Developing an integrated system for biological network exploration

Jennifer Chang
Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/etd
Part of the Bioinformatics Commons

Recommended Citation
Chang, Jennifer, "Developing an integrated system for biological network exploration" (2017). Graduate Theses and Dissertations. 15498.
https://lib.dr.iastate.edu/etd/15498

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
Developing an integrated system for biological network exploration

by

Jennifer Chang

A dissertation submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Bioinformatics and Computational Biology

Program of Study Committee:
Patrick Schnable, Co-major Professor
Basil Nikolau, Co-major Professor
Hui-Hsien Chou
Jarad Niemi
Stephen Gilbert

The student author and the program of study committee are solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University
Ames, Iowa
2017

Copyright © Jennifer Chang, 2017. All rights reserved.
DEDICATION

This dissertation is dedicated to my father, for his commitment and encouragement throughout my life. More importantly, he set an example for me by solving problems with the creative use of limited resources.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF FIGURES</td>
<td>v</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>vi</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>viii</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>ix</td>
</tr>
<tr>
<td>CHAPTER 1 : GENERAL INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>Dissertation Organization</td>
<td>4</td>
</tr>
<tr>
<td>Literature Review</td>
<td>6</td>
</tr>
<tr>
<td>Biological network based discovery</td>
<td>6</td>
</tr>
<tr>
<td>Biological network repositories and standards</td>
<td>6</td>
</tr>
<tr>
<td>Existing biological network analysis pipelines</td>
<td>7</td>
</tr>
<tr>
<td>CHAPTER 2 : MANGO: COMBINING AND ANALYZING HETEROGENEOUS BIOLOGICAL NETWORKS</td>
<td>9</td>
</tr>
<tr>
<td>Abstract</td>
<td>9</td>
</tr>
<tr>
<td>Background</td>
<td>9</td>
</tr>
<tr>
<td>Results</td>
<td>9</td>
</tr>
<tr>
<td>Conclusions</td>
<td>9</td>
</tr>
<tr>
<td>Background</td>
<td>10</td>
</tr>
<tr>
<td>Implementation</td>
<td>12</td>
</tr>
<tr>
<td>The Mango user interface</td>
<td>12</td>
</tr>
<tr>
<td>The Graph Exploration Language (Gel)</td>
<td>15</td>
</tr>
<tr>
<td>Standards for combining heterogeneous graphs</td>
<td>17</td>
</tr>
<tr>
<td>KEGG Connect</td>
<td>21</td>
</tr>
<tr>
<td>Results and Discussion</td>
<td>22</td>
</tr>
<tr>
<td>Network data collection</td>
<td>22</td>
</tr>
<tr>
<td>Large heterogeneous network comparison</td>
<td>23</td>
</tr>
<tr>
<td>Flexible real-time network exploration and visualization</td>
<td>25</td>
</tr>
<tr>
<td>Microarray expression combined with KEGG biological pathways</td>
<td>27</td>
</tr>
<tr>
<td>Conclusion</td>
<td>29</td>
</tr>
<tr>
<td>Availability and Requirements</td>
<td>29</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>31</td>
</tr>
<tr>
<td>CHAPTER 3 : CAVATICA: A PIPELINE FOR IDENTIFYING AUTHOR ADOPTION TRENDS AMONG SOFTWARE TOOLS OR METHODS FROM LITERATURE</td>
<td>32</td>
</tr>
<tr>
<td>Abstract</td>
<td>32</td>
</tr>
<tr>
<td>Introduction</td>
<td>32</td>
</tr>
</tbody>
</table>
CHAPTER 4 : EVALUATION AND PREDICTION OF PROBIOTICS BASED ON HUMAN MICROBIOME PROJECT DATASETS ........................................58
Abstract ........................................................................................................58
Introduction ....................................................................................................59
HMP Datasets ................................................................................................61
Mapping the 18 probiotic strains to significantly different OTUs.............62
  Merging heterogeneous graphs in Mango Graph Studio .........................67
Mapping the 18 probiotic strains to significantly different KEGG pathways ...75
Identify potential strain specific functions in a community.........................77
Discussion ....................................................................................................79

CHAPTER 5 : DISCUSSIONS AND CONCLUSIONS ......................................80
Recommendations for Future Research .......................................................81

REFERENCES .............................................................................................83
LIST OF FIGURES

Figure 2.1 System architecture. The Mango software is made up of multiple code layers seamlessly stacked up to form the stand-alone program. The GPU speedup layer is not included in some Mango versions. .......................................................13

Figure 2.2 Mango user interface. The main window is divided into four areas: data list (left), graph canvases (middle 3D visualizations), Gel editor (bottom left), and Gel command console (bottom right). .................................................................14

Figure 2.3 Graph Exploration Language examples. (a) Graphs A and B have different node attributes. Graph C is the result of attribute merging and promotion of A and B. (b) Graph mathematics. Given two graphs A and B, the dotted addition A .+ B combines nodes and links from graph A and graph B. ..............17

Figure 2.4 KEGG Connect. (Left) The KEGG Connect dialog lists currently available organisms and pathways in the KEGG database. Users can fetch multiple pathways individually or merge them into one network by checking the "Merge Fetched Pathways" box. (Middle) .................................................................22

Figure 2.5 Biological network comparisons. Link intersections among the corr, path, go and ppi networks. The intersections were worked out using Gel commands. WGCNA is the gene-to-gene correlation network corr computed from E. coli microarray data. .................................................................24

Figure 2.6 Gene expression combine with KEGG. A 3D KEGG network visualization comparing the E. coli gene expression values obtained under a treatment condition and a control condition. .................................................................28

Figure 3.1 Cavatica Pipeline to fetch PubMed and PubMed Central (PMC) results. The dotted arrows indicate optional computer-aided human curation steps of search results using automatically generated html files that highlight sentences containing the search keywords. .................................................................41

Figure 3.2 The sentence-highlighting HTML file lists each article live link ID and title in a text block. Under the title is a bullet list of sentences containing the search terms in bold typeface. In this example, we have identified a false positive hit for the Gephi software that is boxed in red. .................................................................42

Figure 3.3 Nine co-author publication networks for each network software tool. The previous code box was used to generate the VisANT network highlighted in blue. Similar code was used to generate the other 8 networks. .........................45
Figure 3.4 Side view of merged 3D PubMed co-author publication networks. Each vertical slice is a different co-author network resulting from a different software name search term. Blue links between different subnetworks indicate authors who have used multiple tools.

