included in this project, would therefore be of great use to the entire field of semantic similarity in the Gene Ontology.

9.4 Conclusion

During the course of this thesis, the questions laid out in Section 1.1 were addressed. The work consisted of two larger aspects, the comparison of a number of semantic and functional similarity approaches and other parameters, and the design, proof-of-concept implementation and testing of a novel grouping algorithm which makes use of semantic and functional similarity approaches to find shared functional aspects of a set of gene products. Both these objectives were addressed successfully in this work.

- The evaluation of semantic and functional similarity approaches and a number of associated parameters was in parts in agreement with literature findings, while it differed in others. Most notably, the semantic similarity approach by Wang et al. [2007] was found to perform worse than Resnik's approach, contrary to the findings by the original authors. The comparison also showed that

the full dataset performed slightly better than the non-electronic one, adding to a growing body of evidence that supports this trend. In many respects, the results showed what is already known from the literature, namely that there are no absolute conclusions possible in current functional similarity research. In fact, absolute conclusions will most likely never be possible. Contradicting findings can at best be reduced to a consensus for one result over another, as the body of evidence favouring one finding over another increases.

- Semantic and functional similarity were combined in a novel way, in a grouping algorithm that operates on two levels, semantic similarity between GO terms and functional similarity between gene products. The novel approach groups gene products by distinct functional aspects rather than leaving it up to the user to manually make sense of multi-dimensional data. This facilitates analysis as the relating functional aspects between gene products are explicitly stated, rather than having to be extracted from underlying data sources.

- The algorithm can assign gene products to multiple groups, reflecting the multi-faceted and complex nature of their biological interactions with other gene products.

- The results of the proof-of-concept implementation of the FuSiGroups algorithm showed that, aside from the minor modifications detailed in Section 9.3.1, the algorithm performs well. The key functional aspects for each dataset were successfully identified, as were functionally unrelated genes, and the results were biologically relevant, as discussed in Chapter 8.

- Limitations in performance are primarily due to the nature of functional annotation, which is constantly evolving and which therefore greatly varies in detail, coverage and quality from gene product to gene product. Most notably, two groups resulted from misannotations that bore no resemblance to any of the known functions of the genes they were annotated to and which have been removed in more recent version of the GO. Therefore the analysis of the algorithm's results requires some knowledge of the gene products under investigation, or at least a general understanding of the quality of the annotations for the dataset in question, such as the extent and depth of coverage, and the sources of electronic mappings.

- The difficulty of benchmarking semantic similarity measures without using other forms of biological similarity that inherently bias the interpretation of

what semantic similarity is in turn makes it difficult to directly evaluate semantic similarity approaches and semantic similarity-related concepts such as the semantic thresholds. This has a noticeable effect on the grouping results, where the overall quality of the semantic component was lower than that of the more easily evaluated functional component.

Taking into account these limitations, the FuSiGroups algorithm shows very promising results in a range of different scenarios. Further evaluation, as detailed in Section 9.3, would be desirable but the stated aims of this work have been successfully achieved. The FuSiGroups algorithm generates biologically meaningful and accurate results that are easy to analyse without compromising on detail and information richness, making it a useful tool for any molecular biologist wishing to perform functional analysis on a list of genes.

References

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P., 2002. Molecular Biology of the Cell, 4th Edition. Garland Science.

Altenhoff, A. M., Dessimoz, C., 2009. Phylogenetic and functional assessment of orthologs inference projects and methods. PLoS Computational Biology 5 (1), e1000262.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., Sherlock, G., 2000. Gene Ontology: Tool for the unification of biology. Nature Genetics 25 (1), 25–29.

Ashburner, M., Ball, C. A., Blake, J. A., Butler, H., Cherry, J. M., Corradi, J., Dolinski, K., Eppig, J. T., Harris, M., Hill, D. P., Lewis, S., Marshall, B., Mungall, C., Reiser, L., Rhee, S., Richardson, J. E., Richter, J., Ringwald, M., Rubin, G. M., Sherlock, G., Yoon, J., 2001. Creating the gene ontology resource: Design and implementation. Genome Research 11 (8), 1425–1433.

Azuaje, F., Wang, H., Bodenreider, O., 2005. Ontology-driven similarity approaches to supporting gene functional assessment. Proceedings of the ISMB' 2005 SIG Meeting on Bio-ontologies, 9–10.

Azuaje, F., Wang, H., Zheng, H., Leonard, F., Rolland-Turner, M., Zhang, L., Devaux, Y., Wagner, D., 2011. Predictive integration of gene functional similarity and co-expression defines treatment response of endothelial progenitor cells. BMC Systems Biology 5 (1), 46.

Baeza-Yates, R., Ribeiro-Neto, B., 1999. Modern information retrieval. Addison-Wesley, New York, Harlow, England.

Barrell, D., Dimmer, E., Huntley, R. P., Binns, D., O'Donovan, C., Apweiler, R., 2009. The GOA database in 2009: an integrated Gene Ontology Annotation resource. Nucleic Acids Research 37 (Suppl 1), D396–D403.

Beinert, H., Kennedy, M., Stout, C., 1996. Aconitase as iron-sulfur protein, enzyme, and iron-regulatory protein. Chemical Reviews 96 (9), 2335–2374.

Benabderrahmane, S., Smail-Tabbone, M., Poch, O., Napoli, A., Devignes, M.-D., 2010. Intelligo: a new vector-based semantic similarity measure including annotation origin. BMC Bioinformatics 11 (1), 588.

Berg, J., Tymoczko, J., L.Stryer, 2002. Biochemistry, 5th Edition. W.H. Freeman and Company, New York.

Bezdek, J., 1981. Pattern recognition with fuzzy objective function algorithms. Plenum, NY.

Biederer, T., Volkwein, C., Sommer, T., 1997. Role of Cue1p in Ubiquitination and Degradation at the ER Surface. Science 278, 1806–1809.

Blake, J., Bult, C., Kadin, J., Richardson, J., Eppig, J., the Mouse Genome Database Group, 2011. The Mouse Genome Database (MGD): premier model organism resource for mammalian genomics and genetics. Nucleic Acids Research 39 (Suppl 1), D842–D848.

Bodenreider, O., Aubry, M., Burgun, A., 2005. Non-lexical approaches to identifying associative relations in the gene ontology. Pacific Symposium on Biocomputing, 91–102.

Bolshakova, N., Azuaje, F., Cunningham, P., 2005. A knowledge-driven approach to cluster validity assessment. Bioinformatics 21 (10), 2546–2547.

Bradford, Y., Conlin, T., Dunn, N., Fashena, D., Frazer, K., Howe, D., Knight, J., Mani, P., Martin, R., Moxon, S., Paddock, H., Pich, C., Ramachandran, S., Ruef, B., Ruzicka, L., Bauer-Schaper, H., Schaper, K., Shao, X., Singer, A., Sprague, J., Sprunger, B., Van Slyke, C., Westerfield, M., 2011. ZFIN: enhancements and updates to the zebrafish model organism database. Nucleic Acids Research 39 (Suppl 1), D822–D829.

Brameier, M., Wiuf, C., 2007. Co-clustering and visualization of gene expression data and Gene Ontology terms for *Saccharomyces cerevisiae* using self-organizing maps. Journal of Biomedical Informatics 40, 160–173.

Braun, S., Matuschewski, K., Rape, M., Thoms, S., Jentsch, S., 2002. Role of the ubiquitin-selective CDC48(UFD1/NPL4)chaperone (segregase) in ERAD of OLE1 and other substrates. EMBO Journal 21 (4), 615–621.

Büchler, M., Tisljar, U., Wolf, D., 1994. Proteinase yscD (oligopeptidase yscD). Structure, function and relationship of the yeast enzyme with mammalian thimet oligopeptidase (metalloendopeptidase, EP 24.15). European Journal of Biochemistry 219(1-2), 627–639.

Camon, E., Magrane, M., Barrell, D., Binns, D., Fleischmann, W., Kersey, P., Mulder, N., Oinn, T., Maslen, J., Cox, A., Apweiler, R., 2003. The Gene Ontology Annotation (GOA) Project: Implementation of GO in SWISS-PROT, TrEMBL, and InterPro. Genome Research 13 (4), 662–672.

Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, D., Harte, N., Lopez, R., Apweiler, R., 2004. The Gene Ontology Annotation (GOA) database: Sharing knowledge in uniprot with gene ontology. Nucleic Acids Research 32 (1), D262–D266.

Cao, S.-L., Qin, L., He, W.-Z., Zhong, Y., Zhu, Y.-Y., Li, Y.-X., 2004. Semantic search among heterogeneous biological databases based on gene ontology. Acta Biochimica et Biophysica Sinica 36 (5), 365–370.

Carman, G. M., Henry, S. A., 1989. Phospholipid biosynthesis in yeast. Annual Review of Biochemistry 58 (1), 635–667.

Chabalier, J., Mosser, J., Burgun, A., 2007. A transversal approach to predict gene product networks from ontology-based similarity. BMC Bioinformatics 8.

Chagoyen, M., Carazo, J., Pascual-Montano, A., 2008. Assessment of protein set coherence using functional annotations. BMC Bioinformatics 9 (1), 444.

Cheng, J., Cline, M., Martin, J., Finkelstein, D., Awad, T., Kulp, D., Siani-Rose, M. A., 2004. A knowledge-based clustering algorithm driven by gene ontology. Journal of Biopharmaceutical Statistics 14 (3), 687 – 700.

Cherry, J. M., Adler, C., Ball, C., Chervitz, S. A., Dwight, S. S., Hester, E. T., Jia, Y., Juvik, G., Roe, T., Schroeder, M., Weng, S., Botstein, D., 1998. SGD: Saccharomyces Genome Database. Nucleic Acids Research 26 (1), 73–79.

Chiang, J., Ho, S., Wang, W., 2008. Similar genes discovery system (SGDS): application for predicting possible pathways by using GO semantic similarity measure. Expert Systems with Applications 35, 1115–1121.

Chiang, J., Shin, J., Liu, H., Chin, C., 2006. GeneLibrarian: an effective Gene-Information summarization and visualization system. BMC Bioinformatics 7, 392–401.

Cho, Y.-R., Hwang, W., Ramanathan, M., Zhang, A., 2007. Semantic integration to identify overlapping functional modules in protein interaction networks. BMC Bioinformatics 8 (1), 265.

Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein, D., Brown, P., Herskowitz, I., 1998. The transcriptional program of sporulation in budding yeast. Science 282, 699–705.

Cobon, G. S., Crowfoot, P. D., Linnane, A. W., 1974. Biogenesis of Mitochondria. Phospholipid synthesis in vitro by yeast mitochondrial and microsomal fractions. Biochemistry Journal 144, 265–275.

Cohen, J., 1960. A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20 (1), 3746.

Couto, F., Silva, M., Coutinho, P., 2003. Implementation of a functional semantic similarity measure between gene products. Technical report, Dept. of Informatics, Faculty of Sciences, Univ. of Lisbon.

Couto, F., Silva, M., Coutinho, P., 2005. Semantic similarity over the gene ontology: Family correlation and selecting disjunctive ancestors. CIKM '05 Proceedings.

Couto, F. M., Silva, M. J., Coutinho, P. M., 2007. Measuring semantic similarity between gene ontology terms. Data & Knowledge Engineering 61 (1), 137–152.

Coux, O., Tanaka, K., Goldberg, A., 1996. Structure and functions of the 20S and 26S proteasomes. Annual Review of Biochemistry 65, 801–847.

Croft, D., O'Kelly, G., Wu, G., Haw, R., Gillespie, M., Matthews, L., Caudy, M., Garapati, P., Gopinath, G., Jassal, B., Jupe, S., Kalatskaya, I., Mahajan, S., May, B., Ndegwa, N., Schmidt, E., Shamovsky, V., Yung, C., Birney, E., Hermjakob, H., D'Eustachio, P., Stein, L., 2011. Reactome: a database of reactions, pathways and biological processes. Nucleic Acids Research 39 (Suppl 1), D691–D697.

Daum, G., Lees, N. D., Bard, M., Dickson, R., 1998. Biochemistry, cell biology and molecular biology of lipids of *Saccharomyces cerevisiae*. Yeast 14 (16), 1471–1510.

del Pozo, A., Pazos, F., Valencia, A., 2008. Defining functional distances over gene ontology. BMC Bioinformatics 9 (1), 50.

Dennis, G., Sherman, B., Hosack, D., Yang, J., Gao, W., Lane, H., Lempicki, R., 2003. DAVID: Database for Annotation, Visualization, and Integrated Discovery. Genome Biology 4 (9), R60.

DeRisi, J., Iyer, V., Brown, P., 1997. Exploring the metabolic and genetic control of gene expression on a genomic scale. Science 278, 680–686.

D'haeseleer, P., 2005. How does gene expression clustering work? Nature Biotechnology 23 (12), 1499–1501.

Dice, L. R., 1945. Measures of the amount of ecologic association between species. Ecology 26 (3), 297–302.

Do, J. H., Choi, D., 2007. Clustering approaches to identifying gene expression patterns from DNA microarray data. Molecules and Cells 25 (2), 279–288.

Dotan-Cohen, D., Kasif, S., Melkman, A. A., 2009. Seeing the forest for the trees: using the Gene Ontology to restructure hierarchical clustering. Bioinformatics 25 (14), 1789–1795.

Dotan-Cohen, D., Letovsk, S., Melkman, A. A., Kasif, S., 2009. Biological process linkage networks. PLoS ONE 4 (4), e5313.

Du, Z., Li, L., Chen, C.-F., Yu, P. S., Wang, J. Z., 2009. G-SESAME: web tools for GO-term-based gene similarity analysis and knowledge discovery. Nucleic Acids Research 37 (Suppl 2), W345–W349.

Dujon, B., Jul. 1996. The yeast genome project: what did we learn? Trends in Genetics 12 (7), 263–270.

Eisen, M. B., Spellman, P. T., Brown, P. O., Botstein, D., 1998. Cluster analysis and display of genome-wide expression patterns. Proceedings of the National Academy of Sciences of the United States of America 95 (25), 14863–14868.

Engel, S. R., Balakrishnan, R., Binkley, G., Christie, K. R., Costanzo, M. C., Dwight, S. S., Fisk, D. G., Hirschman, J. E., Hitz, B. C., Hong, E. L., Krieger,

C. J., Livstone, M. S., Miyasato, S. R., Nash, R., Oughtred, R., Park, J., Skrzypek, M. S., Weng, S., Wong, E. D., Dolinski, K., Botstein, D., Cherry, J. M., 2010. Saccharomyces Genome Database provides mutant phenotype data. Nucleic Acids Research 38 (Suppl 1), D433–D436.

Faria, D., Pesquita, C., Couto, F., Falcão, A., 2007. ProteInOn: a web tool for protein semantic similarity. Technical report, Dept. of Informatics, Faculty of Sciences, Univ. of Lisbon.

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recognition Letters 27, 861–874.

Feldmann, H., 2005. Yeast Molecular Biology: A short compendium on basic features and novel aspects. Online publication.
URL `http://biochemie.web.med.uni-muenchen.de/Yeast_Biol/`

Fellbaum, C., 1998. WordNet: An Electronic Lexical Database. The MIT Press.

Ferreira-Cerca, S., Pll, G., Gleizes, P.-E., Tschochner, H., Milkereit, P., 2005. Roles of eukaryotic ribosomal proteins in maturation and transport of pre-18S rRNA and ribosome function. Molecular Cell 20 (2), 263 – 275.

Fey, P., Gaudet, P., Curk, T., Zupan, B., Just, E. M., Basu, S., Merchant, S. N., Bushmanova, Y. A., Shaulsky, G., Kibbe, W. A., Chisholm, R. L., 2009. dictyBase - Dictyostelium bioinformatics resource update. Nucleic Acids Research 37 (Suppl 1), D515–D519.

Finn, R., Mistry, J., Tate, J., Coggill, P., Heger, A., Pollington, J., Gavin, O., Gunasekaran, P., Ceric, G., Forslund, K., Holm, L., Sonnhammer, E., Eddy, S., Bateman, A., 2010. The Pfam protein families database. Nucleic Acids Research 38, D211–222.

Fleischer, T. C., Weaver, C. M., McAfee, K. J., Jennings, J. L., Link, A. J., 2006. Systematic identification and functional screens of uncharacterized proteins associated with eukaryotic ribosomal complexes. Genes & Development 20 (10), 1294–1307.

Flicek, P., Amode, M. R., Barrell, D., Beal, K., Brent, S., Chen, Y., Clapham, P., Coates, G., Fairley, S., Fitzgerald, S., Gordon, L., Hendrix, M., Hourlier, T., Johnson, N., Khri, A., Keefe, D., Keenan, S., Kinsella, R., Kokocinski, F., Kulesha, E., Larsson, P., Longden, I., McLaren, W., Overduin, B., Pritchard, B.,

Riat, H. S., Rios, D., Ritchie, G. R. S., Ruffier, M., Schuster, M., Sobral, D., Spudich, G., Tang, Y. A., Trevanion, S., Vandrovcova, J., Vilella, A. J., White, S., Wilder, S. P., Zadissa, A., Zamora, J., Aken, B. L., Birney, E., Cunningham, F., Dunham, I., Durbin, R., Fernndez-Suarez, X. M., Herrero, J., Hubbard, T. J. P., Parker, A., Proctor, G., Vogel, J., Searle, S. M. J., 2011. Ensembl 2011. Nucleic Acids Research 39 (Suppl 1), D800–D806.

Fontana, P., Cestaro, A., Velasco, R., Formentin, E., Toppo, S., 2009. Rapid annotation of anonymous sequences from genome projects using semantic similarities and a weighting scheme in Gene Ontology. PLoS ONE 4 (2), e4619.

Francis, W., Kucera, H., 1982. Frequency analysis of English usage: Lexicon and Grammar. Houghton Mifflin, Boston.

Gasch, A., Eisen, M., 2002. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. Genome Biology 3 (11), research0059.

Gentleman, R., 2005. Visualizing and distances using GO.
URL `http://www.bioconductor.org/packages/release/bioc/vignettes/GOstats/inst/doc/GOvis.pdf`

Girke, T., 2010. R & Bioconductor Manual. UC Riverside.

Graybill, E. R., Rouhier, M. F., Kirby, C. E., Hawes, J. W., 2007. Functional comparison of citrate synthase isoforms from *S. cerevisiae*. Archives of Biochemistry and Biophysics 465 (1), 26 – 37.

Green, D., Swets, J., 1966. Signal detection theory and psychophysics. John Wiley and Sons Inc, New York.

Grossmann, S., Bauer, S., Robinson, P. N., Vingron, M., 2007. Improved detection of overrepresentation of Gene Ontology annotations with parent-child analysis. Bioinformatics 23 (22), 3024–3031.

Guo, X., 2007. Gene Ontology-based semantic similarity measures.
URL `http://bioconductor.org/packages/2.1/bioc/vignettes/SemSim/inst/doc/SemSim.pdf`

Guo, X., Liu, R., Hu, H., Liebman, M., 2006. Assessing semantic similarity measures for the characterization of human regulartory pathways. Bioinformatics 22 (8), 967–973.

Hakes, L., Pinney, J., Lovell, S., Oliver, S., Robertson, D., 2007. All duplicates are not equal: the difference between small-scale and genome duplication. Genome Biology 8 (10), R209.

Handl, J., Knowles, J., Kell, D. B., 2005. Computational cluster validation in post-genomic data analysis. Bioinformatics 21 (15), 3201–3212.

Hanley, J., McNeil, B., 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143 (1), 29–36.

Harris, T. W., Antoshechkin, I., Bieri, T., Blasiar, D., Chan, J., Chen, W. J., De La Cruz, N., Davis, P., Duesbury, M., Fang, R., Fernandes, J., Han, M., Kishore, R., Lee, R., Müller, H.-M., Nakamura, C., Ozersky, P., Petcherski, A., Rangarajan, A., Rogers, A., Schindelman, G., Schwarz, E. M., Tuli, M. A., Van Auken, K., Wang, D., Wang, X., Williams, G., Yook, K., Durbin, R., Stein, L. D., Spieth, J., Sternberg, P. W., 2010. Wormbase: a comprehensive resource for nematode research. Nucleic Acids Research 38 (Suppl 1), D463–D467.

Hawkins, T., Chitale, M., Kihara, D., 2010. Functional enrichment analyses and construction of functional similarity networks with high confidence function prediction by PFP. BMC Bioinformatics 11 (1), 265.

Heinemeyer, W., A.Kleinschmidt, J., Saidowsky, J., Escher, C., Wolf, D. H., 1991. Proteinase yscE, the yeast proteasome/multicatalytic-multifunctional proteinase: mutants unravel its function in stress induced proteolysis and uncover its necessity for cell survival. EMBO Journal 10 (3), 555–562.

Herrmann, C., Bérard, S., Tichit, L., 2009. SimCT: a generic tool to visualize ontology based relationships for biological objects. Bioinformatics.

Hill, D. P., Davis, A. P., Richardson, J. E., Corradi, J. P., Ringwald, M., Eppig, J. T., Blake, J. A., May 2001. Program description: Strategies for biological annotation of mammalian systems: Implementing Gene Ontologies in Mouse Genome Informatics. Genomics 74 (1), 121–128.

Hochstrasser, M., 1996. Ubiquitin-dependent protein degradation. Annual Review of Genetics 30, 405–439.

Hrycyna, C., Clarke, S., 1993. Purification and characterization of a novel metalloendopeptidase from *saccharomyces cerevisiae*. Biochemistry 32, 11293–11301.

Huang, D., Sherman, B., Tan, Q., Collins, J., Alvord, W. G., Roayaei, J., Stephens, R., Baseler, M., Lane, H. C., Lempicki, R., 2007. The DAVID Gene Functional Classification Tool: a novel biological module-centric algorithm to functionally analyze large gene lists. Genome Biology 8 (9), R183.

Huttonhower, C., Troyanskaya, O., 2008. Assessing the functional structure of genomic data. Bioinformatics 24 (13), i330–i338.

Huttonhower, C., Troyanskaya, O., 2009. Prediction of Gene Ontology annotations by integrating high-throughput datasets, unpublished SGD paper.

Jaccard, P., 1908. Nouvelles recherches sur la distribution florale. Bulletin de la Société Vaudoise des Sciences Naturelles 44, 223–270.

Jain, S., Bader, G., 2010. An improved method for scoring protein-protein interactions using semantic similarity within the gene ontology. BMC Bioinformatics 11 (1), 562.

Jaiswal, P., Ni, J., Yap, I., Ware, D., Spooner, W., Youens-Clark, K., Ren, L., Liang, C., Zhao, W., Ratnapu, K., Faga, B., Canaran, P., Fogleman, M., Hebbard, C., Avraham, S., Schmidt, S., Casstevens, T., Buckler, E., Stein, L., McCouch, S., 2006. Gramene: a bird's eye view of cereal genomes. Nucleic Acids Research 34, D717–723.