Figure 3.5 Author adoption trends among software tools. The black paper nodes are labeled with year of publication and single-letter code(s) representing the software tool(s) it is associated with.

Figure 3.6 Counts of PubMed publications by year from 1996 to 2016 for all network analysis software tools except GraphLab, whose bar chart looked much like iGraph, with only two publications, one in 2013 and the other in 2014.

Figure 3.7 Counts of PubMed Central publications by year from 1996 to 2016 for all network analysis software tools except GraphLab, whose bar chart only showed three publications, one in 2015 and two in 2016.

Figure 4.1 HMP data analysis pipeline. HMP provided datasets are boxed in blue. The various heterogeneous networks are combined in Mango Graph Studio to answer biological questions.

Figure 4.2 The three biological networks generated from the HMP datasets. In graph g, the body site is listed on the left and are linked to OTUs on the right that were significantly more abundant in GI.

Figure 4.3 Merging the three biological networks. Note that the 9349 significant OTUs remain at their x coordinate in graph g, thus the number of OTUs staying at the x coordinate in g2 is reduced to 36034.

Figure 4.4 Regrouping of layers in the graph. The body sites are grouped into Mouth, Skin, and Vaginal. The taxa are rearranged to follow the taxonomy structure.

Figure 4.5 Subnetworks by location

Figure 4.6 Layered network of several different biological entities grouped by column.

Figure 4.7 Network analysis of layered biological entities

Figure 4.8 Probiotic to enzyme network. The enzymes are labeled by number of probiotic strains connected to it.
LIST OF TABLES

Table 2.1 Comparison of graph visualization software ..........................................................11
Table 2.2 Summary of 4 large heterogeneous biological networks for E. coli ..................23
Table 2.3 Benchmarking the speed of Gel mathematics on massive graphs ..................25
Table 3.1 Total paper count and author adoption trend for microarray and RNA-Seq ....36
Table 3.2 Total paper count and author adoption trend for CRISPR, TALEN, and Zinc Finger genome editing methods .................................................................37
Table 3.3 PubMed author adoption trends among network software tools .................38
Table 3.4 PMC author adoption trends among network software tools ......................38
Table 3.5 PubMed and PMC author adoption trends among probiotics .................40
Table 3.6 Number of PubMed and PubMed Central (PMC) search results for each software .................................................................................................................43
Table 3.7 Growth factors for up to the 3 largest PubMed communities of each software .47
Table 3.8 Sample recent PubMed papers from up to the three largest communities of each software if available .................................................................54
Table 3.9 Sample recent PubMed Central papers from the three largest communities of each software if available .................................................................55
Table 4.1 The probiotic genes and their corresponding microbe names in the HMP microbe counts .............................................................................................................63
Table 4.2 The 18 probiotics and their corresponding organism name in the HMP reference .............................................................................................................64
Table 4.3 Summary of the v35 Microbe samples by body site ..................................64
Table 4.4 Counts of various entities that were significantly greater in stool (GI) than other body site .............................................................................................................65
Table 4.5 Summary of the HMP KEGG pathway data by body location .................66
Table 4.6 Probiotic strains mapped to KEGG organisms and the organisms identified enzymes .............................................................................................................75
ACKNOWLEDGMENTS

I would like to thank my committee members, Drs. Patrick Schnable, Basil Nikolau, Hui-Hsien Chou, Jarad Niemi, and Stephen Gilbert, for their guidance and support throughout the course of this research. I would like to especially thank Dr. Hui-Hsien Chou for his help and advice when writing software, reorganizing rough drafts, and starting a company. I also wish to thank Drs. Di Cook, Heike Hofmann, and Jo Anne Powell-Coffman for their feedback and suggestions. I thank my lab mates and friends for constructive discussions and candid conversations.

In addition, I would also like to thank the department faculty and staff for making my time at Iowa State University a pleasant experience through their support and hard work. I thank my team at Complex Computation, LLC for their support and optimism.
ABSTRACT

Network analysis and visualization have been used in systems biology to extract biological insight from complex datasets. Many existing network analysis tools either focus on visualization but have limited scalability, or focus on analysis but have limited visualizations. The separation of analyzing the raw data from visualizing the analysis results causes systems biologists to jump between forming a question, building a massive network, identifying a subnetwork for visualization, and using the visualization as feedback and inspiration for the next question. This iterative process can take several days, making it difficult for researchers to maintain the mental map of the questions queried. In addition, biological data is stored in different formats and has differing annotations, thus systems biologists often run into hurdles when merging large or heterogeneous networks. The polymorphic nature of the datasets presents a challenge for researchers to integrate data to answer biological questions. A more systematic method for merging networks, resolving data conflicts, and analyzing networks may improve the efficiency and scalability of heterogeneous multi-network analysis.

Towards improving and pushing forward multi-network analysis to help a researcher easily combine multiple heterogeneous biological data networks to answer biological questions, this dissertation reports several accomplishments that provide (i) a set of standard multi-network operations, (ii) standard merging rules for heterogeneous networks, (iii) standard methods to reproduce network analyses, (iv) a single integrated software environment that allows users to visualize and explore the network analysis results and (v) several examples applying these methods in biological analysis. These efforts have culminated in three academic publications.
Biological pathways govern the biochemical processes in living cells. Every aspect of a cell is part of a huge and complex network of enzymes, substrates, and metabolites that interact with one another. In addition, it is estimated that the human body contains 10X more microbial cells than human cells. The microbial and host symbiosis can affect the organism's health and ability to break down important metabolites. Living organisms are made of robust and self-regulating systems of genes, proteins, and microbes. In addition, from genotype to phenotype there are complex associations yet to be discovered. Modern genomic techniques offer high-throughput methods for collecting data from various aspects of a living species. Next-generation sequencing, microarrays and automatic proteomics are some examples of techniques that have allowed publications, annotations, interpretations and a myriad of distilled data in different formats to be created.

Scientists are generating data faster than they can analyze, and the data comes in various formats and quality levels that are difficult to integrate—modern biological research requires new computational solutions for data analysis and inference. Even to determine the difference between healthy and unhealthy states often requires a survey of thousands of genes, compounds or samples. In every such survey, separating signal from noise is a challenging computational biology research endeavor.

A graph (a.k.a. network) can give a visual summary of the data, provide methods to determine emergent properties, and help identify subnetworks to reduce the search space for relevant genes or compounds. Graph analysis tools have helped researchers gain new insights into biological data either by calculating centrality measures or providing visual cues to
heterogeneous biological data. However, merging data into massive graphs (> 10K links) and visualizing them continues to be a limitation.

The problem is further compounded by the common separation of graph analysis steps from graph visualizations; many existing biological graph analysis pipelines require separate command-line scripts to analyze, combine and reduce graph data files, and subsequently need other graph visualization tools to get visual feedback of the analysis results. While iterating between each cycle of graph analysis and visual feedback in those pipelines, there are significant lags that limit the efficiency of multi-graph data exploration and inference.

Better solutions must be developed for analyzing biological networks, to include the visualization of multiple heterogeneous biological data networks with their integration and modeling. Efficient multi-graph operations and data conflict managements are essential. An immediate, real-time feedback of graph operations preferably should be provided even when analyzing massive graphs. To streamline the process of combining heterogeneous data and reproducibly analyzing multiple massive networks, we have developed an integrated graph exploration environment called Mango Graph Studio (Mango stands for Manipulation and Analysis of Networks and Gene Ontology) that provides flexible and efficient commands for multi-graph operations and the immediate feedback of real-time graph visualizations. In the following chapters, the details of the Mango Graph Studio system as well as several of its applications in biological studies will be presented.

In Chapter 2, we introduced details of the Mango Graph Studio software which combines the Graph Exploration Language (Gel) and systematic graph mathematics with a 3D interactive visualization environment. Three case studies were then presented that 1)
demonstrate the heterogeneous merging of 4 different kinds of biological networks, 2) benchmark several mathematical operations involving a 7K link KEGG biological pathway network and a 4M link gene correlation network, and 3) exemplify a biological application to overlay gene expression data directly onto a KEGG biological pathway network to highlight the up and down-regulated subnetworks.

In Chapter 3, we described a novel pipeline called Cavatica which uses Gel mathematics in Mango Graph Studio to merge author-paper networks built from different biomedical literature search terms given to PubMed and PubMed Central databases. Cavatica generates Mango Graph Studio scripts which are used to construct author-paper networks, merge these networks, and identify authors who are present in multiple networks. These scripts help recognize author adoption trends among several methods or concepts defined by the search terms. The Cavatica pipeline was validated by confirming two known trends among biotechnologies for transcriptome measurement (RNASeq vs Microarray) and genome editing (CRISPR vs TALEN vs Cas9); it was subsequently used to identify novel trends among 9 network analysis software tools and among 18 probiotic strains.

In Chapter 4, we analyzed the Human Microbiome Project (HMP) dataset using Mango Graph Studio to determine if the 18 probiotic strains in consumer health supplements were more prevalent in Human GI tract versus other body sites. Since HMP provides a plethora of datasets all related to human microbiomes, this analysis requires the flexibility of merging different types of networks at several levels (e.g., enzyme, KEGG pathway and 16S rRNA) conferred by Mango Graph Studio. In this work, we presented the versatility of Mango Graph Studio in merging multiple biological networks which allows us to push forward multi-network analysis.
Dissertation Organization

The dissertation is divided into the following chapters:

Chapter 1 provides a summary of the goals and specific aims, a literature review, and a description of the dissertation organization.

Chapter 2 is a published manuscript entitled “Mango: combining and analyzing heterogeneous biological networks” (1). This publication describes graph mathematics that include graph addition, subtraction, multiplication and intersections that are either node or link centric. The graph mathematics as well as the merging and data conflict resolution rules are implemented in the Graph Exploration Language (Gel) within the Mango Graph Studio software that is the focus of this publication. Mango can fetch and merge biological pathways from online databases. In Mango allows heterogeneous biological data such as sequence matches, gene expression correlations, protein-protein interactions, and biochemical pathways to be easily merged and analyzed. The Mango software is written in C++ and runs on Mac OS, Windows, and Linux. Mango distributions are freely available for download from https://www.complexcomputation.com. The Mango User Guide listing all features can be found at http://www.gitbook.com/book/j23414/mango-user-guide

Chapter 3 describes Cavatica, a pipeline to identify author adoption trends among software tools or methods based on automatic PubMed and PubMed Central literature searches and subsequent multi-network analysis. Given a set of search terms, Cavatica generates several large networks and makes use of Gel graph mathematics to merge and analyze them. By constructing and analyzing heterogeneous networks representing both papers and authors, Cavatica can identify collaborating communities and authors who are more devoted to certain tools or methodologies. We have validated the Cavatica pipeline
with known adoption trends in gene expression measurement methods and genome editing techniques; a straight count of papers was usually insufficient to identify the trends. We then applied Cavatica on two new studies: the comparison of 9 different network analysis software tools to identify their author adoption trends, and the comparison of 18 probiotic bacteria commonly found in consumer health supplements to identify their popularity as shown in the scientific literature, in contrast to the following biological study based on their actual prevalence in human digestion tract.

Chapter 4 describes Mango Graph Studio applications in microbiome data analysis. There has been increasing interest in the identification and characterization of microorganisms on different parts of the body to better understand their association with human health or disease. The available datasets from the HMP website include reference sequence, microbe abundance counts, and metagenomics shotgun sequences which are difficult to merge. This chapter showcases the Mango Graph Studio system for combining several layers of data, where each layer contains nodes of grouped types (e.g. enzymes, pathways, organisms) and propagating values across layers to identify potential biomedical research directions. Probiotics is a term commonly attributed to strains believed to be beneficial to gut health. We used Mango Graph Studio to examine the commonly marketed 18 probiotic strains in the HMP sequence dataset to 1) provide confirmation if these ‘probiotic’ strains are overly represented in the guts compared to other body sites of healthy individuals, 2) suggest novel potential probiotic strains, and 3) suggest their microbiome community function through a KEGG pathway analysis.

Chapter 5 summarizes the contribution of this dissertation work and suggests directions for future work.
Literature Review

**Biological network based discovery**

There are many different types of biological networks from protein-protein interaction (PPI), biological pathway, and gene regulatory network (GRN) to name a few. Methods to analyzing biological networks have arrived via graph theory in various forms of centrality measures. From yeast PPI studies, mutations in more central or highly connected proteins versus mutations in less central or sparsely connected proteins are more likely to cause death or failure to thrive (2). A more complete review of biological network analysis inspired by graph theory is summarized in a review paper by Pavlopoulos (3).