Jakonienė, V., Rundqvist, D., Lambrix, P., 2006. A method for similarity-based grouping of biological data. In: Proceedings of the 3rd International Workshop on Data Integration in the Life Sciences - DILS06. pp. 136–151.

Jiang, D., Tang, C., Zhang, A., 2004. Cluster analysis for gene expression data: a survey. IEEE Transactions on Knowledge and Data Engineering 16 (11), 1370–1386.

Jiang, J., Conrath, D., 1997. Semantic similarity based on corpus statistics and lexical taxonomy. Proceedings of the International Conference Research on Computational Linguistics.

Jing, L., Ng, M., 2010. Prior knowledge based mining functional modules from Yeast PPI networks with gene ontology. BMC Bioinformatics 11 (Suppl 11), S3.

Jing, L., Ng, M. K., Liu, Y., January 2010. Construction of gene networks with hybrid approach from expression profile and gene ontology. Transactions on Information Technology in Biomedicine 14, 107–118.

Kanehisa, M., Goto, S., 2000. KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Research 28 (1), 27–30.

Kang, B.-Y., Ko, S., Kim, D.-W., 2010. SICAGO: Semi-supervised cluster analysis using semantic distance between gene pairs in Gene Ontology. Bioinformatics 26 (10), 1384–1385.

Khatri, P., Drăghici, S., 2005. Ontological analysis of gene expression data: current tools, limitations, and open problems. Bioinformatics 21 (18), 3587–3595.
URL `http://bioinformatics.oxfordjournals.org/cgi/content/abstract/21/18/3587`

Kim, S. Y., Choi, T. M., 2005. Fuzzy types clustering for microarray data. Proceedings of World Academcy of Science, Engineering and Technology 4, 12–15.

Kuchler, K., Daum, G., Paltauf, F., 1986. Subcellular and submitochondrial localization of phospholipid-synthesizing enzymes in *Saccharomyces cerevisiae*. Journal of Bacteriology 165 (3), 901–910.

Kumar, A., Agarwal, S., Heyman, J. A., Matson, S., Heidtman, M., Piccirillo, S., Umansky, L., Drawid, A., Jansen, R., Liu, Y., Cheung, K.-H., Miller, P., Gerstein, M., Roeder, G. S., Snyder, M., 2002. Subcellular localization of the yeast proteome. Genes & Development 16, 707–719.

Kustra, R., Zagdanski, A., 2010. Data-fusion in clustering microarray data: Balancing discovery and interpretability. IEEE/ACM Transactions on Computational Biology and Bioinformatics 7, 50–63.

Lagesen, K., Ussery, D. W., Wassenaar, T. M., 2010. Genome update: The thousandth genome - a cautionary tale. Microbiology 156, 603–608.

Lance, G., Williams, W., 1967. A general theory of classificatory sorting strategies. 1. Hierarchical systems. Computer Journal 9, 373–380.

Lasko, T., Bhagwat, J., Zou, K., Ohno-Machado, L., 2005. The use of receiver operating characteristic curves in biomedical informatics. Journal of Biomedical Informatics 38, 404–415.

Lee, H. K., Hsu, A. K., Sajdak, J., Qin, J., Pavlidis, P., 2004. Coexpression analysis of human genes across many microarray data sets. Genome Research 14 (6), 1085–1094.

Lei, Z., Dai, Y., 2006. Assessing protein similarity with gene ontology and its use in subnuclear localization prediction. BMC Bioinformatics 7 (1), 491.

Li, J., Yuan, Z., Zhang, Z., 2010. The cellular robustness by genetic redundancy in budding yeast. PLoS Genetics 6 (11), e1001187.

Li, X., Kusmierczyk, A. R., Wong, P., Emili, A., Hochstrasser, M., May 2007. β-Subunit appendages promote 20S proteasome assembly by overcoming an Ump1-dependent checkpoint. EMBO Journal 26 (9), 2339–2349.

Lin, D., 1998. An information-theoretic definition of similarity. Proceeding of the 15th International Conference on Machine Learning, 296–304.

Lin, N., Wu, B., Jansen, R., Gerstein, M., Zhao, H., 2004. Information assessment on predicting protein-protein interactions. BMC Bioinformatics 5 (1), 154.

Lord, P., Stevens, R., Brass, A., Goble, C., 2003a. Investigating semantic similarity measures across the Gene Ontology: The relationship between sequence and annotation. Bioinformatics 19, 1275–1283.

Lord, P., Stevens, R., Brass, A., Goble, C., 2003b. Semantic similarity measures as tools for exploring the Gene Ontology. Proceedings of the Pacific Symposium on Biocomputing 8, 601–612.

Louie, B., Higdon, R., Kolker, E., 2009. A statistical model of protein sequence similarity and function similarity reveals overly-specific function predictions. PLoS ONE 4 (10), e7546.

Marques, A., Glanemann, C., Ramos, P. C., Dohmen, R. J., 2007. The C-terminal extension of the β7 subunit and activator complexes stabilize nascent 20 S proteasomes and promote their maturation. The Journal of Biological Chemistry 282 (48), 34869–34876.

Martin, D., Brun, C., Remy, E., Mouren, P., Thieffry, D., Jacq, B., 2004. GOToolBox: Function analysis of gene datasets based on Gene Ontology. Genome Biology 5 (12), R101.

Martinez, R., Pasquier, C., Pasquier, N., 2007. GenMiner: Mining informative association rules from genomic data. Proceedings of the IEEE BIBM International Conference on Bioinformatics and Biomedecine, 15–22.

McKusick-Nathans Institute of Genetic Medicine, Johns Hopkins University (Baltimore, MD) and National Center for Biotechnology Information, National Library of Medicine (Bethesda, MD), 2011. Online Mendelian Inheritance in Man, OMIM™.
URL http://www.ncbi.nlm.nih.gov/omim/

Mieczkowski, P., Dajewski, W., Podlaska, A., Skoneczna, A., Ciesla, Z., Sledziewska-Gójska, E., 2000. Expression of UMP1 is inducible by DNA damage and required for resistance of *S. cerevisiae* cells to UV light. Current Genetics 38 (2), 53–59.

Miller, G., Leacock, C., Tengi, R., Bunker, R., 1993. A semantic concordance. Proceedings of ARPA Workshop on Human Language Technology, 303–308.

Mistry, M., Pavlidis, P., 2008. Gene Ontology term overlap as a measure of gene functional similarity. BMC Bioinformatics 9.

Mulder, N. J., Apweiler, R., Attwood, T. K., Bairoch, A., Barrell, D., Bateman, A., Binns, D., Biswas, M., Bradley, P., Bork, P., Bucher, P., Copley, R. R., Courcelle, E., Das, U., Durbin, R., Falquet, L., Fleischmann, W., Griffiths-Jones, S., Haft, D., Harte, N., Hulo, N., Kahn, D., Kanapin, A., Krestyaninova, M., Lopez, R., Letunic, I., Lonsdale, D., Silventoinen, V., Orchard, S. E., Pagni, M., Peyruc, D., Ponting, C. P., Selengut, J. D., Servant, F., Sigrist, C. J. A., Vaughan, R., Zdobnov, E. M., 2003. The InterPro Database, 2003 brings increased coverage and new features. Nucleic Acids Research 31 (1), 315–318.

National Human Genome Research Institute, April 1996. International team completes dna sequence of yeast. Press release.
URL http://www.genome.gov/10000510

Ng, A. Y., Jordan, M. I., Weiss, Y., 2001. On spectral clustering: Analysis and an algorithm. In: Advances in Neural Information Processing Systems. MIT Press, pp. 849–856.

Othman, R. M., Deris, S., Illias, R. M., 2008. A genetic similarity algorithm for searching the gene ontology terms and annotating anonymous protein sequences. Journal of Biomedical Informatics 41, 65–81.

Ovaska, K., Laakso, M., Hautaniemi, S., 2008. Fast Gene Ontology based clustering for microarray experiments. BioData Mining 1 (1), 11.

Oxford English Dictionary, 1989. 2nd edition. Oxford: Clarendon Press.

Oyedotun, K. S., Lemire, B. D., 2004. The quaternary structure of the *Saccharomyces cerevisiae* succinate dehydrogenase. Journal of Biological Chemistry 279 (10), 9424–9431.

Pandey, J., Koyuturk, M., Grama, A., 2010. Functional characterization and topological modularity of molecular interaction networks. BMC Bioinformatics 11 (Suppl 1), S35.

Papachristoudis, G., Diplaris, S., Mitkas, P. A., 2010. SoFoCles: Feature filtering for microarray classification based on Gene Ontology. Journal of Biomedical Informatics 43 (1), 1 – 14.

Pekar, V., Staab, S., 2002. Taxonomy learning: factoring the structure of a taxonomy into a semantic classification decision. Proceedings of the Nineteenth Conference on Computational Linguistics 2, 786–792.

Pesquita, C., Faria, D., Bastos, H., Falcão, A., Couto, F., 2007. Evaluating GO-based semantic similarity measures. BioOntologies SIG at ISMB/ECCB - 15th Annual International Conference on Intelligent Systems for Molecular Biology (ISMB) & 6th European Conference on Computational Biology (ECCB).

Pesquita, C., Faria, D., Bastos, H., Ferreira, A., Falcão, A., Couto, F., 2008. Metrics for GO based protein semantic similarity: a systematic evaluation. BMC Bioinformatics 9 (Suppl 5), S4.

Pesquita, C., Faria, D., Falcão, A., Lord, P., Couto, F., Jul. 2009. Semantic similarity in biomedical ontologies. PLoS Comput Biol 5 (7), e1000443.

Pourhashem, M. M., Kelarestaghi, M., Pedram, M. M., 2010. Missing value estimation in microarray data using fuzzy clustering and semantic similarity. Global Journal of Computer Science and Technology 10 (12), 18–22.

Pronk, J. T., Yde Steensma, H., Van Dijken, J. P., 1996. Pyruvate metabolism in *Saccharomyces cerevisiae*. Yeast 12 (16), 1607–1633.

Pruitt, K. D., Tatusova, T., Maglott, D. R., 2006. NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. Nucleic Acids Research 35 (Suppl 1), D61–D65.

R Development Core Team, 2010. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

Rada, R., Mili, H., Bicknell, E., Blettner, M., 1989. Development and application of a metric on semantic nets. In: IEEE Transaction on Systems, Man and Cybernetics. Vol. 19. pp. 17–30.

Ramírez, F., Schlicker, A., Assenov, Y., Lengauer, T., Albrecht, M., 2007. Computational analysis of human protein interaction networks. Proteomics 7 (15), 2541–2552.

Ramos, P. C., Höckendorff, J., Johnson, E. S., Varshavsky, A., Dohmen, R. J., 1998. Ump1p is required for proper maturation of the 20S proteasome and becomes its substrate upon completion of the assembly. Cell 92 (4), 489 – 499.

Regev-Rudzki, N., Battat, E., Goldberg, I., Pines, O., 2009. Dual localization of fumarase is dependent on the integrity of the glyoxylate shunt. Molecular Microbiology 72 (2), 297–306.

Regev-Rudzki, N., Karniely, S., Ben-Haim, N. N., Pines, O., 2005. Yeast aconitase in two locations and two metabolic pathways: Seeing small amounts is believing. Molecular Biology of the Cell 16, 4163–4171.

Resnik, P., 1995. Using information content to evaluate semantic similarity in a taxonomy. Proceedings of the 14th International Joint Conference on Artificial Intelligence, 448–453.

Richards, A. J., Muller, B., Shotwell, M., Cowart, L. A., Rohrer, B., Lu, X., 2010. Assessing the functional coherence of gene sets with metrics based on the Gene Ontology graph. Bioinformatics 26 (12), i79–i87.

Romero-Zaliz, R., del Val, C., Cobb, J. P., Zwir, I., 2008. Onto-CC: a web server for identifying Gene Ontology conceptual clusters. Nucleic Acids Research 36 (Suppl 2), W352–W357.

Roubelakis, M., Zotos, P., Papachristoudis, G., Michalopoulos, I., Pappa, K., Anagnou, N., Kossida, S., 2009. Human microRNA target analysis and gene ontology clustering by GOmir, a novel stand-alone application. BMC Bioinformatics 10 (Suppl 6), S20.

Sanger, F., Air, G. M., Barrell, B. G., Brown, N. L., Coulson, A. R., Fiddes, J. C., Hutchison, C. A., Slocombe, P. M., Smith, M., Feb. 1977a. Nucleotide sequence of bacteriophage ϕX174 DNA. Nature 265 (5596), 687–695.

Sanger, F., Nicklen, S., Coulson, A. R., 1977b. DNA sequencing with chain-terminating inhibitors. Proceedings of the National Academy of Sciences 74 (12), 5463–5467.

Santamaria, P. G., Finley, D., Ballesta, J., Remacha, M., 2003. Rpn6p, a proteasome subunit from saccharomyces cerevisiae, is essential for the assembly and activity of the 26 S proteasome. The Journal of Biological Chemistry 278 (9), 6687–6695.

Schlicker, A., Albrecht, M., 2007. FunSimMat: a comprehensive functional similarity database. Nucleic Acids Research 36 (Database issue), D434–D439.

Schlicker, A., Albrecht, M., 2010. FunSimMat update: new features for exploring functional similarity. Nucleic Acids Research 38 (Suppl 1), D244–D248.

Schlicker, A., Domingues, F. S., Rahnenführer, J., Lengauer, T., 2006. A new measure for functional similarity of gene products based on Gene Ontology. BMC Bioinformatics 7, 302.

Schlicker, A., Huthmacher, C., Ramírez, F., Lengauer, T., Albrecht, M., 2007a. Functional evaluation of domain domain interactions and human protein interaction networks. Bioinformatics 23 (7), 859–865.

Schlicker, A., Lengauer, T., Albrecht, M., 2010. Improving disease gene prioritization using the semantic similarity of gene ontology terms. Bioinformatics 26 (18), i561–i567.

Schlicker, A., Rahnenführer, J., Albrecht, M., Lengauer, T., Domingues, F. S., 2007b. GOTax: investigating biological processes and biochemical activities along the taxonomic tree. Genome Biology 8, R33.

Schön, T., Tsymbal, A., Huber, M., 2010. Gene-pair representation and incorporation of GO-based semantic similarity into classification of gene expression data. Rough Sets and Current Trends in Computing 6086, 217–226.

Schuberth, C., Buchberger, A., 2005. Membrane-bound Ubx2 recruits Cdc48 to ubiquitin ligases and their substrates to ensure efficient ER-associated protein degradation. Nature Cell Biology 7, 999–1006.

Sevilla, J. L., Segura, V., Podhorski, A., Guruceaga, E., Mato, J. M., Martínez-Cruz, L. A., Corrales, F. J., Rubio, A., 2005. Correlation between gene expression and GO semantic similarity. IEEE-ACM Transactions on Computational Biology and Bioinformatics 2 (4), 330–338.

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B., Ideker, T., 2003. Cytoscape: A software environment for integrated models of biomolecular interaction networks. Genome Research 13 (11), 2498–2504.

Sheehan, B., Quigley, A., Gaudin, B., Dobson, S., 2008. A relation based measure of semantic similarity for Gene Ontology annotations. BMC Bioinformatics 9, 468.

Shendure, J., Ji, H., Oct. 2008. Next-generation DNA sequencing. Nature Biotechnology 26 (10), 1135–1145.

Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T., 2005. ROCR: visualizing classifier performance in R. Bioinformatics 21 (20), 3940–3941.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S., Scheuermann, R. H., Shah, N., Whetzel, P. L., Lewis, S., 2007. The OBO foundry: coordinated evolution of ontologies to support biomedical data integration. Nature Biotechnology 25 (11), 1251–1255.

Sneath, P., Sokal, R., 1973. Numerical taxonomy. W.H. Freeman and Company, San Francisco.

Sokal, R., Michener, C., 1958. A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin 38, 1409–1438.

Speer, N., Spieth, C., Zell, A., 2004. A memetic co-clustering algorithm for gene expression profiles and biological annotation. In: Congress on Evolutionary Computation, 2004. CEC2004. Vol. 2. pp. 1631–1638.

Spellman, P., Sherlock, G., Iyer, V., Zhang, M., Anders, K., Eisen, M., Brown, P., Botstein, D., Futcher, B., 1998. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. Molecular Biology of the Cell 9 (12), 3273–3297.

Stevens, R., Goble, C. A., Bechhofer, S., 2000. Ontology-based knowledge representation for bioinformatics. Briefings in Bioinformatics 1 (4), 398–414.

Suzuki, R., Shimodaira, H., 2009. pvclust: Hierarchical Clustering with P-Values via Multiscale Bootstrap Resampling. R package version 1.2-1.
URL `http://www.is.titech.ac.jp/~shimo/prog/pvclust/`

Swarbreck, D., Wilks, C., Lamesch, P., Berardini, T. Z., Garcia-Hernandez, M., Foerster, H., Li, D., Meyer, T., Muller, R., Ploetz, L., Radenbaugh, A., Singh, S., Swing, V., Tissier, C., Zhang, P., Huala, E., 2008. The Arabidopsis Information Resource (TAIR): gene structure and function annotation. Nucleic Acids Research 36 (Database issue), D1009–D1014.

Swets, J., 1988. Measurig the accuracy of diagnostics systems. Science 240, 1285–1293.

Takeuchi, J., Fujimuro, M., Yokosawa, H., Tanaka, K., Toh-e, A., 1999. Rpn9 is required for efficient assembly of the yeast 26S proteasome. Molecular and Cellular Biology 19 (10), 6575–6584.

Tam, A., Schmidt, W., Michaelis, S., 2001. The multispanning membrane protein Ste24p catalyzes CAAX proteolysis and NH2-terminal processing of the yeast a-Factor precursor. Journal of Biological Chemistry 276 (50), 46798–46806.

Tao, Y., Sam, L., Li, J., Friedman, C., Lussier, Y., 2007. Information theory applied to the sparse Gene Ontology annotation network to predict novel gene function. Bioinformatics 23 (ISMB/ECCB 2007), i529–i538.

Tari, L., Baral, C., Kim, S., 2009. Fuzzy c-means clustering with prior biological knowledge. Journal of Biomedical Informatics 42, 74–81.

Tedder, P. M. R., Bradford, J. R., Needham, C. J., McConkey, G. A., Bulpitt, A. J., Westhead, D. R., 2010. Gene function prediction using semantic similarity clustering and enrichment analysis in the malaria parasite *Plasmodium falciparum*. Bioinformatics 26 (19), 2431–2437.

The UniProt Consortium, 2008. The universal protein resource (UniProt). Nucleic Acids Research 36, D190–D195.

Tuikkala, J., Elo, L., Nevalainen, O. S., Aittokallio, T., 2006. Improving missing value estimation in microarray data with gene ontology. Bioinformatics 22 (5), 566–572.

Tweedie, S., Ashburner, M., Falls, K., Leyland, P., McQuilton, P., Marygold, S., Millburn, G., Osumi-Sutherland, D., Schroeder, A., Seal, R., Zhang, H., The FlyBase Consortium, 2009. FlyBase: enhancing Drosophila Gene Ontology annotations. Nucleic Acids Research 37, D555–D559.

Twigger, S. N., Shimoyama, M., Bromberg, S., Kwitek, A. E., Jacob, H. J., the RGD Team, 2006. The Rat Genome Database, update 2007: Easing the path from disease to data and back again. Nucleic Acids Research 35 (Suppl 1), D658–D662.

Valiente, G., 2002. Algorithms on trees and graphs. Springer.

Wang, H., Azuaje, F., Bodenreider, O., Dopazo, J., 2004. Gene expression correlation and Gene Ontology-based similarity: An assessment of quantitative relationships. Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB'2004), 23–31.

Wang, H., Zheng, H., Azuaje, F., 2010a. Ontology- and graph-based similarity assessment in biological networks. Bioinformatics 26 (20), 2643–2644.

Wang, H., Zheng, H., Browne, F., Glass, D. H., Azuaje, F., 2010b. Integration of Gene Ontology-based similarities for supporting analysis of protein-protein interaction networks. Pattern Recognition Letters 31 (14), 2073 – 2082.

Wang, J. Z., Du, Z., Payattakool, R., Yu, P. S., Chen, C., 2007. A new method to measure the semantic similarity of GO terms. Bioinformatics 23 (10), 1274–1281.

Wolf, D. H., Hilt, W., 2004. The proteasome: a proteolytic nanomachine of cell regulation and waste disposal. Biochimica et Biophysica Acta 1695, 19–31.

Wolting, C., McGlade, C., Tritchler, D., 2006. Cluster analysis of protein array results via similarity of Gene Ontology annotation. BMC Bioinformatics 7, 338–340.

Wu, H., Su, Z., Mao, F., Olman, V., Xu, Y., 2005. Prediction of functional modules based on comparative genome analysis and Gene Ontology application. Nucleic Acids Research 33 (9), 2822–2837.

Wu, X., Zhu, L., Guo, J., Zhang, D.-Y., Lin, K., 2006. Prediction of yeast protein-protein interaction network: insights from the Gene Ontology and annotations. Nucleic Acids Research 34 (7), 2137–2150.

Xu, T., Du, L., Zhou, Y., 2008. Evaluation of GO-based functional similarity measures using *S. cerevisiae* protein interaction and expression profile data. BMC Bioinformatics 9, 472.

Ye, P., Peyser, B. D., Pan, X., Boeke, J. D., Spencer, F. A., Bader, J. S., 2005. Gene function prediction from congruent synthetic lethal interactions in yeast. Molecular Systems Biology 1, 2005.0026.

Yu, G., Li, F., Qin, Y., Bo, X., Wu, Y., Wang, S., 2010. GOSemSim: an R package for measuring semantic similarity among GO terms and gene products. Bioinformatics 26 (7), 976–978.

Yu, H., Gao, L., Tu, K., Guo, Z., 2005. Broadly predicting specific gene functions with expression similarity and taxonomy similarity. Gene 352, 75 – 81.

Yu, H., Jansen, R., Stolovitzky, G., Gerstein, M., 2007. Total ancestry measure: quantifying the similarity in tree-like classification, with genomic applications. Bioinformatics 23 (16), 2163–2173.

Yuan, F., Wang, R., Guan, M., He, G., 2010. A novel computational method for predicting disease genes based on functional similarity. In: Proceedings of the Advanced intelligent computing theories and applications, and 6th international conference on Intelligent computing. ICIC'10. pp. 42–51.