Often different types of biological network data must be merged to gain biological insight. Gene expression data analysis is regularly guided by gene ontology and biological pathway information. Examples include the studies of different patterns of brain gene expressions in humans and chimpanzee (4). Differential gene expression studies combined with gene ontology give an indication of which biological functions are emphasized across different species or tissues (5); while gene expression correlation networks combined with biological pathways give an indication on biological functions regulated under biochemical processes. Similar species or tissues within an organism may have minute differences at the DNA level but vastly different phenotypes due to gene expression variations or microbial community content (6,7).

**Biological network repositories and standards**

There are many freely accessible sources for biological network data. Some organism specific websites (e.g., [www.wormbase.org](http://www.wormbase.org)) contain biological network views. The publicly available networks are often limited in scope and the number of interactions; however, the
networks reflect a more systems based exploration of genes and regulatory systems in an organism. NCBI maintains the Gene Expression Omnibus (GEO) database containing freely available microarray expression data that can be used to construct de novo gene-to-gene correlation networks. Since 1995, the Kyoto Encyclopedia of Genes and Genomes (KEGG) collects and curates biological pathway data, and presents them in manually drawn sub-networks and individually downloadable KGML files (8). Data formats for biological networks range from several flavors of XML (e.g., KGML, SBML (9), etc.), several forms of graph adjacency lists (e.g., Cytoscape node and edge files, Gephi inputs, etc.) and other variations.

**Existing biological network analysis pipelines**

The growth of biological network repositories has inspired the development of tools for integrating data from different repositories. For example, STRING (Search Tool for the Retrieval of Interacting Genes/Proteins) combines protein-protein interactions from multiple online databases for a given gene product and provides the user with a 2D network visualization and downloadable network adjacency lists for networks up to 500 links (10). However, STRING is strictly a web portal into pre-computed data and is not a computational tool. Cytoscape, released in 2003, was also developed to integrate bio-molecular interaction networks (11), usually by loading one network and then retrieving node or link attributes by connecting to a few online biological databases. Another tool Gephi, marketed as “Photoshop for graphs”, focuses on analyzing and visualizing networks one at a time (12). Unlike Cytoscape, Gephi provides 3D layouts, but only 2D rotations of the 3D layouts are allowed. As the size and complexity of biological networks grow, live 3D layouts can provide a better view of the data, although they are difficult to be included as static publication figures. In
2008, Pavlopoulos introduced Arena3D for visualizing biological networks in 3D space, using the 3rd dimension to encode another layer of data. Arena3D organized graphs in a hierarchy where each layer is a different type of biomolecule (e.g., gene, protein, and regulatory element) (13).

While graph visualization tools have been available for several years, a few challenges remain. Around 2006, Eytan Adar surveyed users of graph visualization tools. One of the main user complaints was the separation between graph analysis and visualization. Constructing a network usually requires a certain amount of programming or preprocessing, and the network visualization step only provides the visual feedback of the results from the preprocessing. If a user is inspired by a new question after seeing the networks, he or she must go back to the preparation step again to explore the new question. In response to this limitation, Eytan Adar developed the GUESS system at IBM and presented it at the CHI Conference (14). GUESS contained a graph centric language called Gython. However, the software seems to have been abandoned with no new updates since 2007. Chapter 2 and 3 includes summary information of existing graph software in the computational biology field.
CHAPTER 2 : MANGO: COMBINING AND ANALYZING HETEROGENEOUS BIOLOGICAL NETWORKS

Modified from a paper published in *BioData Mining* 2016

Jennifer Chang, Hyejin Cho and Hui-Hsien Chou

**Abstract**

**Background**

Heterogeneous biological data such as sequence matches, gene expression correlations, protein-protein interactions, and biochemical pathways can be merged and analyzed via graphs, or networks. Existing software for network analysis has limited scalability to large data sets or is only accessible to software developers as libraries. In addition, the polymorphic nature of the data sets requires a more standardized method for integration and exploration.

**Results**

Mango facilitates large network analyses with its Graph Exploration Language, automatic graph attribute handling, and real-time 3-dimensional visualization. On a personal computer Mango can load, merge, and analyze networks with millions of links and can connect to online databases to fetch and merge biological pathways.

**Conclusions**

Background

In the present Big Data era, one of the great challenges is to be able to compare or integrate diverse data types. Modern biological research produces large and heterogeneous data sets, and there are many ways to categorize or display each type of data. The 2014 Nucleic Acids Research Database Special Issue counted 1552 online biological databases (15). It is often illuminating, even essential, to examine important biological problems using different types of data. For example, new discoveries often emerge when a biologist is able to interrogate gene expressions in the context of biological pathways (16). A common method to analyze related data relies on graphs, or networks, where data of various types are linked and key network features or subsets are identified (2,3,17).

Many graph analysis solutions have been written in Java, most notably Cytoscape (18). Started in 2002, Cytoscape has an impressive array of features. However, like other Java programs, the software slows to non-operational levels when handling large (>1M link) biological networks due to Java Virtual Machine limitations (19). Non-Java graph tools either do not provide analysis functions, or provide only libraries which users must incorporate into their own software solutions. Overall, many graph tools focus solely on one functionality, i.e., either analysis or visualization, and require users to integrate two or more tools for one project. Multi-graph comparison and integration are further complicated by differing graph attributes from heterogeneous data sets. Many tools ignore or limit the number of attributes associated with a graph. A comparison of currently available graph analysis and visualization software (12,14,18,20) is given in Table 2.1.
Table 2.1 Comparison of graph visualization software

<table>
<thead>
<tr>
<th>Software</th>
<th>Code</th>
<th>Graph Analysis Features</th>
<th>Visualization</th>
<th>Limitations</th>
</tr>
</thead>
</table>
| Cytoscape (v. 3.2.1) | Java | • Many algorithms for systems biology  
 • Can add GO or KEGG attributes  
 • Plug-ins available | • 2D predetermined layout  
 • 3D predetermined layout (via plugin) | • Only can merge 2 graphs at a time  
 • 6 min to load a network with 4M link but no visual afterward |
| Gephi (v. 0.8.2)     | Java | • Intuitive graph statistics  
 • Automated graph algorithm citation  
 • Generalized for all types of graphs  
 • Plug-ins available | • 2D and 3D layouts but graphs cannot be rotated in 3D  
 • Graph layout animation helps maintain mental map | • Cannot display multiple graphs on one screen  
 • Limited by JVM constraints; cannot load a network with 4M links |
| GUESS            | Java | • GYTHON, a language for graph analysis  
 • Can map information attributes to visual attributes | • 2D layout only  
 • Updates with user commands | • Cannot be run on MacOS 10.9, Windows 7, or RedHat Linux 6.0 |
| GraphViz         | C    | • No graph analysis capabilities                                                     | • Rich set of predetermined 2D layouts  
 • Streamlined command line interface | • Not an interactive system  
 • Cannot efficiently handle graphs over 100 nodes |
| Neo4j (v. 2.1.7)   | Java | • Graph database system  
 • Cypher graph query language  
 • Queries are based on a combination of topology and attributes | • Relies on JSON for visualization  
 • 2D layouts only  
 • Must click a node or link to see its attributes on a separate panel | • Designed as a database rather than for visualization  
 • Nodes are only labeled by numbers  
 • The whole database supports one huge graph |
| Tulip            | C++  | • A set of C++ library for graph analysis  
 • Can also be run as a stand-alone program  
 • Customizable python plug-ins | • 2D visualization  
 • 3D is available via a plug-in  
 • Had some 3D layout algorithms | • More useful to users who program C++ or python directly  
 • More as an analysis tool |
| NetworkX (v. 1.6.1) | Python | • Python module for graph analysis  
 • Rich set of network algorithms | • Must export to other software or modules for visualization | • More as an analysis tool |
| Mango (v. 1.10)    | C++  | • Provides general graph mathematics  
 • Heterogeneous graph analysis with ease  
 • Takes ~30 s to load a 4M link network | • Interactive 3D layouts and controls  
 • Real-time large graph visualization  
 • User customizable visual attributes | • Does not yet have plug-in feature  
 • Does not yet use GPU speedup  
 • Has limited preset layouts so far |

Benchmarks were performed on a 2010 Mac mini that has 8 Gb RAM and runs 64-bit MacOS X 10.9
To address these limitations, we have developed a stand-alone graph analysis and visualization software environment called Mango to aid biologists and other researchers efficiently integrate and explore heterogeneous networks larger than previously possible. A 4 million link network can be loaded into Mango in 30 seconds on a Mid 2010 Mac mini computer with a 2.4 GHz (Gigahertz) Intel Core 2 Duo processor and 8GB RAM (random access memory). As a comparison, Cytoscape took 6 minutes to load that same network file on the same computer using its default configurations. Mango possesses the scalability to handle larger networks, the expressive power of a new Graph Exploration Language (Gel) and the convenience of unlimited graph attributes with automatic graph attribute merging and promotion. Within the integrated development environment, Gel commands can be edited, run line-by-line, or saved as scripts to reproduce results. Script files enhance the speed and reproducibility of analysis \cite{sandve2013ten}. Mango provides both comprehensive graph analyses and real-time 3-dimensional (3D) visualization. Mango is a cross-platform C++ program that runs on Mac OS X 10.9 or later, Windows 7 or later, and many Linux variants. It is freely available from our website (http://www.complex.iastate.edu/download/Mango) and the continually updated Mango User Guide is hosted at GitBook (http://www.gitbook.com/book/j23414/mango-user-guide).

Implementation

The Mango user interface

Mango updates its display in real-time at each stage of analysis to facilitate the integration and modification of multiple large networks. Mango contains a primary window divided into four areas (Figure 2.2). The graph canvas area is fully interactive, responding to
mouse and keyboard actions to zoom, move, rotate, and auto-layout the displayed graphs. By dragging and rearranging tabs, multiple graphs can be viewed simultaneously, easing multi-network comparison. Mango functions are mostly carried out through its command console or Gel code editor. The Gel code editor allows commands to be run line-by-line, edited, and saved as Gel script files. Gel script files can then be shared among researchers, reproducing a 3D layout or network analysis pipeline. Finally, the data area lists currently loaded graphs, their sizes and attributes. Interactive real-time network visualization in Mango helps hone and refine each step of analyses. Mango is built on multiple layers of implementation that are seamlessly combined to form an integrated solution for graph analysis (Figure 2.1).

![Figure 2.1 System architecture.](image)
Figure 2.2 Mango user interface. The main window is divided into four areas: data list (left), graph canvases (middle 3D visualizations), Gel editor (bottom left), and Gel command console (bottom right). Shown in the graph canvas area are the following networks: Left column: WGCNA correlation network, KEGG biological pathway network and their combined networks; Middle column: crown-plot of the intersection network between correlation and pathway networks and extracted hub genes sub-network; and Right column: hub and in-between genes laid out in a bipartite graph where nodes are labeled by gene names.
The Graph Exploration Language (Gel)

A graph is defined as a set of nodes ($V$) and links ($E$) where a node represents some entity and a link represents a relationship between a pair of entities. In practice, graphs also have added annotations called attributes. Currently, Gel provides four basic data primitives string, int, float and double as well as aggregate data types node ($V_{attr}$), link ($E_{attr}$) and graph.

$$G = \{V, E\} \quad \text{where} \quad V = \{v_1, v_2, v_3, \ldots, v_n\}$$

$$E \subseteq \{(v_i, v_j) | v_i, v_j \in V\}$$

$$V_{attr} = \{a_1, a_2, a_3, \ldots, a_{m_1} | \text{type}(a, i) \in \{\text{int, float, double, string}\}\}$$

$$E_{attr} = \{a_1, a_2, a_3, \ldots, a_{m_2} | \text{type}(a, i) \in \{\text{int, float, double, string}\}\}$$

Each nodes and link type can have any number of attributes of the four primitive types in any order, and each of the attributes has a distinct name and specified data type (e.g. string, int, float, and double). The first attribute in a node type must be a string to denote the node name, and a link is identified by a pair of node names. All node and link attributes have default values, which are usually zero for numeric types or the empty string, but users can define other default values during node and link type declarations. Graphs are defined based on a pair of node and link types. For example, the following Gel code defines and initializes two graphs $G_A$ and $G_B$, also shown in Figure 3a. Node type and link type are defined with the given attributes inside parentheses and brackets; the brackets denote non-directional link types (whereas arrows <> denote directional link types). For example, $G_A$ is declared with $ntA$ and $ltA$, and is also initialized by the graph literals enclosed within the braces.
Other than defining a graph in the native graph exploration language, Mango can read graph data in tabular or CSV (comma separated values) format using the import command. A properly formatted graph file lists nodes with their attributes and then links with their attributes. A single line containing a hyphen separates the node list from the link list. The full description of the import command is in the Mango User Guide.