Yuan, F., Zhou, Y., 2008. CDGMiner: A new tool for the identification of disease genes by text mining and functional similarity analysis. In: Proceedings of the 4th international conference on Intelligent Computing: Advanced Intelligent Computing Theories and Applications - with Aspects of Artificial Intelligence. ICIC '08. Springer-Verlag, Berlin, Heidelberg, pp. 982–989.

Appendix A

This appendix contains the summaries of the analyses of the performances of the different semantic and functional similarity approaches for the three individual datasets. Each section follows the same approach as that taken in Chapter 4 for the aggregate dataset, although the information in the tables and figures is in a more summarised form.

A.1 Gene expression dataset

As a reminder, the gene expression dataset was generated from the Eisen gene expression data and consists of a positive dataset of 1260 gene products and a negative dataset of equal size. The positive dataset consists of pairs of gene products that were clustered together using hierarchical clustering as described in Section 7.3, at a cut-off of 0.1. The negative dataset consists of pairs where one gene product is among the 37 right-most gene products in the tree and the other gene product is among the 37 left-most gene products. The resampled dataset of 10 times 500 true positives and 500 true negatives was processed using the ROCR package in R, in the same manner as described in Chapter 4 for the aggregate dataset.

A.1.1 ROC curves

Figures A.1 to A.4 show a selection of ROC curves for the gene expression dataset. It is immediately clear that all the gene expression ROC curves are closer to the perfect curve than the aggregate dataset. This suggests that gene expression similarity is either a very close match to functional similarity or that the true positive part of the gene expression dataset consists of very closely related gene products.

From Figure 4.2 (Gene expression dataset, top right), it was derived earlier that all the approaches are very close in their performance, with the exception of Resnik's approach, where the curve differs the most from the others.

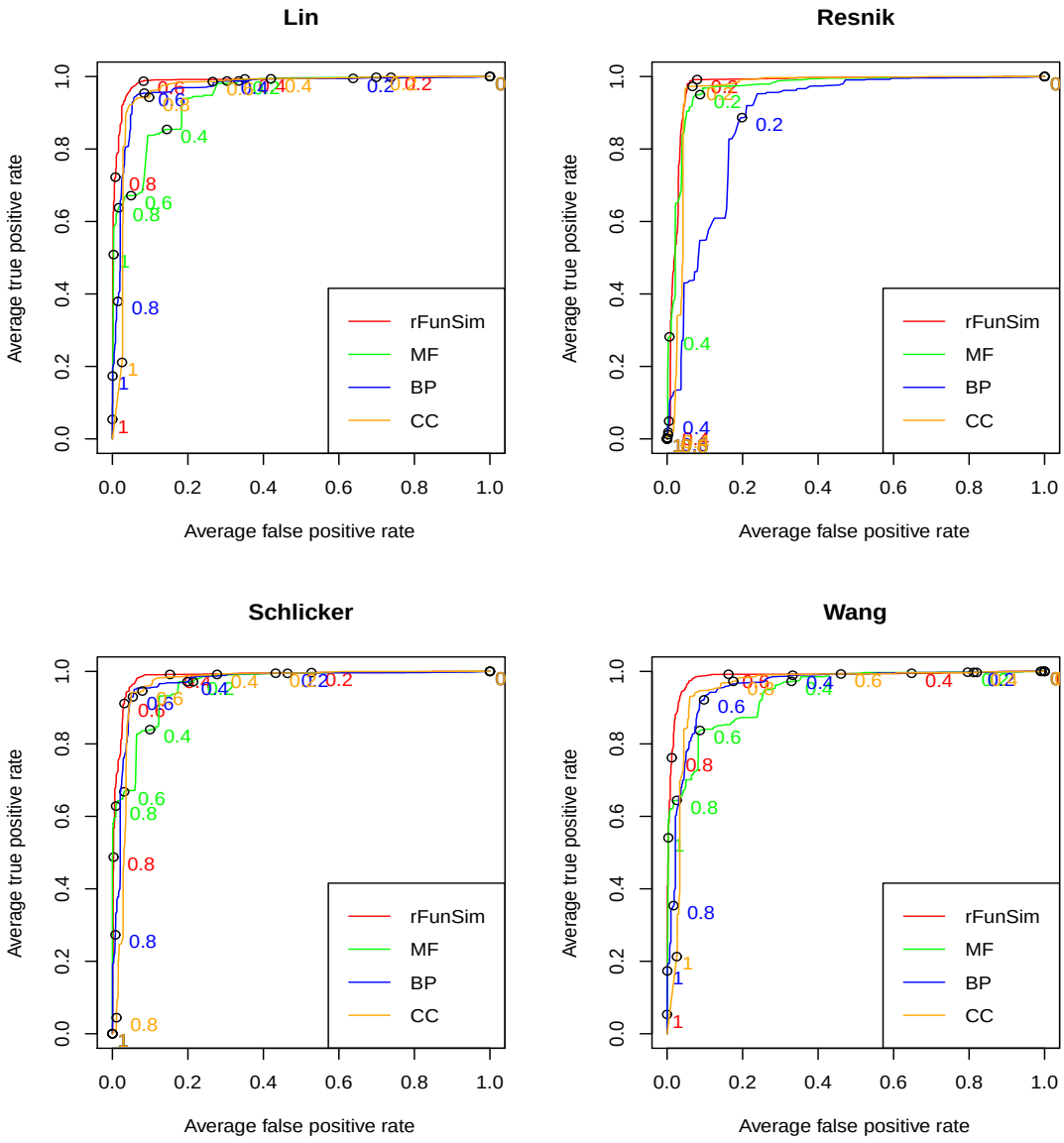

Figure A.1: ROC curves for individual ontology and rFunSim scores for all approaches with "BMA-MICA'

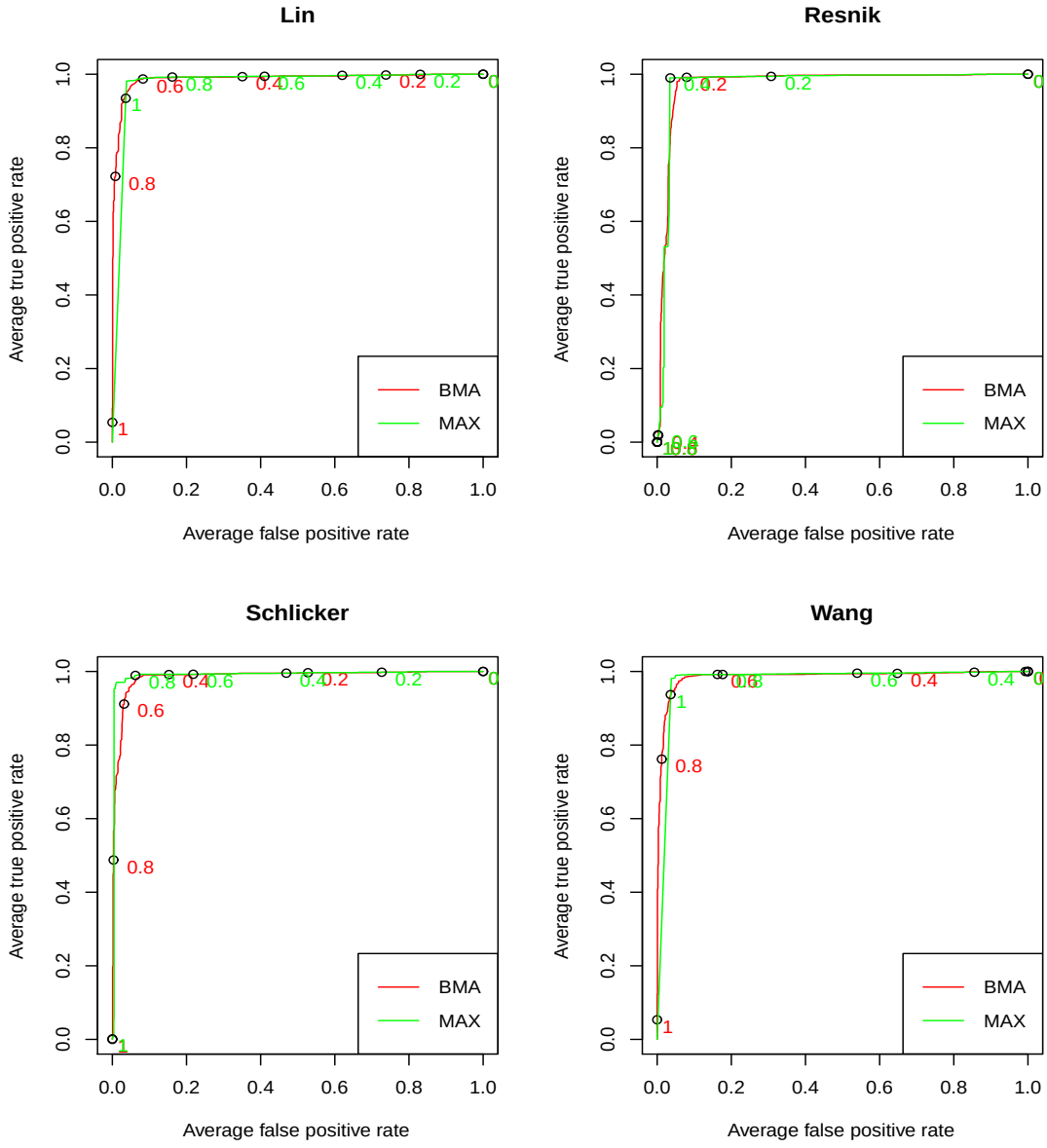

Figure A.2: ROC curves for all approaches for BMA and MAX with "all-MICA-rFunSim"

For the individual ontology scores compared to rFunSim (Figure A.1), there is always one score that performs significantly worse than the others. In most cases, this score corresponds to MF, except for Resnik, when it is BP. rFunSim generally performs very strongly although in some cases this is clearer than in others.

Figure A.2 shows the trends for BMA and MAX on "all-MICA-rFunSim". Each pair of curves are very close and the AUC values will be required to make any judgement about the overall performance of the functional similarity approaches. It is however notable that once again, the ROC curves for MAX for both Lin and Wang have the first threshold not on or near the origin of the curve, but in fact between 0.8 and 1.0 on the Y-axis.

Figure A.3 compares the ROC curves for all the IC-based measures for MICA and GraSM. Wang's approach is shown in a different colour and for reference only as this approach is not affected by ancestor selection. Although all the curve pairs are very close, it appears that the red line, representing MICA, generally shows a slightly better performance than GraSM.

Finally, Figure A.4 shows the comparison between full annotation and non-electronic annotation for all approaches and for "BMA-MICA-rFunSim". As with most ROC curves for the gene expression dataset, the curves are generally very close. In most cases however, the green curve, representing the non-electronic dataset, appears to be slightly higher than the red curve of the full annotation data. Nonetheless, comparison of the AUC indexes is necessary for a conclusive answer.

A.1.2 AUC results

AUCs were computed as previously described. Overall AUCs are shown in Table A.1.

A quick overview of the AUCs makes it obvious that in most cases, rFunSim again has higher AUCs than any of the other scores, suggesting that the aggregate score performs better than the individual ontologies. Exceptions to this are listed in Table A.2.

The CC ontology score outperforms the aggregate score most often, in 7 out of 28 cases (25% of cases). There is no discernible pattern in the exceptions and no single combination of variables where rFunSim is outperformed in all three cases although for "Schlicker-MAX-all-GraSM", both MF and CC outperform rFunSim.

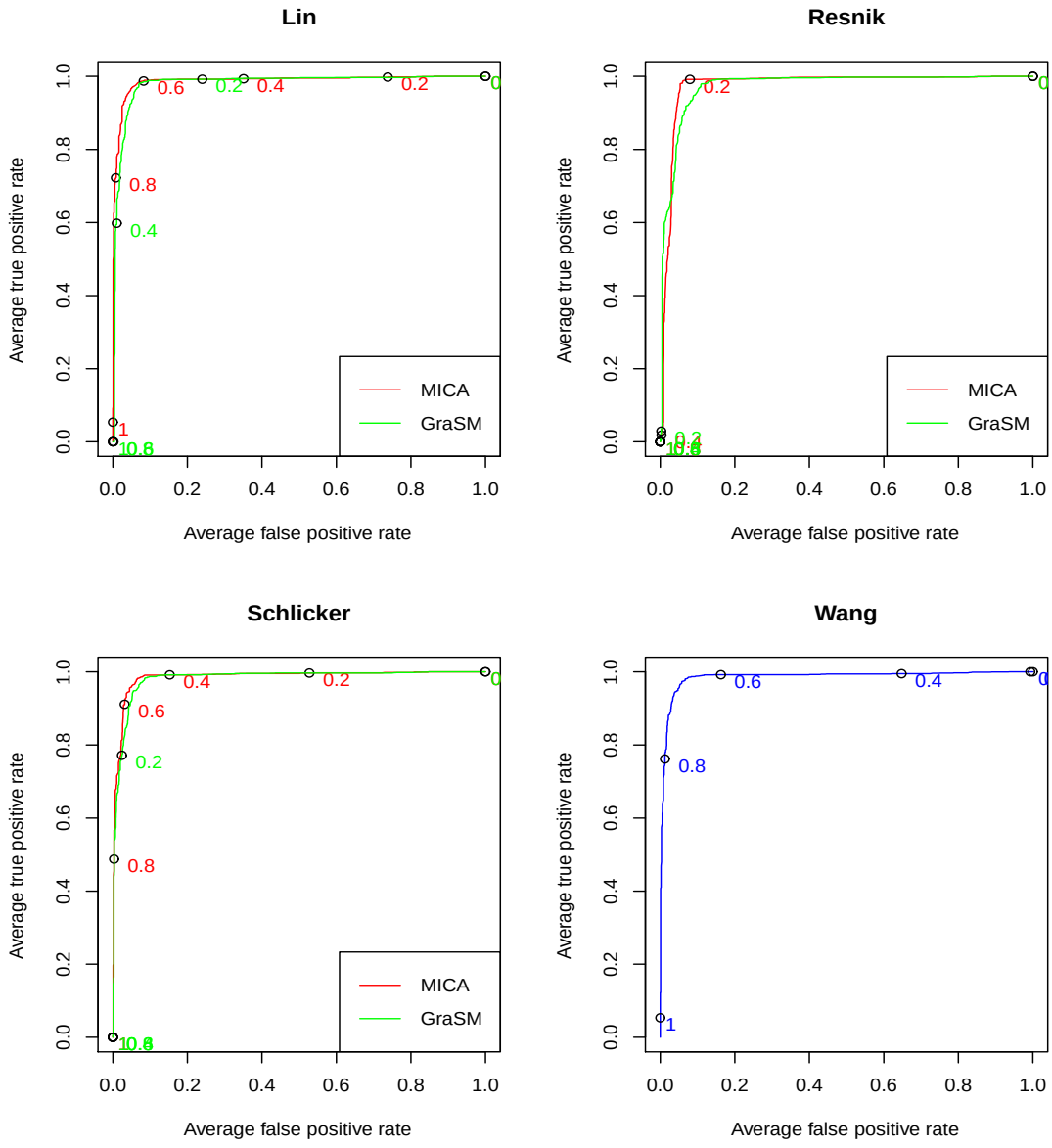

Figure A.3: ROC curves for all approaches for MICA and GraSM with "all-BMA-rFunSim"

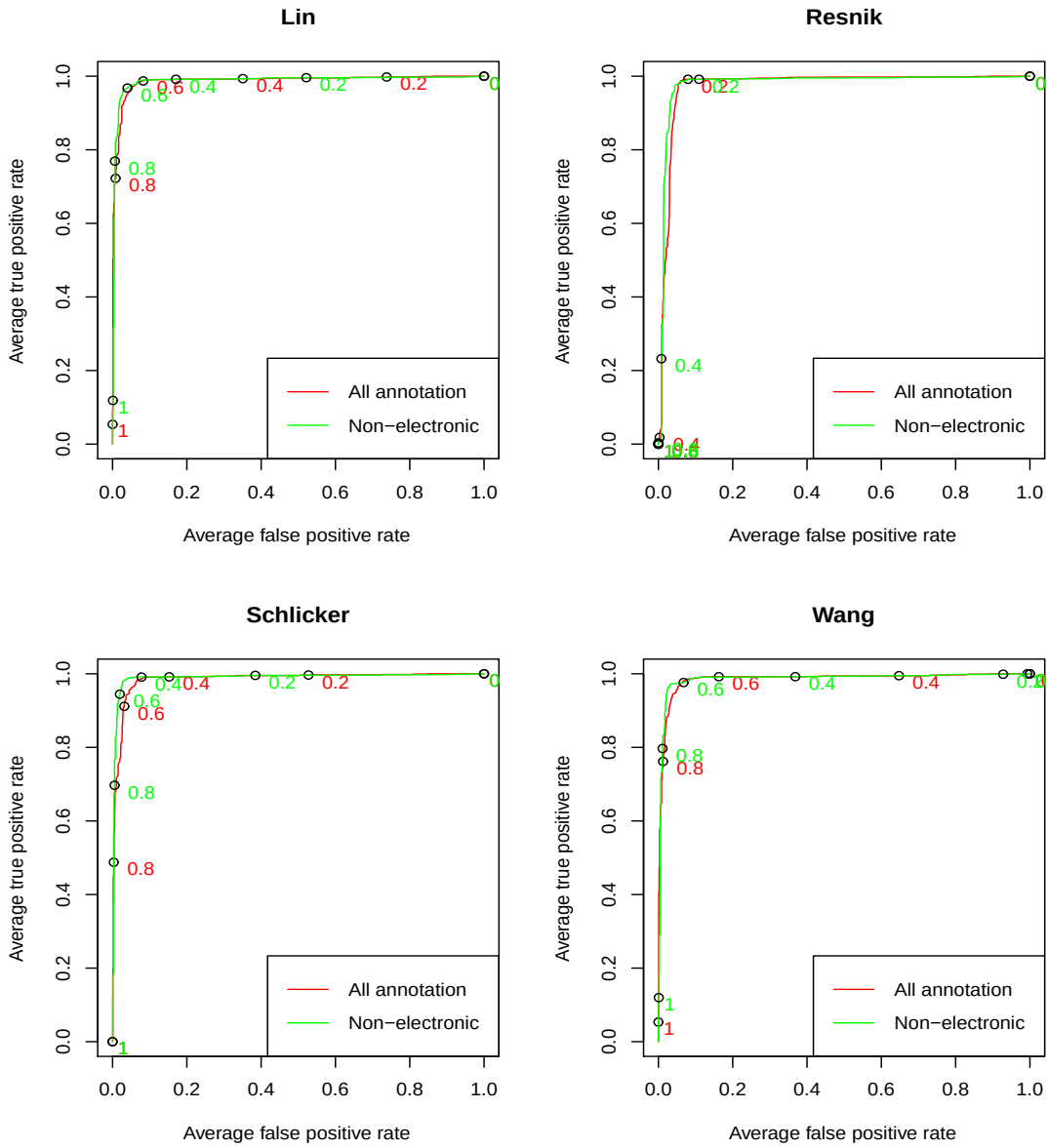

Figure A.4: ROC curves for all approaches for "BMA-MICA-rFunSim" with full and non-electronic annotation

Variables				AUCs			
Sem. sim.	*Func. sim.*	*Dataset*	*Ancestors*	*MF*	*BP*	*CC*	*rFunSim*
Lin	BMA	all	MICA	0.947	0.967	0.964	0.987
Lin	BMA	all	GraSM	0.959	0.946	0.965	0.980
Lin	BMA	nonIEA	MICA	0.969	0.940	0.982	0.988
Lin	BMA	nonIEA	GraSM	0.971	0.982	0.973	0.981
Lin	MAX	all	MICA	0.929	0.923	0.826	0.976
Lin	MAX	all	GraSM	0.972	0.835	0.878	0.956
Lin	MAX	nonIEA	MICA	0.977	0.956	0.905	0.990
Lin	MAX	nonIEA	GraSM	0.979	0.941	0.971	0.982
Resnik	BMA	all	MICA	0.967	0.889	0.960	0.975
Resnik	BMA	all	GraSM	0.963	0.861	0.955	0.973
Resnik	BMA	nonIEA	MICA	0.951	0.981	0.883	0.979
Resnik	BMA	nonIEA	GraSM	0.961	0.868	0.986	0.980
Resnik	MAX	all	MICA	0.957	0.756	0.978	0.973
Resnik	MAX	all	GraSM	0.966	0.720	0.991	0.979
Resnik	MAX	nonIEA	MICA	0.958	0.758	0.979	0.970
Resnik	MAX	nonIEA	GraSM	0.965	0.761	0.988	0.977
Schlicker	BMA	all	MICA	0.960	0.969	0.963	0.985
Schlicker	BMA	all	GraSM	0.970	0.945	0.962	0.981
Schlicker	BMA	nonIEA	MICA	0.973	0.981	0.981	0.990
Schlicker	BMA	nonIEA	GraSM	0.978	0.980	0.991	0.991
Schlicker	MAX	all	MICA	0.973	0.902	0.986	0.991
Schlicker	MAX	all	GraSM	0.958	0.827	0.982	0.954
Schlicker	MAX	nonIEA	MICA	0.975	0.938	0.976	0.990
Schlicker	MAX	nonIEA	GraSM	0.977	0.944	0.992	0.991
Wang	BMA	all	NA	0.942	0.958	0.956	0.985
Wang	BMA	nonIEA	NA	0.968	0.909	0.978	0.986
Wang	MAX	all	NA	0.929	0.923	0.825	0.976
Wang	MAX	nonIEA	NA	0.972	0.960	0.900	0.988

Table A.1: AUCs for all experiments in the expression dataset

Ontology	*Sem. sim.*	*Func. sim.*	*Dataset*	*Ancestors*
MF	Lin	MAX	all	MICA
	Schlicker	MAX	all	GraSM
BP	Lin	BMA	nonIEA	GraSM
	Resnik	BMA	nonIEA	MICA
CC	Resnik	BMA	nonIEA	GraSM
	Resnik	MAX	all	MICA
	Resnik	MAX	all	GraSM
	Resnik	MAX	nonIEA	MICA
	Resnik	MAX	nonIEA	GraSM
	Schlicker	BMA	nonIEA	GraSM
	Schlicker	MAX	all	GraSM
	Schlicker	MAX	nonIEA	GraSM

Table A.2: Cases in which individual scores outperform aggregate scores

A.1.3 Semantic similarity approaches

The gene expression dataset shows some remarkable differences from the aggregate dataset in the performance of the semantic similarity approaches. Overall, Schlicker et al.'s approach still performs the best, but Resnik's approach performs the worst. The Wang measure ranks overall third when compared under MICA but highest when compared under GraSM.