Mango system-defined graph attributes are appended to user defined attributes. The system-defined attributes are related to the 3D visualization of a network and define such attributes like node position, node color, or link width. Therefore, generating any 3D visualization is a matter of mapping user defined information attributes to system defined visualization attributes (21). By dynamically changing these mappings, animations and simulations can be accomplished in Mango. A full listing of the visualization attributes is in the Mango User Guide.
Figure 2.3 Graph Exploration Language examples. (a) Graphs A and B have different node attributes. Graph C is the result of attribute merging and promotion of A and B. (b) Graph mathematics. Given two graphs A and B, the dotted addition $A + B$ combines nodes and links from graph A and graph B. The non-dotted addition $A + B$ combines graph A with links of Graph B whose end nodes are already contained in graph A. Graph subtraction works similarly. Graph mathematic results depend on operand order; attribute merging and promotion are handled automatically as described in the main text but are not shown in this figure.

Standards for combining heterogeneous graphs

When combining two or more graphs, much of the confusion stems from what will happen to the nodes and links. Since a graph contains both node and link sets, our formally defined dotted and non-dotted graph mathematic operators allow users to specify node-centric or link-centric operations precisely. Recall the two graphs $G_A$ and $G_B$.

$$G_A = \{V_A, E_A\} \quad \quad G_B = \{V_B, E_B\}$$

Merging nodes and links is represented by the dotted addition.

$$G_A \cdot +G_B = \{V_A \cup V_B, E_A \cup E_B\}$$
However, suppose that the user is only concerned with the nodes in $G_A$, such as a set of important genes, and merely wants to combine the new links between those genes from $G_B$. The non-dotted addition merges links from $G_B$ only between nodes already in $G_A$.

$$G_A + G_B = \{V_A, E_A \cup \{(v_i, v_j) | v_i, v_j \in V_A, (v_i, v_j) \in E_B\}\}$$

In a similar fashion, dotted and non-dotted subtraction between two graphs are defined as follows.

$$G_A \cdot - G_B = \{V_A \setminus V_B, (v_i, v_j) | v_i, v_j \in \{V_A \setminus V_B\}, v_i, v_j \in \{V_A \setminus V_B\}\}$$

$$G_A - G_B = \{V_A, E_A \setminus E_B\}$$

Other operations such as producing intersections and bipartite graphs are defined as follows.

$$G_A \cdot & G_B = \{V_A \cap V_B, E_A \cap E_B\}$$

$$G_A & G_B = \{V_A, E_A \cap E_B\}$$

$$G_A * G_B = \{V_A \cup V_B, E_A \cup E_B \cup \{(v_i, v_j) | v_i \in V_A, v_j \in V_B, v_i \neq v_j\}\}$$

$$G_A * G_B = \{V_A \cup V_B, E_A \cup E_B \cup \{(v_i, v_j) | v_i \in V_A, v_j \in V_B\}\}$$

The above mathematics can be extended across multiple graphs to create unions ($G_A + G_B + G_C$), differences ($G_A - G_B - G_C$ or $G_A - G_B - G_C$), intersections ($G_A \cdot & G_B \cdot & G_C$) and inverse graphs ($G_A * G_A - G_A$). The graph operations can be mixed and matched to produce more complex results. Figure 2.3b demonstrates a few of the graph mathematics visually.

When graphs are combined in mathematical operations, attributes from two graphs might conflict. For example, the link between $b$ and $d$ nodes in $G_A$ may have a weight attribute of 0.4 while the link between $b$ and $d$ nodes in $G_B$ may have a weight attribute of 0.3. Gel handles attribute conflicts by giving preference to the left operand. During the
operation $G_A + G_B$, the left operand $G_A$ takes precedence and the resulting graph will have weight value 0.4. An exception to this rule is when the conflicting attributes in $G_A$ happen to be at their default values (default values can be defined by users). In those cases, the attributes of graph $G_B$ will be copied. This automatically merges useful non-default information from $G_B$ into the resulting graph.

When heterogeneous graphs are combined, their unique attributes can be selectively preserved. Recall that the nodes in $G_A$ have attributes $id$ and $count$ while nodes in $G_B$ have attributes $id$ and $tag$.

$$V_{A,attr} = \{id, count\} \quad E_{A,attr} = \{weight\}$$

$$V_{B,attr} = \{id, tag\} \quad E_{B,attr} = \{weight\}$$

Because nodes in $G_B$ only share the $id$ attribute with $G_A$, when $G_B$ is added to $G_A$ as in $G_A + G_B$, the $count$ attribute of nodes copied from $G_B$ is automatically set to the default value 0 but their $tag$ attribute is ignored. To preserve both $G_A$ and $G_B$ attributes, users can define a new node type that includes all attributes. This is called attribute promotion. In our example, a new node type containing $id$, $count$ and $tag$ attributes is defined and used by the new $G_C$ to receive all attributes from $G_A$ and $G_B$.

$$V_{C,attr} = \{id, count, tag\} \quad E_{C,attr} = \{weight\}$$

However, simply writing $G_C = G_A + G_B$ will not work as the $tag$ attribute from $G_B$ is already lost after the addition of $G_B$ to $G_A$ but before the result is assigned to $G_C$. The correct steps to preserve graph attributes during heterogeneous graph mathematics are demonstrated below (Figure 2.3a):
Flexible node and link type definition coupled with an intuitive set of attribute promotion and merging rules ease the combination of heterogeneous graphs in Gel. Thus, users can focus on graph level operations instead of attribute level selection, sorting, and merging.