The one exception to Resnik's poor performance is for "MAX-all-GraSM", when Resnik performs best while Schlicker performs worst. "MAX-all-GraSM" is also the case where the Schlicker measure performs better for MF and CC than for rFunSim. From Table A.1, it can be seen that the score the BP ontology alone is particularly poor for Schlicker in this case, which explains the poorer performance of rFunSim compared to MF and CC alone, since rFunSim is an aggregate of the individual scores. When considering the actual AUC values, it is clear that the reversal of the common trend is not due to a better performance by Resnik but to a poorer performance of the Schlicker approach.

	BMA		MAX			
	all	nonIEA	all	nonIEA		
		MICA			Total	*StDev*
Lin	1	2	2	1	**6**	*1*
Schlicker	3	1	1	2	**7**	*1*
Wang	2	3	3	3	**11**	*1*
Resnik	4	4	4	4	**16**	*0*

Table A.3: Semantic similarity approaches for MICA

	BMA		MAX			
	all	nonIEA	all	nonIEA		
	GraSM				Total	*StDev*
Wang	1	2	2	2	**7**	*1*
Schlicker	2	1	4	1	**8**	*1*
Lin	3	3	3	3	**12**	*0*
Resnik	4	4	1	4	**13**	*2*

Table A.4: Semantic similarity approaches for GraSM

	BMA				MAX					
	all		nonIEA		all		nonIEA			
	MICA	GraSM	MICA	GraSM	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	1	1	1	3	2	1	**12**	*1*
Lin	1	2	2	2	2	2	1	2	**14**	*0*
Resnik	3	3	3	3	3	1	3	3	**22**	*1*

Table A.5: Semantic similarity approaches, all combinations

	BMA		MAX			
	all					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	1	3	**7**	*1*
Lin	1	2	2	2	**7**	*1*
Resnik	3	3	3	1	**10**	*1*

Table A.6: Semantic similarity approaches, full dataset

	BMA		MAX			
	nonIEA					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	1	1	2	1	**5**	*1*
Lin	2	2	1	2	**7**	*1*
Resnik	3	3	3	3	**12**	*0*

Table A.7: Semantic similarity approaches, non-IEA

	BMA					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	1	1	**5**	*1*
Lin	1	2	2	2	**7**	*1*
Resnik	3	3	3	3	**12**	*0*

Table A.8: Semantic similarity approaches, BMA only

	MAX					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	1	3	2	1	**7**	*1*
Lin	2	2	1	2	**7**	*1*
Resnik	3	1	3	3	**10**	*1*

Table A.9: Semantic similarity approaches, MAX only

A.1.4 Annotation

Differences between the full dataset and the gene expression dataset can also be seen in respect to the annotation data used. While the full annotation data performs better for the aggregate dataset, the non-electronic annotation performs better in all cases for the gene expression dataset.

However, there are two notable exceptions, where an individual approach differs from the overall trend. For Schlicker's approach and MICA, the full annotation dataset is ranked highest overall in conjunction with MAX (Table A.10). For Resnik's approach and MAX, the highest performance is obtained for "all-GraSM".

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
BMA	nonIEA	MICA	2	1	2	2	**7**	*1*
MAX	nonIEA	MICA	1	4	3	1	**9**	*2*
MAX	all	MICA	4	3	1	4	**12**	*1*
BMA	all	MICA	3	2	4	3	**12**	*1*

Table A.10: All annotation-MICA vs. non-IEA-MICA

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
MAX	nonIEA	GraSM	1	3	1	1	**6**	*1*
BMA	nonIEA	GraSM	2	1	2	2	**7**	*1*
BMA	all	GraSM	3	4	3	3	**13**	*1*
MAX	all	GraSM	4	2	4	4	**14**	*1*

Table A.11: All annotation-GraSM vs. non-IEA-GraSM

A.1.5 Ancestors

For the gene expression dataset, the GraSM algorithm generally performs better with the non-electronic annotation data, whereas using the MICA generates a better performance with the full annotation data. This again differs from the overall trend of the aggregate dataset, where MICA usually outperforms GraSM. The main

disagreement with the overall trend is for Lin's approach, which ranks MICA higher than GraSM, regardless of the dataset.

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	all	MICA	1	2	2	**5**	*1*
BMA	all	GraSM	2	3	3	**8**	*1*
MAX	all	MICA	3	4	1	**8**	*2*
MAX	all	GraSM	4	1	4	**9**	*2*

Table A.12: All annotation - MICA vs. GraSM

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	nonIEA	GraSM	4	1	1	**6**	*2*
BMA	nonIEA	MICA	2	2	3	**7**	*1*
MAX	nonIEA	GraSM	3	3	2	**8**	*1*
MAX	nonIEA	MICA	1	4	4	**9**	*2*

Table A.13: Non-IEA - MICA vs. GraSM

A.1.6 Functional similarity approaches

The overall performance for the two functional similarity approaches is less clear-cut than any of the other parameters. BMA appears to perform better than MAX, if the MICA and non-electronic annotation are used, whereas MAX performs better for MICA and full annotation. These two trends are entirely reversed with the GraSM algorithm. In addition, Resnik's rankings disagree with the overall trend for MICA insofar that BMA always outperforms MAX for this approach.

For a given choice of annotation (either only all data or only non-electronic data), BMA also always performs better than MAX, although whether "BMA-MICA" or "BMA-GraSM" is ranked highest depends on the dataset.

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	nonIEA	GraSM	3	1	1	**5**	*1*
BMA	nonIEA	MICA	1	2	2	**5**	*1*
BMA	all	MICA	2	3	3	**8**	*1*
BMA	all	GraSM	4	4	4	**12**	*0*

Table A.14: BMA only

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	nonIEA	GraSM	2	2	1	**5**	*1*
MAX	nonIEA	MICA	1	4	3	**8**	*2*
MAX	all	MICA	3	3	2	**8**	*1*
MAX	all	GraSM	4	1	4	**9**	*2*

Table A.15: MAX only

A.1.7 Conclusions

Most of the ROC curves shown for the gene expression dataset are very close and without analysing the AUC indices, it is difficult to draw any definite conclusions. It is clear from the AUC data that the general trends for the gene expression dataset are very different from the aggregate dataset. In terms of semantic similarity approaches, Schlicker's approach performs the best, and Resnik's approach performs the worst on the gene expression dataset. Lin ranks equal to Wang's approach overall, as they are each once ranked highest and third respectively, depending on whether the ancestor choice is MICA or GraSM.

Performance is better with the non-electronic annotation data than with all annotations, which is in direct contradiction with the aggregate dataset's trend. The MICA, which always performs best in the aggregate dataset, only performs better than GraSM on the full annotation data whereas the reverse is true for non-electronic annotations. Although the BMA functional similarity approach performs better in many cases, there are cases where MAX performs better, so no overall conclusion can be drawn for this.

A.2 Protein interaction dataset

The dataset of all recorded protein-protein interactions in yeast was downloaded from the SGD website. From these, the subset of interactions that were determined by mass spectroscopy affinity measure and manually curated were selected. Of these, about 1900 pairs of gene products were actually found in the Eisen dataset. Of the final 1900, 1745 were randomly selected for the positive dataset. The negative dataset was made up of an equal number of randomly selected gene product pairs that were not present at all in the full interaction dataset.

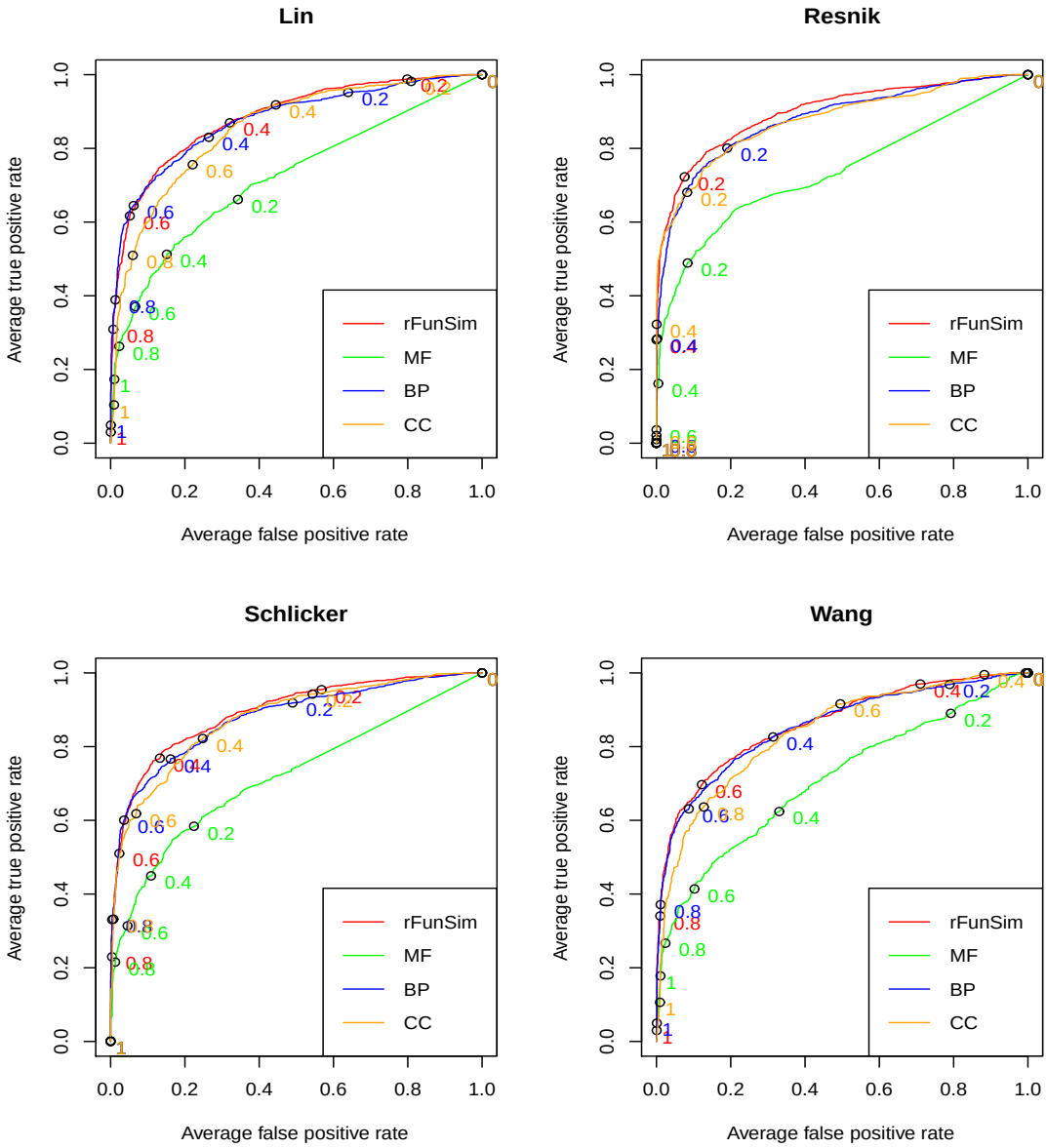

Figure A.5: ROC curves for individual ontology and rFunSim scores for all approaches with "BMA-MICA'

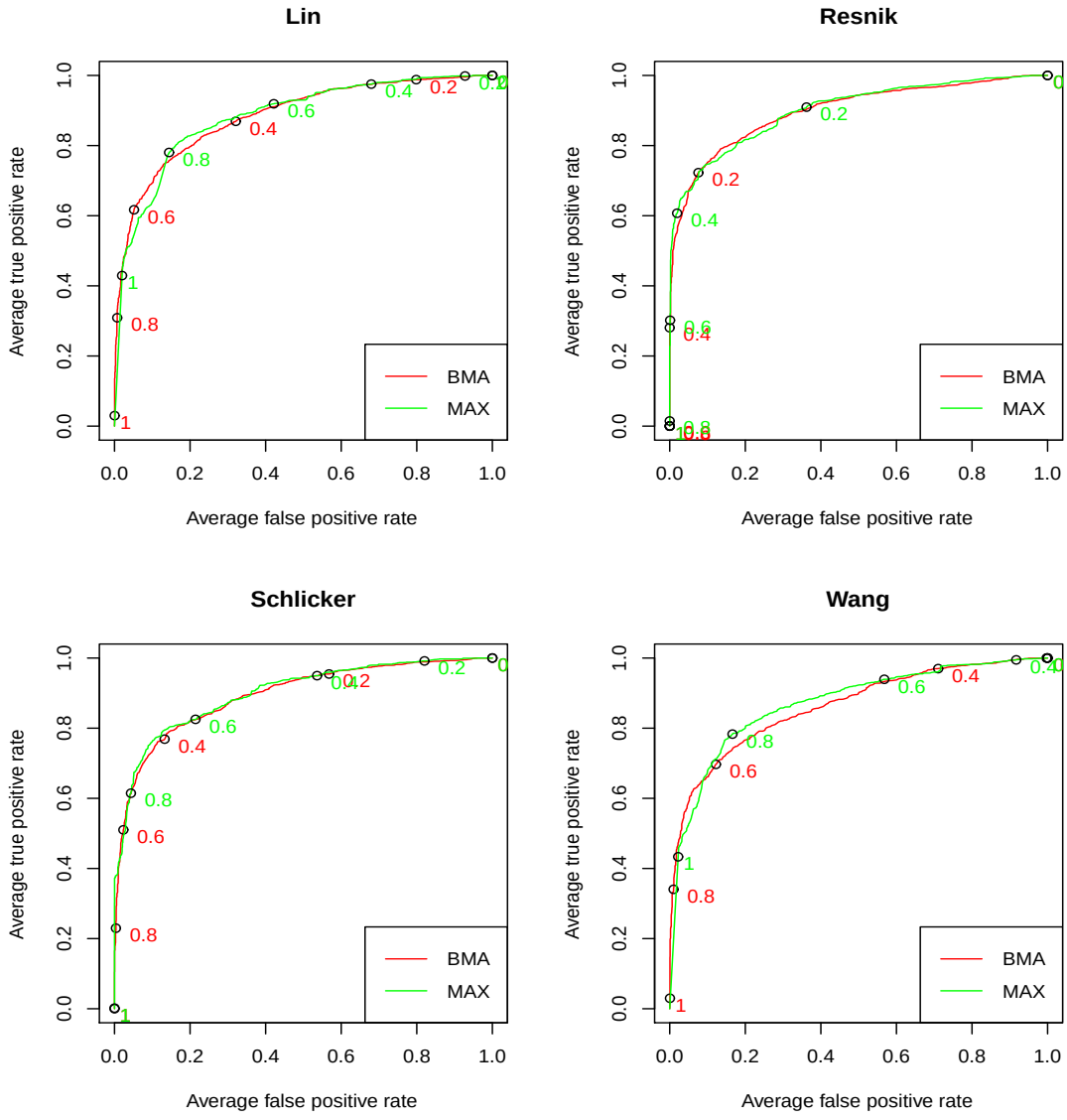

Figure A.6: ROC curves for all approaches for BMA and MAX with "all-MICA-rFunSim"

A.2.1 ROC curves

The ROC curves for the protein interaction dataset, shown in Figures A.5 to A.8, are fairly similar in overall trend to the curves for the aggregate dataset. This suggests that while protein interaction is a very good match to functional similarity, the two concepts are not as closely related as gene expression and functional similarity. This could be due either to biological reality or to the nature of the dataset; since the protein interaction dataset is based on manually curated, but unquantified interaction pairs, it is possible that some of the true positives have a lower similarity than the true positives in the gene expression dataset.

Comparison of the four semantic similarity approaches previously showed (Figure 4.2, Protein interaction dataset, bottom left) that for "BMA-all-MICA-rFunSim", Wang's approach definitely performs the worst, while Lin's approach is also worse than Resnik and Schlicker for large parts of the graph.

Out of the individual and aggregate scores (Figure A.5), MF definitely performs worst for all approaches, with a ROC curve that is significantly lower than any of the others. rFunSim appears to generally perform best, although some of the curves are too close to provide a conclusive result without consulting the AUC indexes.

The differences between the curves are even less clear for the comparisons between BMA and MAX (Figure A.6), where especially Resnik and Schlicker are simply too close to call. For Lin and Wang, BMA (red curve) appears to be performing better at higher thresholds, whereas MAX (green curve) shows a better performance further to the right (lower thresholds). Again, AUC indexes are required for conclusive results. As in the previous two sections, the MAX curves for Lin and Wang do not have the first threshold coinciding with the curve's origin but at around 0.4 on the Y-axis.

For the comparison of MICA and GraSM (Figure A.7), MICA appears to generally perform slightly better than GraSM although there is no large difference between the curves.

Finally, Figure A.8 shows the ROC curves for full and non-electronic annotation data. For all the IC-based measures, the full dataset (red curve) appears to be performing slightly better than the non-electronic dataset (green curve). The curves for the Wang approach are too close to draw any conclusions on the performance of the two datasets.

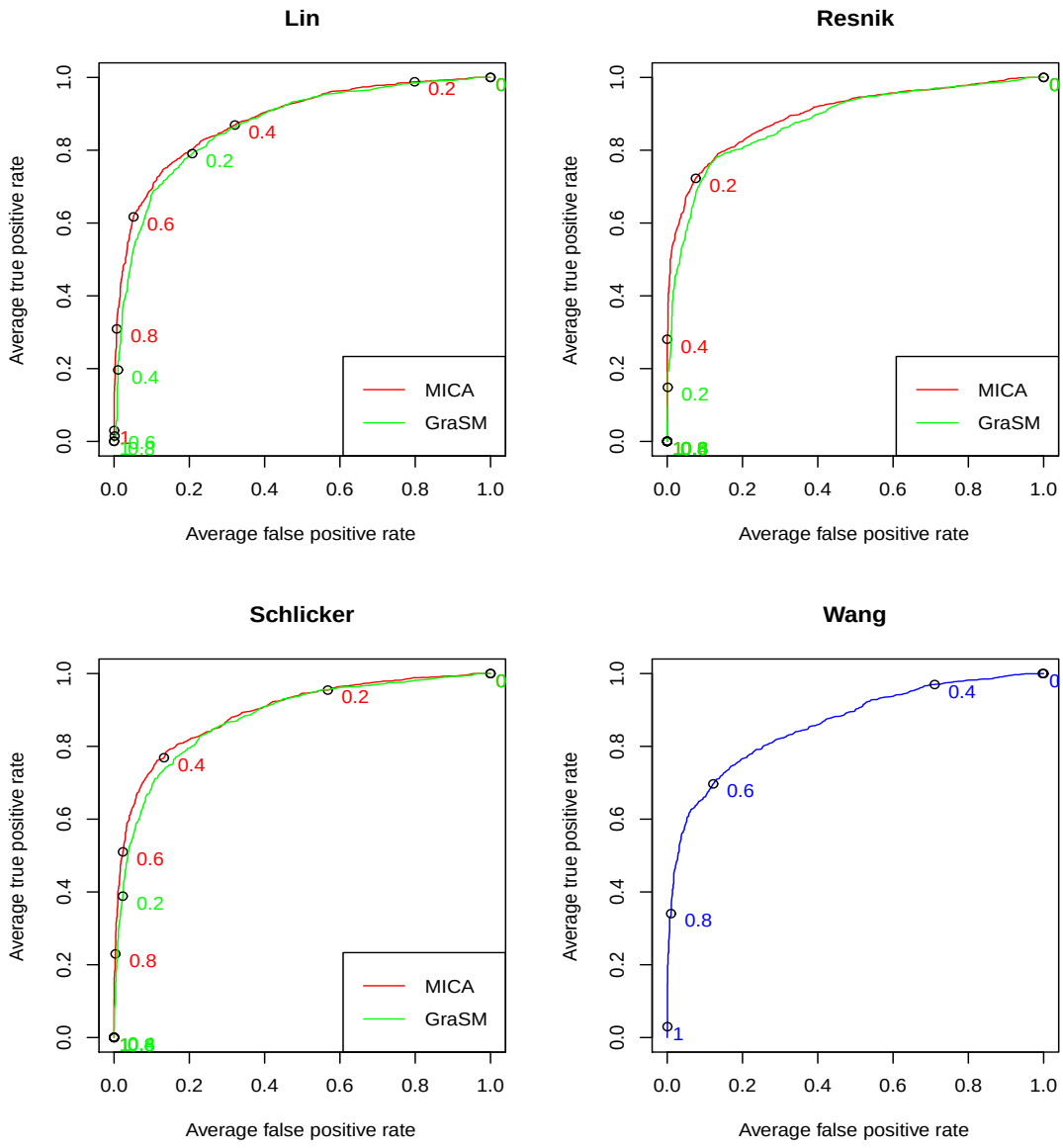

Figure A.7: ROC curves for all approaches for MICA and GraSM with "all-BMA-rFunSim"

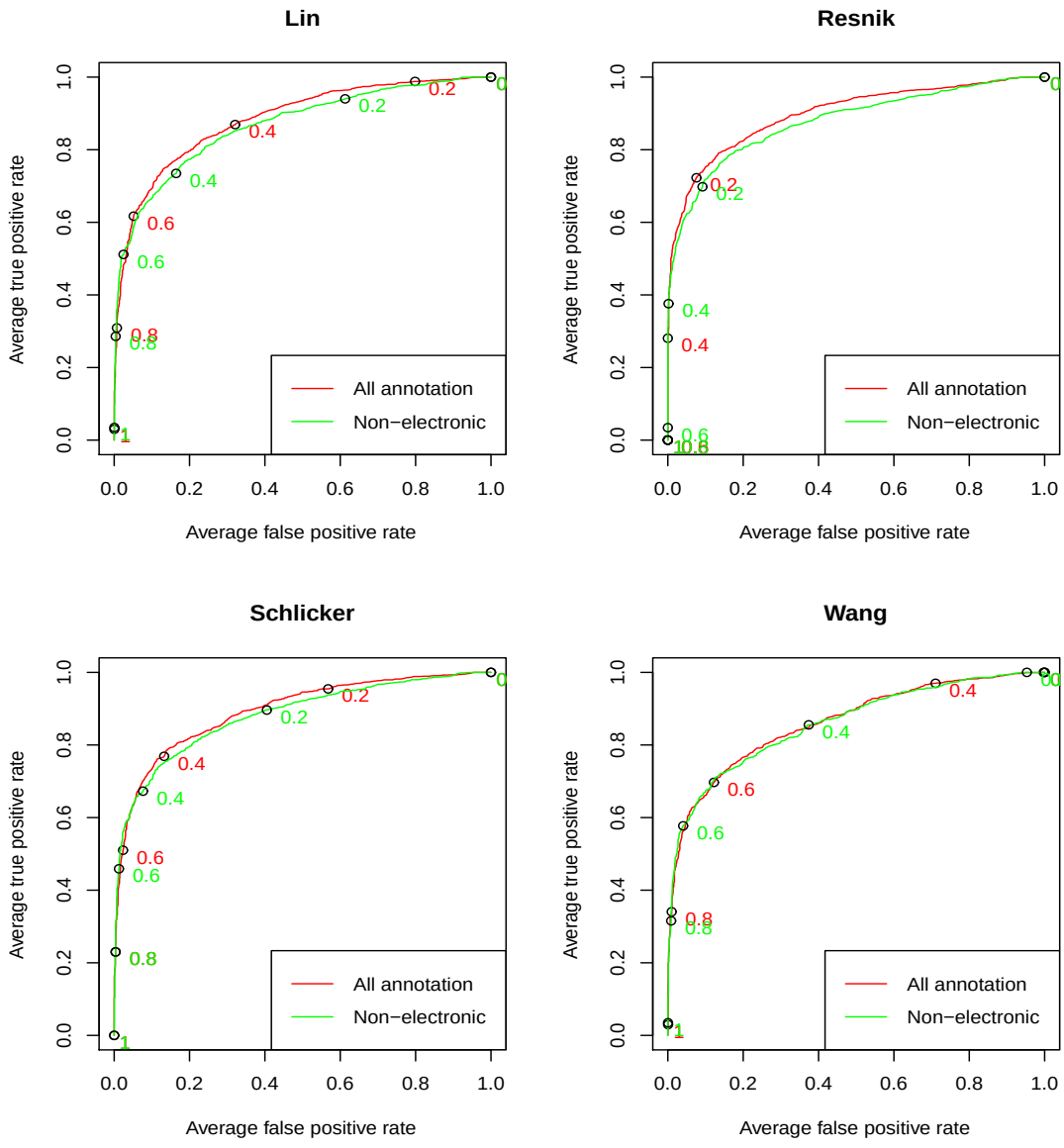

Figure A.8: ROC curves for all approaches for "BMA-MICA-rFunSim" with full and non-electronic annotation

A.2.2 AUC results

Table A.16 shows the AUCs for all combinations of semantic and functional similarity and the dataset and ancestor choices for the protein interaction dataset. Again, rFunSim performs better than the individual ontology scores in most cases. The only exception, listed in Table A.17, is the CC score for "Schlicker-MAX-all-GraSM".

Variables				AUCs			
Sem. sim.	*Func. sim.*	*Dataset*	*Ancestors*	*MF*	*BP*	*CC*	*rFunSim*
Lin	BMA	all	LCA	0.721	0.876	0.855	0.884
Lin	BMA	all	GraSM	0.719	0.856	0.845	0.869
Lin	BMA	nonIEA	LCA	0.722	0.826	0.866	0.867
Lin	BMA	nonIEA	GraSM	0.718	0.852	0.817	0.863
Lin	MAX	all	LCA	0.727	0.843	0.776	0.882
Lin	MAX	all	GraSM	0.725	0.826	0.836	0.854
Lin	MAX	nonIEA	LCA	0.732	0.859	0.816	0.882
Lin	MAX	nonIEA	GraSM	0.724	0.829	0.808	0.864
Resnik	BMA	all	LCA	0.736	0.878	0.878	0.899
Resnik	BMA	all	GraSM	0.718	0.869	0.867	0.882
Resnik	BMA	nonIEA	LCA	0.702	0.857	0.862	0.879
Resnik	BMA	nonIEA	GraSM	0.6975	0.850	0.849	0.864
Resnik	MAX	all	LCA	0.7425	0.868	0.880	0.901
Resnik	MAX	all	GraSM	0.736	0.850	0.868	0.884
Resnik	MAX	nonIEA	LCA	0.712	0.861	0.868	0.890
Resnik	MAX	nonIEA	GraSM	0.706	0.832	0.857	0.870
Schlicker	BMA	all	LCA	0.719	0.878	0.875	0.893
Schlicker	BMA	all	GraSM	0.720	0.862	0.870	0.878
Schlicker	BMA	nonIEA	LCA	0.709	0.850	0.867	0.881
Schlicker	BMA	nonIEA	GraSM	0.706	0.855	0.848	0.874
Schlicker	MAX	all	LCA	0.741	0.874	0.884	0.899
Schlicker	MAX	all	GraSM	0.723	0.838	0.865	0.863
Schlicker	MAX	nonIEA	LCA	0.718	0.869	0.862	0.892
Schlicker	MAX	nonIEA	GraSM	0.711	0.834	0.849	0.870
Wang	BMA	all	NA	0.708	0.854	0.838	0.862
Wang	BMA	nonIEA	NA	0.729	0.825	0.851	0.859
Wang	MAX	all	NA	0.726	0.828	0.764	0.870
Wang	MAX	nonIEA	NA	0.746	0.849	0.812	0.878

Table A.16: AUCs for all experiments in the protein interaction dataset

A.2.3 Semantic similarity approaches

With respect to the semantic similarity approaches, the protein interaction dataset shows fairly similar trends to the overall dataset. The approaches by Resnik and

Ontology	*Sem. sim.*	*Func. sim.*	*Dataset*	*Ancestors*
CC	Schlicker	MAX	all	GraSM

Table A.17: Cases in which individual scores outperform aggregate scores

Schlicker perform best, although Resnik actually performs the same as Schlicker, while Schlicker performs better than Resnik for the aggregate dataset. It should be noted that Resnik always performs best for the full dataset but Schlicker always performs best for the non-electronic dataset. If the common ancestor is selected using MICA, Lin's approach outperforms Wang. For GraSM, Wang performs better than Lin; in fact, Wang is ranked highest for "MAX-nonIEA" in this case. Overall however, Wang's approach performs no better than Lin.

	BMA		MAX			
	all	nonIEA	all	nonIEA		
			MICA		Total	*StDev*
Resnik	1	2	1	2	**6**	*1*
Schlicker	2	1	2	1	**6**	*1*
Lin	3	3	3	3	**12**	*0*
Wang	4	4	4	4	**16**	*0*

Table A.18: Semantic similarity approaches for MICA

	BMA		MAX			
	all	nonIEA	all	nonIEA		
			GraSM		Total	*StDev*
Resnik	1	2	1	3	**7**	*1*
Schlicker	2	1	3	2	**8**	*1*
Wang	4	4	2	1	**11**	*2*
Lin	3	3	4	4	**14**	*1*

Table A.19: Semantic similarity approaches for GraSM

A.2.4 Annotation

For the annotation data, the full dataset performs better overall than the non-electronic dataset. Main exceptions to this include Lin's approach in conjunction with MAX and MICA, where the non-electronic dataset ranks better than the full one. The same applies to the Wang approach. Lin's rankings most closely resemble Schlicker's on the full dataset whereas they are identical to Resnik's on the non-electronic dataset.

	BMA				MAX					
	all		nonIEA		all		nonIEA			
	MICA	GraSM	MICA	GraSM	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	1	2	2	1	1	2	2	**12**	*1*
Schlicker	2	2	1	1	2	2	1	1	**12**	*1*
Lin	3	3	3	3	3	3	3	3	**24**	*0*

Table A.20: Semantic similarity approaches, all combinations

	BMA		MAX			
	all					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	1	1	1	**4**	*0*
Schlicker	2	2	2	2	**8**	*0*
Lin	3	3	3	3	**12**	*0*

Table A.21: Semantic similarity approaches, full dataset

	BMA		MAX			
	nonIEA					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	1	1	1	1	**4**	*0*
Resnik	2	2	2	2	**8**	*0*
Lin	3	3	3	3	**12**	*0*

Table A.22: Semantic similarity approaches, non-IEA

	BMA					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	1	2	2	**6**	*1*
Schlicker	2	2	1	1	**6**	*1*
Lin	3	3	3	3	**12**	*0*

Table A.23: Semantic similarity approaches, BMA only

	MAX					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	1	2	2	**6**	*1*
Schlicker	2	2	1	1	**6**	*1*
Lin	3	3	3	3	**12**	*0*

Table A.24: Semantic similarity approaches, MAX only

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
MAX	all	MICA	3	1	1	2	**7**	*1*
BMA	all	MICA	1	2	2	3	**8**	*1*
MAX	nonIEA	MICA	2	3	3	1	**9**	*1*
BMA	nonIEA	MICA	4	4	4	4	**16**	*0*

Table A.25: All annotation-MICA vs. non-IEA-MICA

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
BMA	all	GraSM	1	2	1	3	**7**	*1*
MAX	nonIEA	GraSM	2	3	3	1	**9**	*1*
MAX	all	GraSM	4	1	4	2	**11**	*2*
BMA	nonIEA	GraSM	3	4	2	4	**13**	*1*

Table A.26: All annotation-GraSM vs. non-IEA-GraSM

A.2.5 Ancestors

In terms of ancestor selection, the single ancestor selection always performs better than the disjoint ancestor selection. This trend is consistent with that found for the aggregate dataset. Although for both annotation datasets, no single case is different enough to merit discussion, it should be noted that Resnik and Schlicker et al. show the same behaviour for MICA whereas they differ for GraSM.

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	all	MICA	2	1	1	**4**	*1*
BMA	all	MICA	1	2	2	**5**	*1*
BMA	all	GraSM	3	4	3	**10**	*1*
MAX	all	GraSM	4	3	4	**11**	*1*

Table A.27: All annotation - MICA vs. GraSM

A.2.6 Functional similarity approaches

For the full annotation dataset, MAX overall performs better than BMA,except for “all-GraSM”, when BMA performs best. For the non-electronic dataset, MAX performs overall better than BMA in all cases. Two approaches disagree with this ranking in relation to GraSM: for Resnik, MAX performs best for “all-GraSM”, unlike the other approaches, whereas for Schlicker, BMA performs better overall than MAX if GraSM is used. Wang’s approach on the other hand always performs best in conjunction with MAX.

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	nonIEA	MICA	1	1	1	**3**	*0*
BMA	nonIEA	MICA	2	2	2	**6**	*0*
MAX	nonIEA	GraSM	3	3	4	**10**	*1*
BMA	nonIEA	GraSM	4	4	3	**11**	*1*

Table A.28: Non-IEA - MICA vs. GraSM

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	all	MICA	1	1	1	**3**	*0*
BMA	all	GraSM	2	2	3	**7**	*1*
BMA	nonIEA	MICA	3	3	2	**8**	*1*
BMA	nonIEA	GraSM	4	4	4	**12**	*0*

Table A.29: BMA only

A.2.7 Conclusions

Like the previous two cases, the ROC curves for the protein interaction dataset for most sets of variables are very close although not quite as close as for the gene expression dataset. Nonetheless, the use of the AUC indexes is necessary in most cases to identify which variable results in the best performance. As for the aggregate dataset, the semantic similarity approaches by Schlicker et al. and Resnik perform better than those by Lin and Wang et al., although the individual rankings of the two best approaches differ. In terms of ancestor selection, MICA always performs better than GraSM, while the full dataset performs better than the non-electronic one, except for "MAX-GraSM". Finally, MAX performs better than BMA, except with "all-GraSM".

A.3 Phenotypes dataset

The full yeast phenotype dataset was downloaded from the SGD database. The similarity between gene products based on the phenotypes they are associated with was calculated using a vector space model with cosine normalisation (incl. Ref).

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	all	MICA	2	1	1	**4**	*1*
MAX	nonIEA	MICA	1	2	2	**5**	*1*
MAX	nonIEA	GraSM	3	4	3	**10**	*1*
MAX	all	GraSM	4	3	4	**11**	*1*

Table A.30: MAX only

Before calculation, the following restrictions were made:
- Phenotype data resulting from knockouts (null phenotype) only
- All experiment types
- Strain background "288C" only
- Excluding overly common phenotypes (associated with more than 250 gene products), such "viable", "inviable" etc. as they were deemed to be too uninformative due to their frequency
- Only gene products associated with more than 2 phenotypes (many of the gene products associated with only 1 or 2 phenotypes were already eliminated by the previous limitation point)

After the similarity between all pairs of the remaining gene products had been calculated, 2000 gene products with a phenotype similarity of 100% were selected as the positive dataset. A negative dataset of equal size was randomly selected from all gene product pairs with phenotype similarity of 0.

A.3.1 ROC curves

The ROC curves for the phenotype dataset are visibly lower than the ROC curves for any of the other datasets although they are still at an acceptable level above the "random-guess" line. This is most likely due to the nature of the phenotype dataset. As mentioned before, some phenotypes are fairly generic and can result from the non-expression of many genes, so although the most common and non-informative phenotypes have been removed, 100% phenotype-based similarity between two gene products is still no guarantee for a true high functional similarity.

Previously, Figure 4.2 (Phenotype dataset, bottom right) showed the most clear separation of all the ROC curves for all four semantic similarity approaches, out of the three individual and the aggregate datasets. Without recourse to the AUC values, it can be stated with high confidence that Resnik's approach performs best for this dataset and for "BMA-all-MICA-rFunSim", followed by Schlicker's approach. Lin's approach ranks a close third while Wang's approach clearly performs worst.

The case of the annotation comparison is somewhat clearer (Figure A.12), although most of the curves are still very close. The three IC-based methods appear to perform slightly better with the full annotation dataset. The two curves for the Wang approach are too close to draw conclusions without the AUC analysis.

For the ontology versus aggregate scores (Figure A.9), the MF score appears to perform worst for all approaches, although this is not as clear as for the protein interaction dataset. For all approaches apart from Resnik, the BP and rFunSim

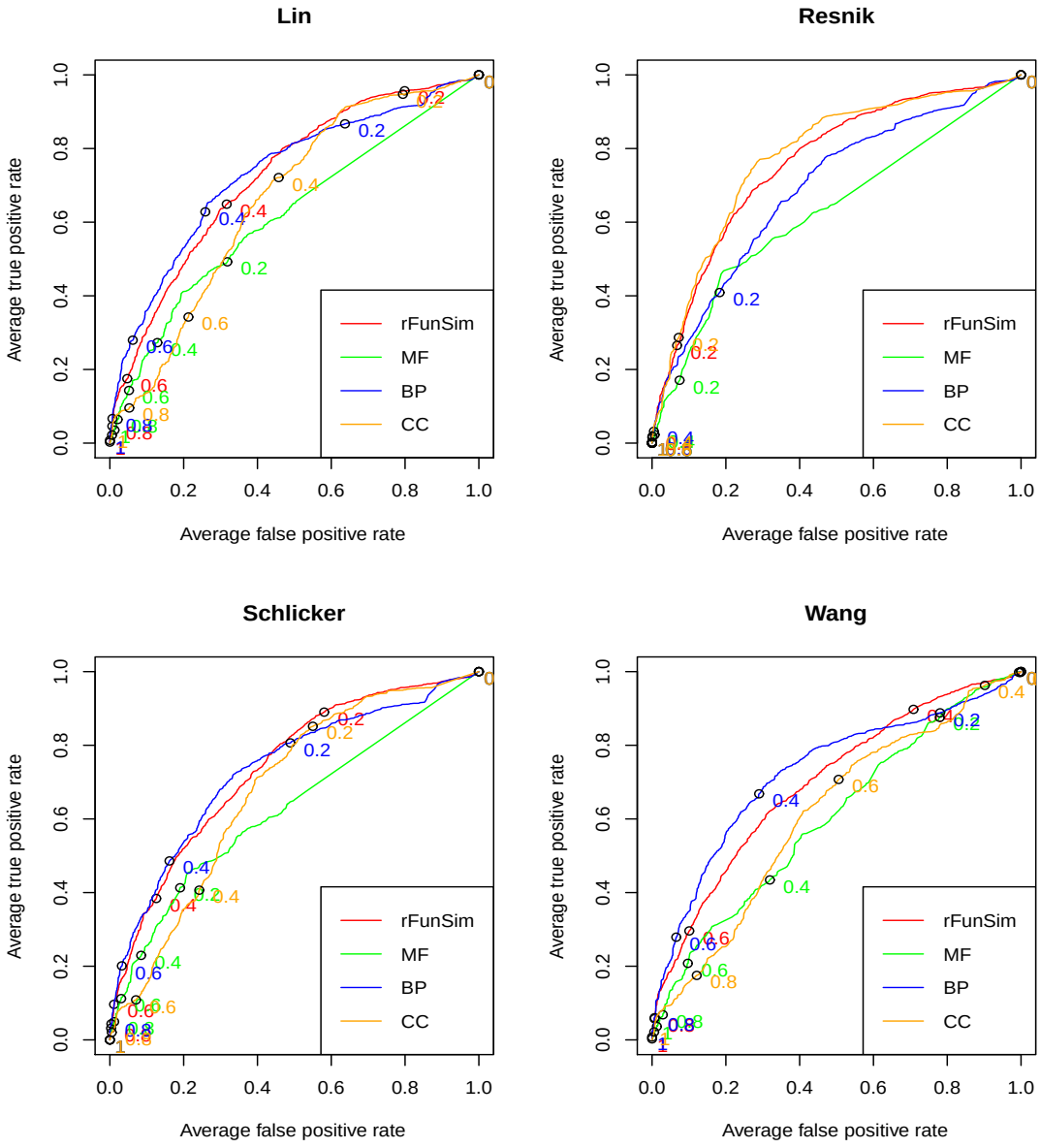

Figure A.9: ROC curves for individual ontology and rFunSim scores for all approaches with "BMA-MICA'

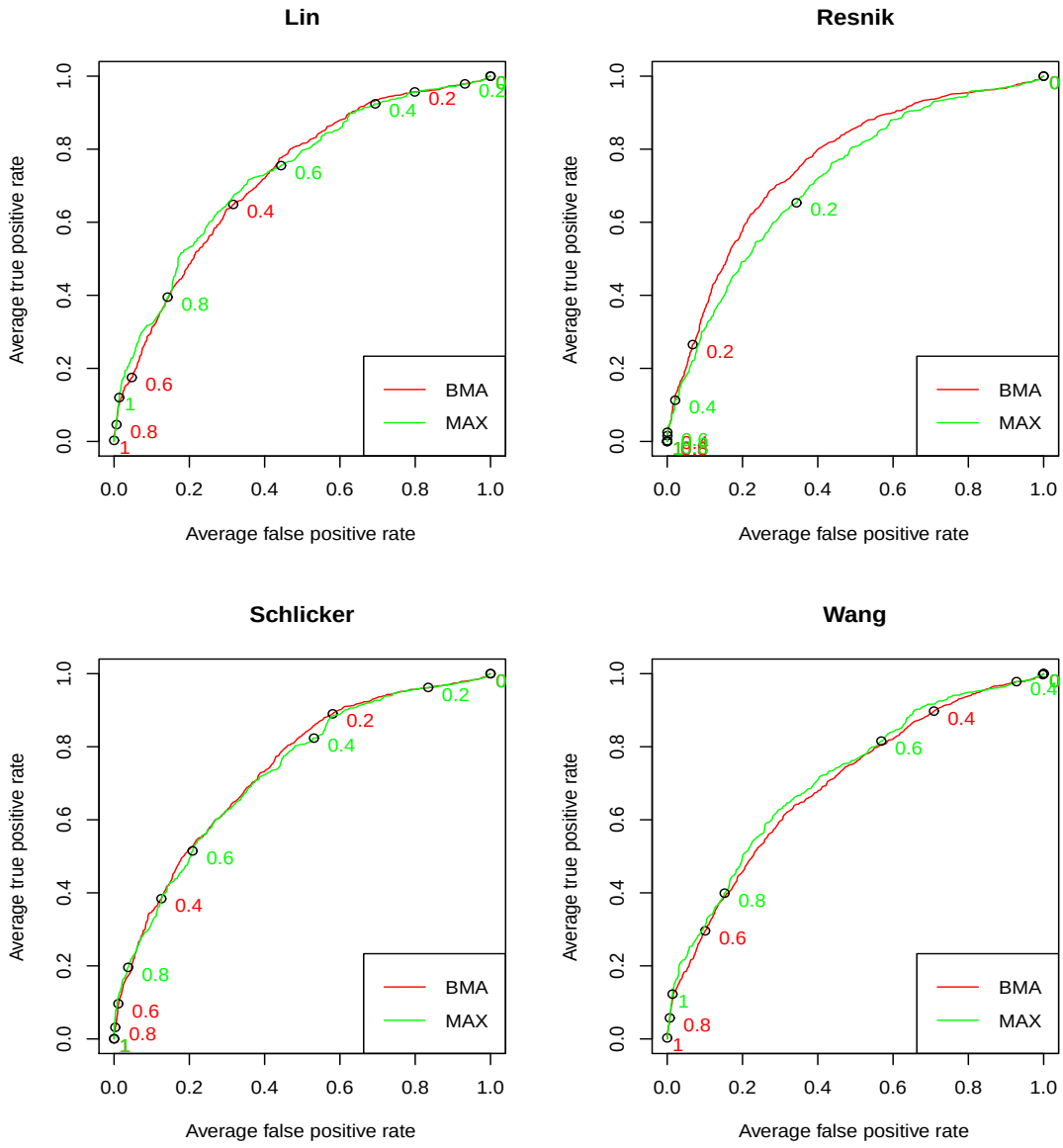

Figure A.10: ROC curves for all approaches for BMA and MAX with "all-MICA-rFunSim"