Many graph analyses require traversing all nodes and links to perform a calculation based on graph attributes or topology. Gel provides the `select` command to pull out a subgraph based on user-defined conditions. These conditions can be related to stored attribute values or topology properties. Gel also allows mapping or computing new attribute values across a graph on a per-node or per-link basis with the `foreach` command, which efficiently applies a set of user-defined calculations across all nodes or links that optionally meet certain conditions. The same command can also be used to tally attribute values across all nodes and links. The following demonstrates the two types of Gel commands:

```
graph(nt,lt) hubs = select node from A where in+out>3;
graph(nt,lt) thresh = select link from A where weight>0.2;
foreach link in thresh where weight>1.0 set weight=1.0;
foreach link in thresh set _r=weight, _g=weight, _b=weight;
foreach node in hubs where type=="gene" set _radius=0.2+(in+out)/2.0,count++;
```
also provided. To explore all Gel commands and functions, type the help command in Mango or consult the online User Guide.

The Mango system and its Graph Exploration Language are data agnostic, meaning that any type of network can be loaded and analyzed — users have total control of node and link attribute definitions and their associations within Mango. Our goal is to make this software widely available to all researchers and promote its use in solving ever more complex biological research problems.

**KEGG Connect**

The KEGG Connect dialog demonstrates how Mango can fetch network data directly from online biological databases. KEGG Connect queries the KEGG (Kyoto Encyclopedia of Genes and Genomes) database (http://www.genome.jp/kegg) and selectively downloads pathways grouped by organisms. Within the downloaded pathway, nodes maintain their 2-dimensional (2D) coordinates from the KEGG visualization. The nodes are colored red, blue, green and yellow representing pathway maps, compounds, genes, and orthologs respectively (Figure 2.4). Multiple pathways can be downloaded either as individual networks or as one merged network. If multiple networks are merged, each pathway will be given a different z coordinate value, so the pathways are layered in 3D space. We intend to connect Mango to more biological databases soon.
Figure 2.4 KEGG Connect. (Left) The KEGG Connect dialog lists currently available organisms and pathways in the KEGG database. Users can fetch multiple pathways individually or merge them into one network by checking the "Merge Fetched Pathways" box. (Middle) Mango maintains the x-y coordinates from KEGG website drawing and colors nodes red (pathway map), green (enzymes), blue (compounds), and yellow (orthologs). (Right) Corresponding KEGG website drawing for the same pathway.

Results and Discussion

We present a few network analysis examples to illustrate the use of Mango in this section. Examples of comparing different types of biological networks and the scalability of Mango to large networks are provided.

Network data collection

Four large *E. coli* network data sets were collected. The *corr* 4M link network was computed using the WGCNA (weighted gene coexpression network analysis) package in R (22) on microarray data measuring the expression of 4454 *E. coli* genes in cells grown under 10 different conditions (GSE61736, (23)). The *path* biological pathways of *E. coli* were downloaded from the KEGG database (http://www.genome.jp/kegg) and combined into a single pathway network. The *go* network was constructed using *E. coli* GO (gene ontology) information retrieved from the gene ontology website (https://geneontology.org/page/download-annotations); *E. coli* genes that share at least one GO term are linked. Finally, the protein-protein interaction (*ppi*) network was retrieved from
the supplementary materials of a 2014 paper (24). Sizes and attributes for the 4 large networks are summarized in Table 2.2.

<table>
<thead>
<tr>
<th>Network</th>
<th>Nodes</th>
<th>Links</th>
<th>Node attribute(s)</th>
<th>Link attribute(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>corr</td>
<td>4,454</td>
<td>4,408,269</td>
<td>gene name</td>
<td>WGCNA correlation weight</td>
</tr>
<tr>
<td>path</td>
<td>2,353</td>
<td>6,703</td>
<td>gene name</td>
<td>None</td>
</tr>
<tr>
<td>go</td>
<td>3,764</td>
<td>2,208,090</td>
<td>gene name</td>
<td>Count and string of shared GO terms</td>
</tr>
<tr>
<td>ppi</td>
<td>2,042</td>
<td>3,888</td>
<td>gene name</td>
<td>Source of evidence (Y2H, LIT or both)</td>
</tr>
</tbody>
</table>

Unconnected nodes and duplicate links have been removed from some of the networks. In all 4 networks, nodes are identified by gene names and differ in their link attributes.

Large heterogeneous network comparison

For all networks, nodes are identified by gene names with no additional attributes, thus the following node type declaration can be shared among the networks:

```cpp
node(string name) nt;
```

All networks have undirected links but differ in their link attributes (the path network does not contain any link attributes), thus the following 4 link type declarations are used to load the different networks:

```cpp
link[float corr_weight] corr_lt;
link[] path_lt;
link[int count, string go_terms] go_lt;
link[string source] ppi_lt;
```

After the node and link type declarations, the `corr` network, `path` network, `go` network, and `ppi` network can be imported into Mango for all-to-all network comparisons:

```cpp
graph(nt,corr_lt) corr = import("wgcna.csv");
graph(nt,path_lt) path = import("kegg.csv");
graph(nt,go_lt) go = import("go.csv");
graph(nt,ppi_lt) ppi = import("ppi.csv");
```

For the integration of the networks, a common link type including all available link attributes is declared:
Once the networks are loaded into Mango, Gel mathematics allow network integration and comparisons. For example, the comparison of the \textit{corr} and \textit{path} networks are visualized in the top two panels in the left column. The top middle panel in Figure 2.2 is the result of the following Gel intersect operation.

\begin{verbatim}
// intersection of path and corr networks
graph(nt,c_lt) intersect = path .& corr;
\end{verbatim}

The \textit{corr-path} intersection network contains 961 links with 1020 nodes. The all to all comparisons of these four networks were completed in Mango and the common links among the networks were summarized in Figure 2.5. All possible intersections among the four \textit{E. coli} networks can be worked out with a few lines of Gel code each. Bench-marked time for different types of Gel mathematics between the large \textit{corr} and \textit{path} networks are listed in Table 2.3.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{network_comparisons.png}
\caption{Biological network comparisons. Link intersections among the \textit{corr}, \textit{path}, \textit{go} and \textit{ppi} networks. The intersections were worked out using Gel commands. WGCNA is the gene-to-gene correlation network \textit{corr} computed from \textit{E. coli} microarray data. PPI is the protein-protein interaction network \textit{ppi} of \textit{E. coli}. GO is the network \textit{go} that connects any two \textit{E. coli} genes sharing at least one gene ontology term. KEGG is the entire KEGG biological pathway network \textit{path} of \textit{E. coli}.}
\end{figure}
Table 2.3 Benchmarking the speed of Gel mathematics on massive graphs.