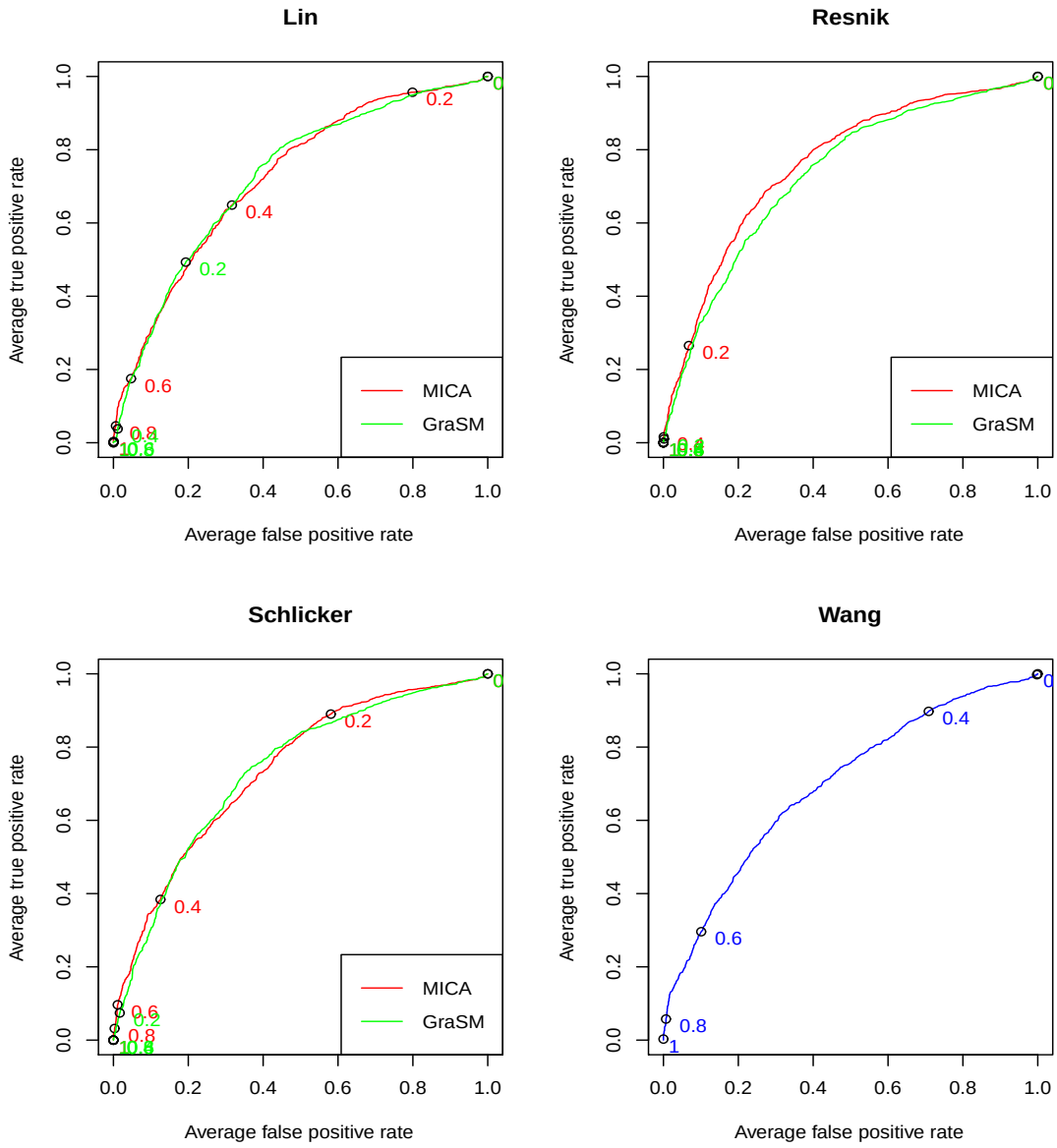

Figure A.11: ROC curves for all approaches for MICA and GraSM with "all-BMA-rFunSim"

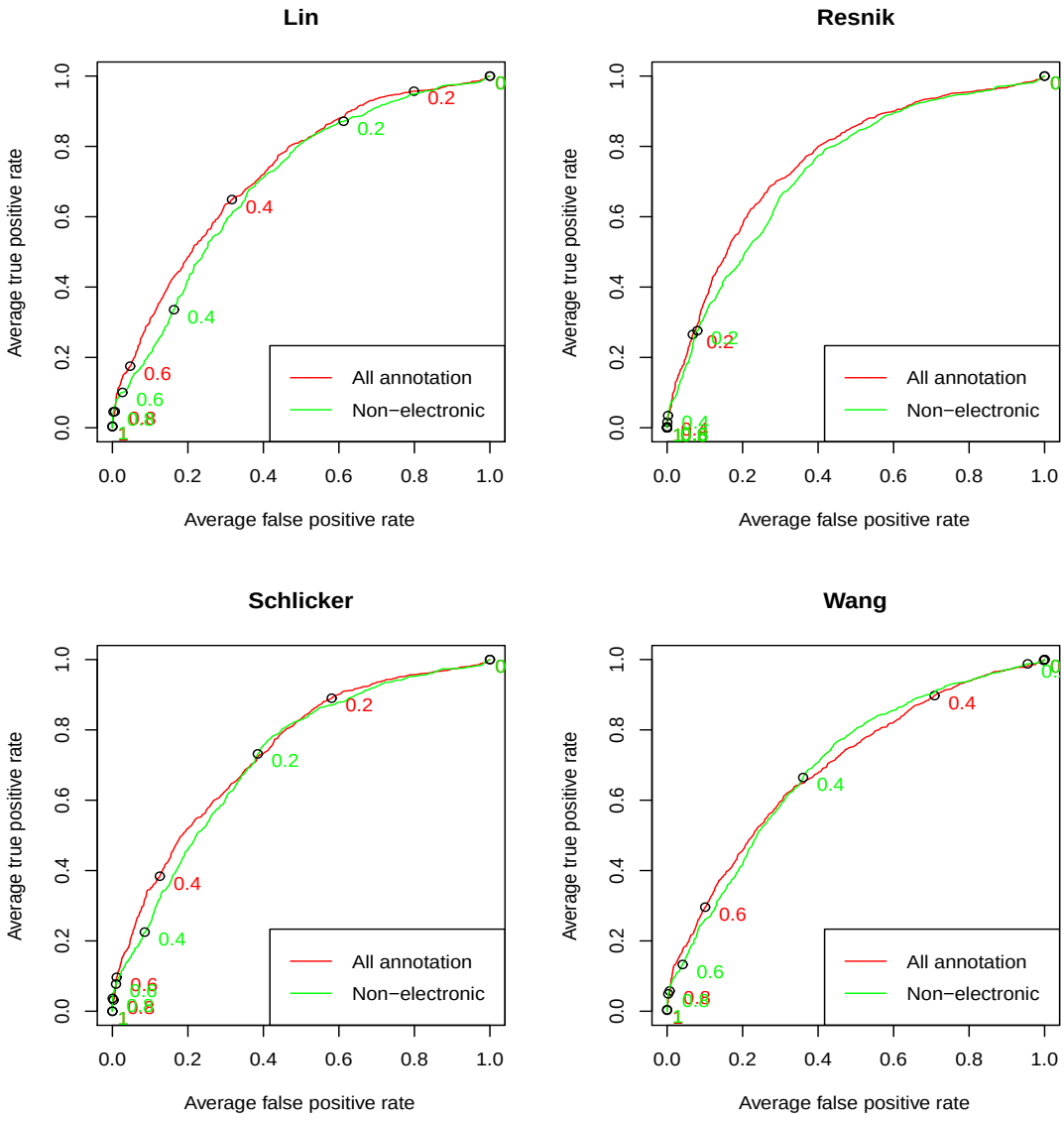

Figure A.12: ROC curves for all approaches for "BMA-MICA-rFunSim" with full and non-electronic annotation

scores are very close and their respective curves cross. For Resnik's approach on the other hand, the CC score seems to outperform rFunSim along a large part of the curve.

Figure A.10 shows that except for Resnik's approach, the curves for BMA and MAX are generally very close, too close for conclusive results. For Resnik on the other hand, BMA clearly performs much better than MAX. Again the MAX curves for Lin and Wang have their first threshold away from the origin, although the difference is much less pronounced than in the previous cases.

As far as the comparison between MICA and GraSM is concerned (Figure A.11), only Resnik's approach shows a discernible difference between the two curves, with MICA performing better than GraSM. For Lin and Schlicker, the two curves cross at least once but neither displays a visibly higher performance.

A.3.2 AUC results

Table A.31 shows the AUCs for all variable combinations for the phenotype data. It is notable that the phenotype AUCs are the lowest of all the AUCs for any dataset, with an average AUC for rFunSim of 0.716, compared to 0.858 for the aggregate dataset, 0.980 for expression and 0.877 for protein interaction. This result suggests that out of the three aspects selected for the comparison of the different semantic and functional similarity approaches, phenotype-based similarity between gene products is least comparable to annotation similarity. There are a number of possible explanations for this. It is possible that the phenotype annotation is simply of a lower quality, thus making for a poorer dataset. A more plausible explanation is however that similar phenotypes can be obtained in so many different ways that simple phenotype similarity does not automatically imply functional similarity. This is particularly true for very common phenotypes, which may appear to associate genes that do not in fact have any common functional aspects.

Nonetheless, the AUCs, particularly those obtained for rFunSim, suggest that phenotype similarity has some relation to functional similarity as even the worst AUC in Table A.31 (0.553, for "MF-Resnik-BMA-nonIEA-LCA") is better than a random-guess result. In fact, the highest AUC in the table (0.800, for "CC-Resnik-MAX-nonIEA-GraSM") is close to the levels of AUCs found for some of the individual ontological scores in the protein interaction and aggregate datasets.

For the phenotype dataset, rFunSim is outperformed by individual scores about 20% of the time (17 out of 84 cases), the worst performance out of the three datasets. Again, there are no overall trends, such as the same approach outperforming for all

Variables				AUCs			
Sem. sim.	*Func. sim.*	*Dataset*	*Ancestors*	*MF*	*BP*	*CC*	*rFunSim*
Lin	BMA	all	MICA	0.617	0.732	0.660	0.725
Lin	BMA	all	GraSM	0.617	0.723	0.638	0.725
Lin	BMA	nonIEA	MICA	0.561	0.632	0.723	0.696
Lin	BMA	nonIEA	GraSM	0.561	0.716	0.642	0.717
Lin	MAX	all	MICA	0.627	0.691	0.658	0.731
Lin	MAX	all	GraSM	0.627	0.687	0.667	0.706
Lin	MAX	nonIEA	MICA	0.568	0.698	0.637	0.698
Lin	MAX	nonIEA	GraSM	0.563	0.692	0.652	0.705
Resnik	BMA	all	MICA	0.630	0.691	0.777	0.761
Resnik	BMA	all	GraSM	0.620	0.687	0.757	0.732
Resnik	BMA	nonIEA	MICA	0.553	0.766	0.687	0.733
Resnik	BMA	nonIEA	GraSM	0.557	0.679	0.783	0.721
Resnik	MAX	all	MICA	0.635	0.639	0.760	0.718
Resnik	MAX	all	GraSM	0.635	0.633	0.758	0.698
Resnik	MAX	nonIEA	MICA	0.562	0.653	0.766	0.714
Resnik	MAX	nonIEA	GraSM	0.563	0.640	0.798	0.701
Schlicker	BMA	all	MICA	0.626	0.732	0.677	0.737
Schlicker	BMA	all	GraSM	0.626	0.719	0.667	0.733
Schlicker	BMA	nonIEA	MICA	0.561	0.718	0.676	0.717
Schlicker	BMA	nonIEA	GraSM	0.560	0.712	0.706	0.729
Schlicker	MAX	all	MICA	0.637	0.690	0.682	0.729
Schlicker	MAX	all	GraSM	0.631	0.681	0.676	0.697
Schlicker	MAX	nonIEA	MICA	0.567	0.692	0.681	0.709
Schlicker	MAX	nonIEA	GraSM	0.566	0.681	0.733	0.709
Wang	BMA	all	NA	0.6015	0.726	0.610	0.697
Wang	BMA	nonIEA	NA	0.663	0.580	0.740	0.699
Wang	MAX	all	NA	0.612	0.692	0.641	0.714
Wang	MAX	nonIEA	NA	0.674	0.718	0.586	0.702

Table A.31: AUCs for all experiments in the phenotype dataset

Ontology	*Sem. sim.*	*Func. sim.*	*Dataset*	*Ancestors*
BP	Lin	BMA	all	MICA
	Lin	BMA	nonIEA	GraSM
	Lin	MAX	nonIEA	MICA
	Resnik	BMA	nonIEA	MICA
	Schlicker	BMA	nonIEA	MICA
	Wang	BMA	all	NA
	Wang	MAX	nonIEA	NA
CC	Lin	BMA	nonIEA	MICA
	Resnik	BMA	all	MICA
	Resnik	BMA	all	GraSM
	Resnik	BMA	nonIEA	GraSM
	Resnik	MAX	all	MICA
	Resnik	MAX	all	GraSM
	Resnik	MAX	nonIEA	MICA
	Resnik	MAX	nonIEA	GraSM
	Schlicker	MAX	nonIEA	GraSM
	Wang	BMA	nonIEA	NA

Table A.32: Cases in which individual scores outperform aggregate scores

three ontological scores or the same combination of variables outperforming for all semantic similarity approaches.

A.3.3 Semantic similarity approaches

The phenotype dataset shows overall the same trend as the aggregate dataset with regard to the semantic similarity approaches. Schlicker's approach performs best, with Resnik's ranked second and Lin and Wang third and fourth respectively. Resnik generally performs better with MICA and Schlicker with GraSM, except for "MAX-all" in Resnik's case, when Resnik actually performs worse than Lin and "MAX-nonIEA" in Schlicker's case, when Schlicker perform's worst overall and Wang tops the ranking.

	BMA		MAX			
	all	nonIEA	all	nonIEA		
			MICA		Total	*StDev*
Resnik	1	1	3	1	**6**	*1*
Schlicker	2	2	2	2	**8**	*0*
Lin	3	4	1	4	**12**	*1*
Wang	4	3	4	3	**14**	*1*

Table A.33: Semantic similarity approaches for MICA

	BMA		MAX			
	all	nonIEA	all	nonIEA		
	GraSM				Total	*StDev*
Schlicker	1	1	1	4	**7**	*2*
Lin	3	3	2	2	**10**	*1*
Resnik	2	2	4	3	**11**	*1*
Wang	4	4	3	1	**12**	*1*

Table A.34: Semantic similarity approaches for GraSM

	BMA				MAX					
	all		nonIEA		all		nonIEA			
	MICA	GraSM	MICA	GraSM	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	2	1	2	1	2	3	**14**	*1*
Resnik	1	2	1	2	3	3	1	2	**15**	*1*
Lin	3	3	3	3	1	2	3	1	**19**	*1*

Table A.35: Semantic similarity approaches, all combinations

	BMA		MAX			
	all					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Schlicker	2	1	2	1	**6**	*1*
Lin	3	3	1	2	**9**	*1*
Resnik	1	2	3	3	**9**	*1*

Table A.36: Semantic similarity approaches, full dataset

	BMA		MAX			
	nonIEA					
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	2	1	2	**6**	*1*
Schlicker	2	1	2	3	**8**	*1*
Lin	3	3	3	1	**10**	*1*

Table A.37: Semantic similarity approaches, non-IEA

	BMA					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Resnik	1	2	1	2	**6**	*1*
Schlicker	2	1	2	1	**6**	*1*
Lin	3	3	3	3	**12**	*0*

Table A.38: Semantic similarity approaches, BMA only

	MAX					
	all		nonIEA			
	MICA	GraSM	MICA	GraSM	Total	*StDev*
Lin	1	2	3	1	**7**	*1*
Schlicker	2	1	2	3	**8**	*1*
Resnik	3	3	1	2	**9**	*1*

Table A.39: Semantic similarity approaches, MAX only

A.3.4 Annotation

As with the aggregate and protein interaction datasets, the full annotation data gives better results than the non-electronic dataset, except for "MAX-GraSM". Minor exceptions include Resnik's approach, which, with MICA, performs better for BMA, regardless of dataset, and Wang's approach, which does the same for MAX. If the functional similarity approach is the constant, Resnik's approach also always performs best for MICA, regardless of annotation.

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
MAX	all	MICA	1	3	2	1	**7**	*1*
BMA	all	MICA	2	1	1	4	**8**	*1*
BMA	nonIEA	MICA	4	2	3	3	**12**	*1*
MAX	nonIEA	MICA	3	4	4	2	**13**	*1*

Table A.40: All annotation-MICA vs. non-IEA-MICA

			Lin	Resnik	Schlicker	Wang	Total	*StDev*
BMA	all	GraSM	1	1	1	4	**7**	*2*
BMA	nonIEA	GraSM	2	2	2	3	**9**	*1*
MAX	all	GraSM	3	4	4	1	**12**	*1*
MAX	nonIEA	GraSM	4	3	3	2	**12**	*1*

Table A.41: All annotation-GraSM vs. non-IEA-GraSM

A.3.5 Ancestors

In terms of ancestor selection, the MICA performs better than the GraSM algorithm on the full annotation dataset, while for the non-electronic dataset, the reverse is true. This trend is particularly noticeable in Lin's approach, which performs better with GraSM on the non-electronic dataset. Resnik's approach on the other hand performs better with MICA, regardless of dataset.

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	all	MICA	3	1	1	**5**	*1*
BMA	all	GraSM	2	2	2	**6**	*0*
MAX	all	MICA	1	3	3	**7**	*1*
MAX	all	GraSM	4	4	4	**12**	*0*

Table A.42: All annotation - MICA vs. GraSM

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	nonIEA	GraSM	1	2	1	**4**	*1*
BMA	nonIEA	MICA	4	1	2	**7**	*2*
MAX	nonIEA	GraSM	2	4	3	**9**	*1*
MAX	nonIEA	MICA	3	3	4	**10**	*1*

Table A.43: Non-IEA - MICA vs. GraSM

A.3.6 Functional similarity approaches

Overall, BMA performs better than MAX. The only exception occurs for "MICA-all" and only if Wang's approach is included in the comparison, as Wang's approach always performs better in conjunction with MAX. As mentioned above, Resnik's approach always performs best with BMA.

			Lin	Resnik	Schlicker	Total	*StDev*
BMA	all	MICA	2	1	1	**4**	*1*
BMA	all	GraSM	1	3	2	**6**	*1*
BMA	nonIEA	MICA	4	2	4	**10**	*1*
BMA	nonIEA	GraSM	3	4	3	**10**	*1*

Table A.44: BMA only

A.3.7 Conclusions

Although the phenotype dataset clearly performs the worst out of the three dataset in terms of providing clear true and false positive datasets, the ROC curves are still good enough to draw conclusions about the overall behaviour of the various variables and many of the trends found are similar to those of one of the other two datasets.

In summary, Schlicker and Resnik perform the best out of the four semantic similarity approaches, the same trend as found for the aggregate dataset. The full annotation dataset performs better than the non-electronic one, except in conjunction with "MAX-GraSM". In terms of ancestor selection, MICA performs better in conjunction with the full annotation data, GraSM better in conjunction with

			Lin	Resnik	Schlicker	Total	*StDev*
MAX	all	MICA	1	1	1	**3**	*0*
MAX	nonIEA	GraSM	3	3	2	**8**	*1*
MAX	nonIEA	MICA	4	2	3	**9**	*1*
MAX	all	GraSM	2	4	4	**10**	*1*

Table A.45: MAX only

the non-electronic annotation data. BMA performs better than MAX, except for "all-MICA" and then only if Wang's approach is included.

Appendix B

In this appendix, a group-by-group analysis for each of the three evaluation datasets discussed in Chapter 8 is given. Figures and tables from the evaluation chapter were not replicated for space reasons.

B.1 Proteasome analysis

The proteasome dataset, discussed in Section 8.1, consists of 42 genes which were grouped into 5 supergroups and 38 original unmerged groups, of which 8 meet minimum size requirements. The grouping results are summarised in Figure 8.3 and Table 8.2.

The largest group in the resultset, supergroup 101, called "protein metabolic process", contains 31 genes, 30 of which are part of the official KEGG proteasome definition. Only one gene, PRD1, is included in the group but not the official proteasome definition. The two excluded genes are PRE2 and UMP1.

The inclusion of PRD1 and exclusion of PRE2 and UMP1 is a perfect illustration of the short comings of semantic and functional similarity. PRD1 is a zinc metalloendopeptidase active in the cytoplasm and the intermembrane space of mitochondria [Hrycyna and Clarke, 1993; Büchler et al., 1994]. Due to its location and proteolytic function, it shares several key annotations such as proteolysis (GO:0006508), peptidase activity (GO:0008233), hydrolase activity (GO:0016787) and cytoplasm (GO:0005737) with most proteasome genes. In addition, its only BP annotation is proteolysis. This leads to overall high functional similarity scores between this gene and proteasome genes. In fact, even functional similarity between PRD1 and PRE2 exceeds the FT.

The exact opposite of this problem exists for PRE2, the beta 5 subunit of the 20S proteasome. In the version of the GO used in this work (2009-04), the gene in question has four reproduction-related BP annotations, all labelled with the evidence

code "RCA" (inferred from Reviewed Computational Analysis) [Ashburner et al., 2000]. Three of these annotations are shared with no other genes in the dataset, so some of the functional similarity scores between PRE2 and other proteasome genes are lower than the FT, leading to PRE2's exclusion from the group. It should be noted that there is no indication in the functional description of PRE2 that it is involved in reproductive processes. These annotations are also absent in the latest version (2011-01) of the GO, suggesting that they have been removed as incorrect in one of the intervening versions.

Finally, UMP1, a short-lived chaperone required for correct maturation of the 20S proteasome, suffers from a different form of poor annotation. Although not directly part of the proteasome complex, KEGG still lists it as part of the proteasome pathway. First described by Ramos et al. [1998], UMP1 is referred to as "the best characterized" proteasome assembly factor by Li et al. [2007]. Yet despite clearly being addressed in more than one study, UMP1 has no MF annotation, removing one of three dimensions from its overall functional similarity score with any other gene product. In fact, UMP1 is absent from any of the very large groups in Figure 8.3 and it is also in an isolated location in the semantic tree. It is however, both in Eisen's version and in our clustering, clustered with the majority of proteasome genes in the expression tree, which is consistent with its role as a proteasome assembly factor.

The second-largest group in Figure 8.3 and Table 8.2 is an unmerged CC group called cytosol and it contains the same genes as the previous group except for PRD1. The reason for this exclusion is that while PRD1 has the required level of functional similarity to be in the group, it is not in fact annotated with either of this group's definition terms, which are cytosol (GO:0005829) and proteasome storage granule (GO:0034515). UMP1 and PRE2 are excluded from this group for the same reason as described above for group 102.

The third group in the list is peptidase activity, an unmerged MF group. Unlike the first two groups, it contains PRE2 but excludes RPN5, RPN6, RPN7 and RPN9. Non-proteasome genes PRD1 and STE24 are also included. The exclusion of the four proteasome lid (see Figure 8.1) components from this group is again due to poor annotation. Each of the four genes either has only one single MF annotation, or none, compared with the average of four for the 29 genes in the group. The one annotation is also very high level, with RPN6, RPN7 and RPN9 being annotated with structural molecule activity (GO:0005198), a direct descendant of the root node and RPN5 has no real MF annotation except the root term, molecular function. The fact that these genes are included in some of the other groups is an indicator of how closely the rest of their annotations match the average annotation profile of the

other proteasome genes. The reason for their exclusion from group 1047 is not an insufficiently high functional similarity, but the absence of any annotation that matches the group definition. In the other two ontological domains, where their annotation is more comprehensive, they meet the inclusion criteria for functional groups. The separation of these four genes from the main group of proteasome genes can also be observed in the semantic tree (see second column in Table 8.1, where they are found in the same cluster but far away from the main proteasome cluster.

The explanation for the inclusion of PRD1 in group 1047 is the same as for its inclusion in supergroup 102. A similar explanation applies for STE24, described as a highly conserved zinc metalloprotease that functions in two steps of a-factor maturation, C-terminal CAAX proteolysis and the first step of N-terminal proteolytic processing [Tam et al., 2001]. As is the case for PRD1, STE24 has a number of annotations referring to its proteolytic function that qualify it for inclusion in this group. In fact, the presence of these two genes in the dataset affects the group definition of group 1047 to the extent that two of the group's five definition terms are exclusive to these two genes and not part of any proteasome annotations. These terms are metallopeptidase activity (GO:0008237) and metalloendopeptidase activity (GO:0004222). This is a further indication that it is always advisable to consider both group content and group definition, rather than to blindly rely on the accuracy of the algorithm.

The next two groups are supergroups 102 and 104 which are both named proteasome complex. Their definitions overlap by one term out of four, namely proteasome complex, while in terms of content, they share four out of their 18 respective genes. From the definitions, it is clear that group 104 is based on the proteasome core complex (distinct definition terms: proteasome core complex, alpha-subunit complex (GO:0019773), proteasome core complex, beta-subunit complex (GO:0019774) and proteasome core complex (GO:0005839)) and group 102 is based on the 19S regulatory particle (distinct definition terms: proteasome regulatory particle, base subcomplex (GO:0008540), proteasome regulatory particle, lid subcomplex (GO:0008541) and proteasome regulatory particle (GO:0005838)). The genes the two groups share are PRE1, PRN1, RPN2, RPN3 and RPN10.

Based on its definition, group 104 should not contain any RPN genes as these are all part of the regulatory complex (see Figure 8.1). PRE1 on the other hand, the gene coding the β-4 subunit of the core complex, would not be expected in group 102. While the most likely explanation for this overlap is the overall similarity in annotation of the genes that should be in the group and the incorrectly grouped

genes, which leads to a functional similarity score that is higher than the FT, thereby allowing these genes into the group. It could also be argued that in the absence of the more general shared definition term of proteasome complex, this overlap in group content would not have occurred, as there would then have been no definition terms allowing PRE1 to be added to group 102 or the RPN genes to group 104. The semantic similarity between the proteasome complex and any of the other six terms concerned is always 0.54, so a much higher ST would have been necessary to eliminate the common term from the definitions. This ST would in fact have to be higher than the maximum FT derived for Resnik. Another consideration is whether the inappropriate terms might have been more appropriately separated in the original groups than in the supergroups. Analysing the group definitions of the original groups for supergroup 102 (groups 1026 and 1027) and supergroup 104 (groups 1043 and 1045) (see Figure 8.4) showed that in both cases, the original group definitions already contained the more general term proteasome complex. The merging of the original groups into supergroups did therefore not generate less appropriate groups.

Supergroup 103, the product of the merging of groups 1002 and 1023, is a BP group called "regulation of protein metabolic pathway (GO:0051246)" containing five proteasome genes, RPN1, RPN2, RPN3, PRE4 and SCL1, and one non-proteasome gene, UBC7. Since proteolysis, the key function of the proteasome, is a protein metabolic process and considering the function of the 19S regulatory particle, it is unsurprising to find the genes RPN1, RPN2 and RPN3 in this group. The same applies to UBC7 which, while not a proteasome subunit, is one of a number of ubiquitin-conjugation enzymes found in yeast. Without tagging with ubiquitin, proteins cannot be processed by the proteasome, so the ubiquitination mechanism is indeed a regulatory mechanism of proteasome-mediated proteolysis. It should however be noted that these three RPN genes are the only subunits of the regulatory particle that have an annotation related to the regulatory aspect of their function.

The presence of genes PRE4 and SCL1, which both code for subunits of the proteasome core complex, is more difficult to explain biologically. In terms of annotation, their presence in the group is due to the term "negative regulation of protein metabolic process" (GO:0051248), with which both genes are annotated. There is no aspect of their function that would confer them a regulatory role not shared by other core complex subunits. The evidence code for this association is the "RCA" code already associated with biologically inconsistent annotation in one of the other groups and this annotation is also no longer present in the latest version of the GO. Group 105 is therefore another example of potentially incorrect annotations affecting

the accuracy of the groups.

The next group in Table 8.2 is BP group 1036, called "cellular process" (GO:000987). This extremely generic (direct child of the branch root) name is the result of a very diverse definition consisting of three reproduction-related terms, cell differentiation (GO:0030154), sporulation resulting in formation of a cellular spore (GO:0030435) and ascospore formation (GO:0030437), and the term cell death (GO:0008219). The genes in the group are PRE1, PRE2, PRE3, PUP2, RPT4 and UFD1, i.e. four core complex subunits, one regulatory particle subunit and one gene that is not directly part of the proteasome. PRE2's questionable reproduction-related annotations have already been discussed in relation to another group. The same problem applies to the cell death annotation found for PUP2, RPT4 and UFD1. In all three cases, the association is qualified with the RCA evidence code. Although all the major biological processes in a cell are of course interlinked, there is no evidence available that directly links these genes to cell death. The reference cited for all these associations is the same as that given for the afore-mentioned PRE2 annotations, as well as most other RCA-based annotations in SGD, namely Huttonhower and Troyanskaya [2009].

In the version of GO used here, PRE1, PRE3 and PUP2 are all annotated with the term ascospore formation, under the evidence code TAS (Traceable Author Statement). The associated reference, Hochstrasser [1996], however does not provide any evidence for the involvement of these genes in ascospore formation. It only states "Required for sporulation [...]" as a comment in a list of proteasome genes, without reference or further discussion of this statement. In the latest version of the GO, this annotation has been withdrawn. Instead, the three genes are annotated with the term's parent sporulation resulting in formation of a cellular spore, and the evidence code RCA is referenced with Huttonhower and Troyanskaya [2009]. In addition, it has been shown [Heinemeyer et al., 1991] that for certain mutations of PRE1, one mutant phenotype includes the absence of sporulation, suggesting this subunit's involvement in the process. The actual role of PRE1 in this process has however not been described.

Considering the nature of the annotations at that gave rise to this group, the biological relevance of the group is questionable. In particular, the association of the cell death and reproduction annotations is due to the chosen ST. At ST40, these two aspects are assigned to different groups as the semantic similarity between cell death (GO:0008219) and the other three terms is 0.34, which is below the maximum ST.

The next two groups in Table 8.2 are groups 1025, nucleoside-triphosphate activ-

ity, and 1042, ATP binding. These two MF groups contain the same six genes (see Figure 8.3), RPT1, RPT2, RPT3, RPT4, RPT5 and RPT1. These genes each code for one of the 19S regulatory particle's six ATPase subunits, located in the particle's base unit, as shown in Figure 8.1. It is unsurprising that the two groups have identical content as an ATPase has to bind to ATP, although the reverse is not necessarily true. In fact, the proteasome dataset contains one further gene annotated with the term ATP binding, GSH2, which catalyses an ATP-dependent synthetic process but which is not an ATPase. GSH2 was not included in group 1042 as its functional similarity with the group's six genes is well below the minimum FT.

The reason groups 1025 and 1042 were not merged into a supergroup despite their identical content is due to their complete lack of definition overlap. In most cases where two groups contain the same genes, there is at least some level of overlap between their annotations. The two molecular functions covered in these two groups are however completely different on an ontological level, even though they refer to similar concepts. The semantic similarity of the definition terms from group 1025 (nucleoside-triphosphatase activity and ATPase activity) and ATP binding (single-term definition) is 0 as their only common ancestor is the MF root.

BP group 1038, cellular macromolecular complex assembly, contains the five genes UMP1, RPN2, RPN6, RPN9 and PRE4. The group's definition consists of the three terms proteasome assembly (GO:00432480), DNA recombinase assembly (GO:0000730) and meiotic DNA recombinase assembly (GO:0000707). Only the first of these three terms is annotated to the group's five genes. The other two terms are annotated to only one gene in the proteasome dataset, namely RAD52. The inclusion of RAD52's annotations in the definition, but exclusion of the gene from the group content is due to the ST and the FT values. While the semantic similarity between proteasome assembly and each of the two other GO terms is 0.41, i.e. higher than even the maximum ST, the functional similarity between RAD52 and most of the other genes is below the minimum FT, except for UMP1, with which it has a similarity of 0.18. This slightly higher functional similarity between RAD52 and UMP1 is due to some shared annotation not related to the proteasome pathway, but because of the maximal completeness rule requiring above FT similarity between all pairs of genes in a group, RAD52 is not included in group 1038.

Of the group's five genes, UMP1 and RPN6 most obviously deserve the proteasome assembly annotation, as their full descriptive names include this aspect of their functionality (see Table 8.1). In all five genes, the annotation with proteasome assembly is however qualified with an experimental evidence code of either IGI or IMP (see Section 2.1.2 for details) and a reference to a paper specifically describing

that gene's role in proteasome assembly. The references for UMP1 [Ramos et al., 1998; Li et al., 2007] were already cited earlier in this section. PRE4 and RPN2, the most recently documented, were studied together [Marques et al., 2007], while RPN6 [Santamaria et al., 2003] and RPN9 [Takeuchi et al., 1999] each had their role in proteasome assembly studied separately. From the literature, it is clear that the five genes in this group are indeed involved in proteasome assembly and should be grouped together. The inclusion of the two other GO terms in the group's definition suggests that a re-evaluation of the semantic thresholds may be necessary in order to avoid such inclusion. In addition, it would be useful to add an additional step to the algorithm to recheck each group's definition after all gene products are added, in order to ensure that the definition matches the group's content.

The final three groups in Table 8.2 which have not yet been discussed include one CC supergroup and two normal BP groups, each containing four genes. Group 105 is the previously mentioned result of the merger of groups 1029, 1035 and 1040. The CC supergroup, named cell part, contains the genes RPN1, UFD1, UBC7 and STE24, of which only the first is a proteasome subunit. In terms of definition, the supergroup consists of seven GO terms, three of which relate to the endoplasmic reticulum (ER), one to an ATPase complex, one to the nuclear inner membrane and two to a retromer complex. When comparing the definition terms to the annotations of the four genes, it is immediately clear that none of the four genes are annotated with either of the retromer complex terms. The only gene in the proteasome dataset which has this annotation is PEP8. Functional similarity between PEP8 and all genes in the group is lower than the minimum FT, which is why PEP8 is not included in the group. This group is therefore another good example of why the algorithm would benefit from a further step checking group content against definitions. It also means that although the merging of groups 1029, 1035 and 1040 into a supergroup was appropriate from a computational point of view, it did not improve the group's biological meaning. Group 1040 on its own was in fact a more suitable group, as it contained all the supergroup's definition terms, except for the two retromer complex terms and nuclear inner membrane, as well as its four genes.

The definition term "Cdc48p-Npl4p-Ufd1p AAA ATPase complex" is a child term of endoplasmic reticulum membrane, so does belong in a group based on ER-related cellular locations. Three of the four genes have clearly documented associations with the ER and its sub-components (UFD1 [Schuberth and Buchberger, 2005], UBC7 [Biederer et al., 1997], STE24 [Tam et al., 2001]). The annotation of RPN1 with endoplasmic reticulum is also quantified with a direct assay evidence code. The associated reference [Kumar et al., 2002] is to a high-throughput experiment which

identified the cellular locations of several thousand yeast genes. While it is not the purpose of this work to question the validity of individual gene annotations, in the absence of a more gene-specific annotation and considering that RPN1 is the only proteasome subunit with an ER-associated annotation, the biological relevance of the inclusion of RPN1 in this group is unclear.

The last two groups, group 1015 (reproduction) and group 1048 (response to stress), are both BP groups and differ in their content only by one gene. The three genes PRE1, PRE3 and PUP2 are included in both groups, with group 1015 additionally including PRE2 and group 1048 UMP1. Based on its definition and gene content, it is clear that group 1015 is the result of the questionable reproduction-related annotation of PRE1, PRE3 and PUP2 (ascospore formation), as well as PRE2 (reproduction, ascospore formation and reproductive cellular process), which was already discussed in the context of groups 102 and 1036.

Group 1048 on the other hand reflects an important function of the proteasome. In response to cellular stress such as heat shock, which causes an increase in misfolded or unfolded proteins, the activity of the ubiquitination and proteasome pathways is increased in order to facilitate the removal of the damaged proteins [Coux et al., 1996]. As is the case for other groups, there is some difference in the definition of group 1048 between the stress response terms annotated to the genes in the group and all the terms in the definition. In this case, only two out of the group's ten definition terms are actually associated with its content. The remaining eight terms come from the annotations of the genes RAD52 (seven terms) and GSH2 (one term), neither of which are included in the group due to their low functional similarity with the group's genes. The RAD52 annotations are all based on DNA repair processes, which is why they have sufficient semantic similarity with "response to stress" as DNA damage and the repair processes described by these GO terms are generally the result of external stress factors.

The term contributing to the definition through UMP1's annotations is "response to DNA damage stimulus". The induction of UMP1 expression by a number of DNA damaging agents is a documented fact [Mieczkowski et al., 2000]. The stress response annotation of PRE1, PRE3 and PUP2 on the other hand, whilst qualified with the TAS evidence code, is based on the same paper [Hochstrasser, 1996] as these genes' ascospore formation annotation. As with the latter, this annotation is no longer present in the latest version of the GO. However, while this may bring the validity of the specific annotation of the three proteasome subunits into question, there is little doubt that the overall function represented by this group is still appropriate in the context of the proteasome.

The remaining groups shown in Figure 8.3 contain three or fewer genes and will not be discussed here. It is notable that six of the 42 genes in the proteasome dataset, PEP8, RAD52, GSH2, SMT3, PRO3 and BET4, were not included in any of the 13 groups discussed in this section. A few of the discussed groups included some of these genes' annotations in their definition but the genes themselves were never grouped as their functional similarity to all the other genes in the groups in question was never sufficiently high. The only time these genes are grouped are in the non-meaningful groups and only in three of the 30 groups in this category are these genes grouped with another gene rather than on their own. There is nothing in their annotations that suggests that any of these six genes are involved in the proteasome pathway. The grouping results presented here indicate that the FuSiGroups algorithm successfully excluded these unrelated genes while identifying the main functions of the remaining 36 genes.

B.2 Ribosome analysis

The ribosome dataset, discussed in Section 8.2, consists of 49 genes which were grouped into 4 supergroups and 45 original unmerged groups, of which 13 meet minimum size requirements. The grouping results are summarised in Figure 8.6 and Table 8.5.

Supergroup 103 and group 1009 have the same name and content but differ significantly in their definitions. Group 1009 only has two definition terms, translation (GO:0006412) and translational elongation (GO:0006414), the latter of which is not found in supergroup 103's 10-term definition. All definition terms from these two groups in question are only annotated to a few of the groups' genes, except for the term translation, which is annotated to every single one of these 24 genes.

The genes that are annotated with definition terms other than translation are RPS9B, TEF4, TIF1, TIF2, YEF3 and YNL247W in supergroup 103 and TEF4 and YEF3 in group 1009. Only the first of these, RPS9B, is a ribosomal protein. The other five include four of the dataset's five translation initiation and elongation factors, and one aminoacyl-tRNA synthetase. Since each of the six genes is also annotated with the term translation, they would have been grouped into a translation-based group even with a higher ST which might have eliminated the other definition terms.

The most unexpected gene to be included in either of these groups is YNL247W. This gene codes for a protein that attaches tRNAs to their corresponding amino acids. While this process is a prerequisite for translation, it is not necessarily di-

rectly involved with overall ribosome function. The annotation of YNL247W with "translation" is derived from electronic annotation. On the other hand, the annotation with "ribosome", on which its inclusion in group 1056 is based, is qualified with the experimental evidence IDA, supported by Fleischer et al. [2006]. Since group 1056 has exactly the same gene content as the two BP groups currently under discussion, the same arguments are likely to apply to the inclusion of each gene in all groups.

More unexpected than the inclusion of YNL247W in groups 103, 1009 and 1056 is the exclusion from these groups of genes such as RPS19B, RLP24 and TEF2, a ribosomal protein, a non-ribosomal protein listed in the KEGG ribosome pathway and a translation elongation factor. These genes are all annotated with both translation and ribosome and are related to the other genes included in the groups. Also annotated with translation but not with ribosome are GRS1 and SSZ1, coding for another aminoacyl-tRNA synthetase and an HSP70 protein, respectively. Yet despite being biologically related to the genes in the groups and despite some similar annotations, none of these fives genes are included in any of the groups in question.

Genes RPS19B and RLP24 are however included in group 1045, structural constituent of ribosome (GO:0003735), while genes TEF4, TIF1, TIF2, YEF3 and YNL247W are excluded from this group. This means that all genes in the ribosome dataset which are also listed in the KEGG pathway are included in this group. The appropriateness of the inclusion of RLP24 in group 1045 is not entirely clear. On the one hand, it is of course listed as part of the pathway in KEGG. On the other hand, group 1045 is a single-term definition group, i.e. structural constituent of ribosome is the only GO term associated with the group. The annotation of RLP24 with this term is based solely on electronic annotation, while annotations relating to ribosome assembly are supported by published evidence. Although the findings in Chapter 4 showed that the inclusion of electronic annotation makes functional similarity measures perform better than their exclusion, individual electronic annotations may not be entirely appropriate. The electronic annotations in question here are still present in the latest version of the GO.

RLP24 is not present in group 1044, cytosol (GO:0005829), which consists of all of the dataset's 20 ribosomal proteins. Only two of this group's four definition terms are actually found in the annotations of its gene content, namely cytosolic large ribosomal subunit and cytosolic small ribosomal subunit. Its two other definition terms, cytosol and cytosolic ribosome, are annotated to genes not found in the group. The reason for the exclusion of RLP24 from group 1044 is due to its annotation with cytoplasm, a parent term of cytosol, rather than cytosol itself. An annotation with

cytsol might be appropriate here as there is clearly a physical interaction between the protein coded by RLP24 and ribosomal proteins, which would suggest that the protein should also be found in the cytosol.

In light of the existing annotation however, groups 1044 and 1045 can be considered as near-perfect matches. Group 1045 reflects the data as considered by a different data source, in this case KEGG, while group 1044 brings together the most closely related genes in the dataset. The presence of terms in group 1044's definition that are not represented in its genes' annotations can be addressed with a minor modification of the grouping algorithm, as suggested in Section 8.1.2.

The remaining groups for the ribosome dataset are much less consistent than these first five groups. The next group in Table 8.5, BP supergroup 102, is named with the generic term cellular process (GO:0009987). The group contains one extra gene (YNL247W) not found in in the content-wise identical group 104 but the other 15 genes are common to both groups. The original groups from which these supergroups were created are groups 1004, 1006, 1038, 1043, 1049 and 1059 for supergroup 102, and groups 1032, 1043 and 1049 for supergroup 104. The reason that supergroup 104 was not absorbed into supergroup 102 is that although they share two of their original groups, the overlap between the sets is overall insufficient. In addition, there is no overlap at all between the definition of group 1032 and groups 1004, 1006, 1038 and 1059, which is why these groups were never part of the same set of mergeable groups.

Group 102's definition is slightly more focussed on the RNA aspects of ribosome biogenesis, while group 104's definition covers more of the protein aspects. This is reflected in the difference in group content between the two groups. Group 102 contains an aminoacyl-tRNA synthetase (YNL247W) not found in group 104. Overall however, the same conclusion applies to group 102 as to group 104, namely that there are no unexpected genes in the group but a number of genes that would have fitted into the group based on their annotation were excluded due to their level of functional similarity with other genes.

All the genes in BP supergroup 104, called ribosome biogenesis (GO:0042254) are in some way involved in the process of ribosome biogenesis. There are however a number of genes which might be expected to be included in this group as it has definition terms which are not part of the annotations of any of its content genes. In total, there are 14 genes which, based on their annotation, could be part of either or both of these groups. Their absence from the groups is due to the very diverse nature of the proteins involved in ribosome biogenesis, which in turn means a wide range of GO term annotations and hence an insufficiently high level of functional

similarity between some of the genes. The main reason for the wide range of genes that could belong to the group based on their annotation is due to the size of the group's definition. Although all the terms in the definition are related to the process of ribosome biogenesis, their number suggests a slightly higher ST might be appropriate. This would lead to more groups with more specific definitions consisting of fewer terms, a fact that is observable to a certain extent if the maximum ST is used. However, while the specificity of the groups is greater at this level, almost two thirds of the groups covering any of the aspects of ribosome biogenesis found in group 104 contain fewer than 4 genes.

The next group in Table 8.5, group 1025, a CC group named preribosome (GO:0030684), groups together all the genes from the dataset which have a documented association with this early complex created during ribosome biogenesis. The group contains many of the genes also found in the previous two sets of groups, although some of the ribosomal proteins are absent, as they may not be added to the ribosome until the final stages of its assembly.

Group 1013, also a CC group, named nucleolus (GO:0005730), covers another cellular location of the early stages of ribosome biogenesis. The group contains 11 of the 12 genes associated with one of its two definition terms. The only gene not included in this group when it might reasonably be expected to be included is NSR1, a gene coding for a nucleolar protein. NSR1's exclusion from the group is due to its level of functional similarity with only one other gene, RLP24, which at 0.168 is only marginally below the FT.

In addition to the various nucleolar proteins involved in ribosome biogenesis that are included in group 1013, the group also contains two RNA polymerase subunits, RPA49 and RPC19. Their inclusion is due to the presence of the term "DNA-directed RNA polymerase I complex" (GO:0005736) in the group definition. RNA polymerase I primarily produces rRNA [Alberts et al., 2002], so this association of RNA polymerase subunits and other ribosome creation-related proteins is biologically valid. The association of the two GO terms of group 1013's definition only occurs at the minimum ST as their semantic similarity value is 0.38. There is no nucleolus-based group at all at the maximum ST because the semantic similarity of the term "nucleolus" with itself is also 0.38, so any group with this term in its definition would violate the maximum completeness rule with itself.

The two RNA polymerase subunits are absent from group 1016, which is also called nucleolus and is identical in content to group 1013 apart from those two genes. The difference in content is due to the replacement of the definition term "DNA-directed RNA polymerase I complex" with "box C/D snoRNP complex"

(GO:0031428), an evidence-based annotation term of genes NOP56 and NOP58. In the GO release used here, RPA49 and RPC19 are not annotated with the term nucleolus so are excluded from group 1016. If the latest version of the GO were used, they would be included in the group as both genes are now annotated with the term under the IEA evidence code, in which case groups 1013 and 1016 would qualify for merging into a supergroup.

Group 1051 is only the second MF group in Table 8.5. This group, named RNA binding (GO:0003723), consists of ten genes, six of which code for ribosomal components and four for proteins involved in ribosome assembly. There are a further five genes whose annotations would have qualified them for inclusion in the group but who did not meet the minimum FT. Since a large part of a ribosome consists of rRNA, RNA binding would be an expected functional aspect of genes associated with the ribosome. Although the annotation of the ribosomal components genes with RNA binding terms is only electronic and not evidence-based, it has been derived from at least two sources for each gene-GO term pair in the latest version of the GO.

Of the six groups in Table 8.5 which have not yet been discussed, groups 1036 and 1050, and groups 101 and 1052 have the same gene content. The first two of these groups, MF groups 1036 (purine ribonucleotide binding (GO:0032555)) and 1050 (hydrolase activity, acting on acid anhydrides (GO:0016817)), cover two distinct but conceptually related functional aspects. Any ribonucleotide hydrolase activity requires the binding of the appropriate compound by the hydrolase enzyme [Berg et al., 2002]. While conceptually linked, the sets of functions covered in the two groups have an overall semantic similarity of 0 because they belong to different branches (catalytic activity and binding) of the sub-ontology, i.e. their only common ancestor is the MF ontology root.

There are three further genes, GRS1, GUA1 and YNL247W, which are annotated with ATP binding but not included in group 1036. However none of these genes have any hyrdolase activity-related annotation, which may explain why their functional similarity with all or some of the genes in the group is too low to pass the FT.

The other pair of groups with identical content of the final six groups from Table 8.5 are the BP supergroup 101 (amino acid metabolic process (GO:0006520)) and MF group 1052 (ligase activity (GO:0016874)). Supergroup 101 was created by merging original groups 1024 and 1031, of size 4 and 2, respectively. The only element from group 1031 not found in group 1024 is one definition term, which is however not actually annotated to any of the groups' genes. In fact, five out of 1031's six definition terms and four of 1024's nine terms are not annotated to their content.

In the supergroup, this translates to five out of ten terms. The six definition terms of group 1052 are all found in the annotations of the group's genes.

Both groups contain genes which, whilst clearly having enough similarity to be grouped together based on two different types of annotation, code for two different types of synthases. GRS1 and YNL247W code for aminoacyl-tRNA synthases, while GUA1 and URA7 are ribonucleotide synthases which catalyse the biosynthesis of GMP and CTP, respectively. Each of the two group names reflects the overall nature of these functions and processes. In the case of the MF group, any kind of synthesis activity involves the joining of two substances, hence the shared ancestor term of ligase activity. For the BP group, both types of synthesis involve amino acids, glutamine for both ribonucleotide synthases, glycine and cysteine for the two aminoacyl-tRNA synthases, hence the shared ancestor term of amino acid metabolic process.

The two final groups in Table 8.5 that have not yet been discussed are group 1035, a BP group and group 1039, an MF group. Group 1035, named intracellular transport (GO:0046907), has a seven-term definition of which only two terms are actually annotated to the group's content. The two relevant definition terms are rRNA export from nucleus and ribosomal large subunit export from nucleus, which, if they were the only terms in the definitions, would have given a group name of nuclear export. Five of the group's six genes are ribosomal subunit proteins associated with rRNA export from the nucleus [Ferreira-Cerca et al., 2005]. The sixth gene is NMD3, which codes for a protein involved in the nuclear export of the large ribosomal subunit. Although functional similarity between all six genes is of course higher than the minimum FT, the functional similarity between NMD3 and the other genes in lower than between the five genes associated with rRNA export. This group is a rare example where the use of the maximum FT would in fact generate an even more appropriate group as NMD3 would be excluded, leaving only genes involved in rRNA export. Nonetheless, this does not mean that the inclusion of NMD3 in group 1035 is biologically inappropriate as all genes in the group are involved in some way in ribosome formation, as well as in nuclear export.

Finally, group 1039, called translation factor activity, nucleic acid binding (GO:0008135), which is the direct parent of the group's two definition terms, translation initiation factor activity and translation elongation factor activity. The group's content consists of the five translation initiation and elongation factors found in the ribosome dataset. The similarity between the elements of this group, both in content and definition, is so high that raising either the ST or the FT to maximum would not change the group.

The remaining groups for the ribosome dataset all contain fewer than four genes, as can be seen in Figure 8.6. It is notable that of the ribosomal proteins, only two are ever grouped in non-meaningful group, suggesting a high level of consistency in the annotation of these proteins. It is also notable that with the exception of the groups 101 and 1052, none of the six genes that are not obviously related to some aspect of ribosome function are grouped with the other genes in the dataset. While the grouping in the discussed groups made biological sense, this suggests that the FuSiGroups algorithm is sufficiently sensitive to filter out noise from overly generic terms such as cytoplasm, the most common annotation term in the dataset. As in Section 8.1, the algorithm also managed to identify the main functional aspects represented in the bulk of the dataset.

B.3 Pathways analysis

The pathways dataset, discussed in Section 8.3, consists of 34 genes which were grouped into 6 supergroups and 47 original unmerged groups, of which 18 contain four or more genes. Two of the groups contain genes from both superpathways. The grouping results are summarised in Figure 8.9 and Table 8.9.

The three largest supergroups, 104, 105 and 102 have the same name and very similar content, although they are not identical. All three supergroups share some of the original groups they were derived from. All three supergroups contain the original group 1021, while groups 104 and 105 share group 1019, and 102 and 104 share group 1034. In all cases, insufficent overlap in content and definition of the distinct original groups is the the reason for their presence in two or more supergroups.

Group 104 contains two of the three genes unique to the glyoxylate cycle, MLS1 and ICL1, but not the third one, DAL7. Group 105 on the other hand does not contain ICL1 but does contain DAL7. Both groups contain only one of the two isoforms of the pyruvate carboxylase unique to the TCA cycle, PYC2, but not the other isoform, PYC1. This is because the functional similarity of PYC1 with many of the other pathway's unique genes is below the functional threshold, so it was not included in any of the original groups that gave rise to the supergroups. In fact, PYC1 and PYC2's functional similarity with many of the TCA cycle genes is lower than the functional threshold, which is reflected in the exclusion of these two genes from all the non-supergroups which contain most of the other TCA cycle genes. This is due to the enzymes' role in the creation of oxaloacetate, which feeds into the TCA cycle but is not part of the actual cycle [Berg et al., 2002]. The only group for which this absence is not the case is supergroup 102, which contains both PYC1

and PYC2, as well as most of the other glyoxylate and TCA cycle genes. Aside from the addition of PYC1, group 102's content is very similar to that of group 104, with the exception of genes SDH1, SDH2, SDH3, SDH4, LCS1 and LCS2.

Overall, group 104 covers most of the superpathway genes. Its definition is however quite diverse and covers a range of biological processes which are not all directly related to the TCA cycle. While this is partly the result of the merging of three group definitions into the supergroup definition, the original definitions were already equally diverse, suggesting once again that the semantic threshold is insufficiently high. The exact same problem exists for groups 102 and 105, which have almost equally diverse group definitions. However unlike group 104, which has one definition term not found at all in the annotations of its content genes, all definition terms in groups 102 and 105 are annotated to at least one of the groups' genes. The same type of diverse group definitions are not found at maximum ST, where the largest groups are similar to groups 1025 and 1029 in Table 8.9.

The diversity of these definitions is due to the fact that a large number of the GO terms in the pathways dataset are only annotated to one or two genes, reflecting the various other pathways the different steps of the TCA cycle feed into. While the resulting groups are not biologically incorrect, their definitions do not provide any meaningful insight into the common functional aspects of the genes in the groups beyond the fact that some or all of the genes are also involved in other metabolic processes.

The problem of overly diverse definitions does not exist in groups 1025 and 1029, which despite having largely the same gene content, share only one of their seven and five respective definition terms. Both groups contain all TCA cycle and glyoxylcate cycle genes except for ICL1, LPD1, PYC1 and PYC2. In the case of group 1025, the genes are grouped under the banner of generation of precursor metabolites and energy (GO:0006091), while group 1029 covers the area of coenzyme metabolic process (GO:0006732).

Both groups cover processes which genes from the two pathways are indeed involved in. They are both essential in the cell's energy production, as well as generating intermediates for various biosynthetic processes. Both pathways also involve several coenzymes, such as acetyl CoA and NADH. The inclusion of all the genes in the group is therefore biologically appropriate in this context and the exclusion of four is genes due only to the insufficient level of overall functional similarity between these genes and some of the groups' genes.

It should however be noted that of all the terms in the two group definitions, the only one that accounts for the inclusion of the two glyoxylate cycle-specific genes

MLS1 and DAL7 is tricarboxylic acid cycle (GO:0006099). For both genes, this annotation is qualified with the "IEA" evidence code. As neither of these genes is actually involved in the TCA cycle, the validity of this annotation is questionable. Therefore, while the inclusion of the two genes in group 1025 and 1029 is appropriate, the reason for their inclusion may not be.

The next four groups in Table 8.9, supergroups 101 and 103 and groups 1001 and 1058, do not suffer from this problem. Supergroups 101, called mitochondrial part (GO:0044429), contains in its definition the term mitochondrial outer membrane (GO:0005741), which is not annotated to any of the genes in the group. In fact, this term is annotated to CHO1, a gene from the phospholipid biosynthesis pathway. There are also two further genes from this pathway, CRD1 and PSD1, that are annotated with mitochondria-related GO terms, although their overall functional similarity with the genes from the other superpathway is insufficient for inclusion in the group. The group does not contain all TCA cycle genes, due to the fact that several of these genes are only annotated with the more general term mitochondrion, which is too general to be included in the group definitions. The resulting exclusion from the group of certain genes means that the groups have a higher specificity in terms of the cellular locations they represent but a lower accuracy in terms of their content.

As mentioned at the beginning of this section, two groups contain genes from both superpathways in the dataset. One of these groups is CC supergroup 103, intracellular membrane-bounded organelle (GO:0043231). This group contains 11 TCA cycle genes and two phospholipid biosynthesis genes. Its name is more high-level than most of its actual definition terms of which all but one, nuclear envelope, are mitochondrion-related. However nuclear envelope is not annotated to any of the group's genes, so the group definition effectively relates only to mitochondrial sub parts. The presence of cross-superpathway CC groups relating to mitochondria had already been predicted in Section 8.3.1.

There are two reason why only 13 of 22 potential genes are included in the group. First of all, the term mitochondrion, the one term annotated to all 22 genes, is absent from the group definition because it is not sufficiently semantically similar to the rest of the definition terms. This makes five genes ineligible for inclusion in the group. The same problem applies to another GO term, mitochondrial matrix, affecting further three genes. Secondly, several of these genes do not have sufficient functional similarity to the other genes to be included in the group. For these reasons, it is not possible to obtain a comprehensive group with all or most of the mitochondrion-based genes, despite mitochondrion-related cellular locations being a

key common aspect of many genes in the dataset.

BP group 1001 and MF group 1058, which have the same gene content except for one gene not present in group 1058, contain only genes from the TCA cycle. Even though the two groups belong to different ontologies, their names refer to two aspects of the same concept, namely the process of oxidation reduction (GO:0055114) and the corresponding function of oxidoreductase activity(GO:0016491).

The only difference in content between the two groups is the gene ACO1, which is present in the BP group due to its annotation with the term oxidation reduction but absent from the MF group due to the lack of any appropriate annotation. In fact, the BP annotation of ACO1 is qualified with the RCA evidence code, based on Huttonhower and Troyanskaya [2008], and can no longer be found in the latest version of the GO. This is consistent with the chemical process of aconitase activity, which is not a redox reaction [Beinert et al., 1996]. Group 1058 is therefore more representative of the functionality of its content than group 1001.

The next two groups in Figure 8.9, BP supergroup 106 and BP group 1005, contain the same ten genes, namely all the genes from the phosphatidic acid and phospholipid biosynthesis superpathway. This is reflected in their respective names, lipid metabolic process (GO:0006629) and cellular lipid metabolic process (GO:0044255), the former of which is a direct parent term of the latter and represents the most generic way of describing the superpathway. All genes in the group are annotated with the more specific term phospholipid biosynthetic process (GO:0008654) but the presence of other process terms in the annotations leads to a more high-level group name. The reason for the presence of the two groups despite their identical content is insufficient overlap in definition between group 1005 and the constituent groups of supergroup 106.

It should be noted that one term in group 106's definition, fatty acid beta-oxidation, is not annotated to any of the group's genes. This term is annotated to one of the TCA cycle genes, MDH3. In terms of overall functional similarity however, this gene is far too different from the phospholipid biosynthesis genes to be included in a group with them.

BP group 1043, carbohydrate metabolic process (GO:0005975), contains ten genes from the TCA and glyoxylate cycles, including two glyoxylate cycle-specific genes, two TCA cycle-specifc genes and six shared genes. All five terms from the group definition are found in the genes' annotations, while two genes, DAL7 and PYC2, that are annotated with one or more definition terms are excluded from the group due to insufficient overall functional similarity to other group members. The group's definition includes the terms glyoxylate cycle, glycolysis and gluconeogene-

sis but not the term tricarboxylic acid cycle. It would in fact be impossible for the terms glyoxylate cycle and tricarboxylic acid cycle to be found in the same group definition as they have a semantic similarity of 0.08, on account of the fact that they are located in different branches of the BP ontological tree, despite being very similar processes.

The presence of the terms glycolysis and gluconeogenesis in group 1043's definition is the reason for the inclusion of non-glyoxylate cycle genes KGD1 and PYC1 in the group. Although from a biological perspective, the TCA cycle should be more similar to any of the three other terms than glyoxylate cycle to glycolysis or gluconeogenesis, the location of all the terms in the GO tree makes this impossible in semantic terms. The similarity between glyoxylate cycle and the other two terms is in fact so high that they are also found together in group definitions at maximum ST. While neither the grouping of the definition GO terms nor of the genes is biologically incorrect, the combination of this definition and content into group 1043 is not entirely accurate as it could incorrectly imply a functional relationship between the glyoxylate cycle-specific genes and the pathways of glycolysis and gluconeogenesis.

The next group in Table 8.9, MF group 1056, covers the functional aspect of transferase activity, transferring acyl groups (GO:0016746), a high-level name for a number of crucial reactions in both the TCA and glyoxylate cycle, namely the transfer of an acyl group, usually from acetyl-CoA, to another molecule. The individual group definition terms reflect the number of reactions which include an acyl group transfer. One definition term, 1-acylglycerol-3-phosphate O-acyltransferase activity, is not annotated to any of the group genes. It is annotated to the phosphatidic acid biosynthesis gene SLC1. This gene was excluded due to insufficient functional similarity with some, though not all, of the group's genes. All the other genes in the dataset which have any kind of acyl group transfer function are however included in the group.

MF groups 1011 and 1051 both cover forms of binding activity, transition metal ion binding (GO:0046914) and cofactor binding (GO:0048037), respectively. The former includes iron, zinc and manganese ion binding, as well as heme binding, as child term of iron ion binding. Not included, due to insufficient semantic similarity, are the terms magnesium ion binding and metal ion binding. Their absence does not affect the group content as the genes annotated with these terms are also annotated with terms present in the definition. Due to insufficient functional similarity, the genes annotated with zinc ion binding, PYC1 and PYC2 are not in the group. The second group's definition includes thiamin pyrophosphate, FAD, lipoic acid and quinone binding. All genes annotated with one of these terms are included in the

group.

BP group 1020, named cellular aldehyde metabolic process (GO:0006081), contains only genes from the glyoxylate cycle, specifically the three glyoxylate cycle-specific genes and two of the seven shared genes. The group's definition consists of only two terms, the group name and glyoxylate cycle, the former of which is a parent term of the latter and found only in the annotations of DAL7. Interesting is the absence of annotation with glyoxylate cycle of the five genes MDH1, MDH2, ACO1, CIT1 and CIT3, even in the form of electronic annotation, and especially considering that the three glyoxylate cycle-specific genes are all annotated with the term tricarboxylic acid cycle. While a questionable "guilt by association" annotation exists for these three genes, the reverse has not occurred for the genes absent from the group. It could be argued that MDH1 should not have a glyoxylate cycle annotation because its full descriptive name, mitochondrial malate dehydrogenase, implies that the protein it codes for is limited to the TCA cycle. A similar justification also exists for CIT1 and CIT3, two isoforms of citrate synthase active in the mitochondrial reaction, while CIT2 catalyses the same reaction in the glyoxylate cycle [Graybill et al., 2007]. This of course brings into question the appropriateness of including MDH1, CIT1 and CIT3 in the glyoxylate cycle at all as these three genes are clearly only active in the mitochondrial TCA cycle. No justification can be provided for the absence of ACO1 and MDH2 from group 1020.

A similar problem does not exist for CC group 1045, named microbody (GO:0042579) on the basis of its definition terms peroxisome,peroxisomal matrix and glyoxysome. This group contains four of the ten glyoxylate cycle genes. Unlike the previous group, the absence of each of the other six genes from the group and their lack of annotation with any of the group definition terms can be explained. Several of the reactions of the glyoxylate cycle, such as the transformation of citrate into isocitrate via cis-aconitate, followed by isocitrate to glyoxylate, as well as in part the transformation of malate into oxaloacetate, actually take place in the cytosol rather than the peroxisome [Feldmann, 2005]. The genes involved in these reactions are ACO1, ICL1 and MDH2, which is in fact described as cytoplasmic malate dehydrogenase, unlike its isoenzyme MDH3, described as peroxisomal malate dehydrogenase and included in the group. The justification for the absence of MDH1, CIT1 and CIT3 is the same as for the previous group.

The group following group 1020 in Table 8.9 is CC group 1035, cytosol (GO:0005829). This single-term definition group contains five TCA cycle genes, ACO1, FUM1, MDH2, PYC1 and PYC2. These five genes are the only ones in the dataset annotated with the term cytosol. For ACO1 and FUM1, this annotation reflects the dual

localisation of the two enzymes in both cytosol and mitochondria [Regev-Rudzki et al., 2009]. It has already been established in previous groups that MDH2 is a cytoplasmic isoform of malate dehydrogenase. Finally, unlike most eukaryotic organisms, in which the synthesis of oxaloacetate from pyruvate occurs in the mitochondria, yeast's pyruvate carboxylases PYC1 and PYC2 are active only in the cytosol [Pronk et al., 1996].

BP group 1053, called phosphorus metabolic process (GO:0006793), is based on two definition terms, its name term and mitochondrial electron transport, succinate to ubiquinone (GO:0006121). The former is annotated to one of the group's five genes, ACO1, while the latter is part of the annotation of all four succinate dehydrogenase subunits. The annotation of these four subunits is consistent with their function [Oyedotun and Lemire, 2004], whereas there is no evidence that aconitase activity involves the manipulation of a phosphorus atom or compound containing phosphorus. The association between ACO1 and phosphorus metabolic process is based on the RCA evidence code, citing the paper by Huttonhower and Troyanskaya [2008]. It is no longer present in the latest version of the GO, suggesting that ACO1 should not be present in this group and that the group should in fact contain the four succinate dehydrogenase subunits, under the single-term definition of mitochondrial electron transport, succinate to ubiquinone.

Group 1060 is a MF group containing only genes from the phospholipid biosynthesis superpathway. Under the name of transferase activity, transferring phosphorus-containing groups (GO:0016772) are covered a range of very specific enzymatic activities. Each enzyme in the group catalyses a reaction which involves a phosphorus-containing residue. All the genes from the same superpathway not included in the group on the other hand involve acyl or methyl group transfers or decarboxylase activity and their inclusion in this group would therefore not be appropriate.

The remaining four undiscussed groups in Table 8.9, three MF groups and one CC group, all contain four genes. The first two of these groups, group 1007, purine nucleotide binding (GO:0017076) and group 1026, ligase activity (GO:0016874), describe two different functional aspects of the same four genes. Ligase activity, the joining of two substances, requires energy from ATP or another triphosphate so it would be expected that a protein with some kind of ligase activity would also have an active site for triphosphate binding. The reverse is however not true since many different reactions require energy in the form of ATP. In fact, there are three genes in the dataset annotated with one of group 1007's definition terms that do not have ligase activity. These are two dehydrogenases, LPD1 and SDH1, and a synthase from the phospholipid biosynthesis pathway, PGS1. The two dehydrogenases are

annotated with the term FAD binding, which is not annotated to any other term in the group, while PGS1 has ATP binding functionality. Group 1026 on the other hand contains all the genes that are annotated with any of its definition terms. The four genes that the two groups contain are the two subunits of succinyl-CoA ligase, LSC1 and LSC2, and the two isoforms of pyruvate carboxylase, PCY1 and PCY2.

This leaves MF group 1033 and CC group 1039 to be discussed. The former of these two, called lyase activity (GO:0016829), is one of the two groups containing genes from both superpathways. In addition to the three TCA or glyoxylate cycle genes ACO1, FUM1 and ICL1, the group also contains the phosphatidylserine decarboxylase PSD1. Based on the definition terms, it could also contain the non-mitochondrial form of this decarboxylase, PSD2, but the functional similarity between this gene and ACO1 is below the minimum FT, leading to the gene's exclusion from the group. This is a good example of the drawback of the exclusion process of genes from groups based on the FT: both ACO1 and PSD2 have appropriate levels of functional similarity with all the other members of the group except each other. In a situation like this the grouping algorithm simply excludes the first element in the list of elements that violate the maximum completeness rule from the group. In the greater context of the function reflected by the group, neither gene would have been more or less appropriate in the group than the other. The definition of a secondary threshold, lower than the primary ST or FT, to deal with cases such as this one was considered but such a threshold would have been even harder to determine than the primary threshold and would have made the algorithm unnecessarily complex.

It is not surprising to find genes from both superpathways in an MF group such as this one. Lyase activity is an umbrella term for a set of different chemical reactions: any cleavage of a carbon-carbon, carbon-oxygen or carbon-nitrogen bond by a process other than hydrolysis or oxidation is classed as lyase activity. Apart from the four genes in the group and PSD2, none of the other genes in the dataset are lyases, which is reflected in their annotation.

Finally, CC group 1039, named endoplasmic reticulum (GO:0005783), contains four genes from the phospholipid biosynthesis superpathway. The four genes in the group all code for proteins known to be active in the endoplasmic reticulum [Daum et al., 1998; Kuchler et al., 1986], one of the main cellular locations, aside from the mitochondria, in which phospholipid biosynthesis takes place.

All groups in Figure 8.9 not discussed so far contain fewer than four genes and will therefore not be considered here. It should however be noted that due to the relatively small number of genes from the phosphatidic acid and phospholipid biosynthe-

sis pathway, several of these groups containing some of these genes do in fact reflect functional aspects seen only in two or three of the genes, such as methyltransferase activity for genes CHO2 and OPI3.

Overall, the FuSiGroups algorithm was successful in separating the genes from the two superpathways, except for cases where the genes shared a common functional aspect. Previously identified issues with the algorithm such as repetitive supergroups and definition terms not found in the annotations of a group's content were also found in this dataset.