<table>
<thead>
<tr>
<th>Gel operation</th>
<th>Time (in seconds)</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>4M + = 8K</td>
<td>0.92, 0.35, 0.27, 0.60, 0.56</td>
<td>0.54</td>
</tr>
<tr>
<td>8K + = 4M</td>
<td>1.25, 1.15, 1.03, 1.02, 1.02</td>
<td>1.09</td>
</tr>
<tr>
<td>4M - = 8K</td>
<td>0.52, 0.33, 0.62, 0.33, 0.25</td>
<td>0.41</td>
</tr>
<tr>
<td>8K - = 4M</td>
<td>1.09, 1.28, 1.09, 1.16, 1.19</td>
<td>1.16</td>
</tr>
<tr>
<td>4M += 8K</td>
<td>0.69, 0.60, 0.57, 0.31, 0.40</td>
<td>0.51</td>
</tr>
<tr>
<td>8K += 4M</td>
<td>12.06, 12.09, 12.05, 12.23, 12.32</td>
<td>12.15</td>
</tr>
<tr>
<td>4M -= 8K</td>
<td>0.55, 0.41, 0.25, 0.26, 0.32</td>
<td>0.36</td>
</tr>
<tr>
<td>8K -= 4M</td>
<td>0.90, 0.85, 0.83, 0.98, 0.74</td>
<td>0.86</td>
</tr>
<tr>
<td>4M *= 8K</td>
<td>22.94, 23.74, 23.35, 22.98, 23.03</td>
<td>23.21</td>
</tr>
<tr>
<td>8K *= 4M</td>
<td>36.75, 35.33, 35.23, 35.38</td>
<td>35.67</td>
</tr>
<tr>
<td>copy = 4M</td>
<td>7.90, 7.76, 7.85, 7.73, 7.87</td>
<td>7.82</td>
</tr>
<tr>
<td>copy = 8K</td>
<td>0.30, 0.52, 0.45, 0.34, 0.29</td>
<td>0.38</td>
</tr>
</tbody>
</table>

The 4M link network is the gene correlation network generated by WGCNA. The 8K link network is the combined KEGG pathway network. Benchmarks were performed consecutively on a 2010 Mac mini that has 8 Gb and runs 64-bit Mac OS X 10.10. The time to copy the networks is also listed.

Flexible real-time network exploration and visualization

Over-plotting of nodes and links becomes more of a challenge as network sizes get bigger. For example, the corr and path networks and their combination can be visualized in Mango but provide limited biological interpretation (the left column of panels in Figure 2.2). In this example, we continue to explore the intersection of the two networks by querying certain node and link attributes, imposing thresholds to reveal important features, and map these features to network visualization.

First, we arrange all nodes in the intersection network along a circle in the x-y plane and map the node connectivity to their z-axis coordinates. Nodes are assigned random colors and higher z-axis node colors are bled down the links to emphasize hubs. Nodes above a threshold are emphasized by increasing their radius and labeling them with gene names and connectivity.

```plaintext
Layout(intersect,"circle");
Foreach node in intersect set _z=(in+out);
Foreach node in intersect set _r=rand(),_g=rand(),_b=rand();
Foreach link in intersect where in._z>=out._z set _r=in._r,_g=in._g, _b=in._b;
```
Foreach link in intersect where in._z<out._z set _r=out._r, _g=out._g, _b=out._b;

// label nodes by connectivity to choose a threshold
Foreach node in intersect set _text=(in+out);

// emphasize hubs
Foreach node in intersect where (in+out)>10 set _radius=0.8;
Foreach node in intersect where (in+out)<=10 set _text="";

The resulting network layout, called a **crown-plot**, is shown on the top pane in the middle column of Figure 2.2. The hub genes and their links can be pulled into a new sub-network. The sub-network called hubs is then flattened and spread out using a force-directed layout built into the graph panel by right-clicking on the panel. The hub genes are raised one level. Genes that are not themselves hubs but connect two or more hubs are raised to a third level. The following Gel code accomplishes all these except the force-directed layout, which is performed by right-clicking on the panel:

```gel
auto hubs = select link from intersect where in._radius>0.3 || out._radius>0.3;
foreach node in hubs set _x=rand(-5,5),_y=rand(-5,5),_z=0;
/* right click on graph to start and stop force-directed algorithm */
foreach node in hubs where _radius>0.3 set _z=3;
foreach node in hubs where _radius<0.3 && (in+out)>1 set _z=6;
```

The 3-layer hubs network is shown in the lower panel in the middle column of Figure 2.2, which contains other genes on the bottom layer, hub genes on the middle layer and in-betweener genes on the top layer. It is worth mentioning that the in-betweener genes on layer 3 would have been obscured by other genes in a simple list of genes ordered by connectivity. We can further pull out the hubs and in-betweener into another sub-network for closer inspection with the following Gel code:
This sub-network is laid out as a bipartite graph shown on the right panel in Figure 2.2, with hubs on the left and the in-betweeners on the right. This example shows how to map informational attributes of a graph to its visual attributes using Mango. The resulting visual displays help the user decide threshold values, extract sub-networks of interest, and further explore the data.

**Microarray expression combined with KEGG biological pathways**

*E. coli* gene expression under control and multiple treatment conditions were measured by microarrays (GSE61736, (23)). A subset of the data containing one control and one treatment expression values was loaded into Mango and overlaid onto downloaded *E.coli* KEGG biological pathways. The expression data, *E. coli* KEGG pathways, and Gel script are available for download from https://github.com/j23414/Mango_Workshop.

The results of the visualization can be seen in Figure 2.6. Genes are colored green or red where their expression levels are up or down relative to the control condition. KEGG pathway components that do not have mapped gene expression values are colored gray. Compounds are colored blue and are largely ignored although they could be used to incorporate metabolomic concentration values. The Gel commands to color gene nodes are given below:

```plaintext
foreach node in sum where tr2==control && type="gene" set _r=0.2, _g=0.2, _b=0.2;
foreach node in sum where tr2>control && type="gene" set _r=0, _g=1, _b=0;
foreach node in sum where tr2<control && type="gene" set _r=1, _g=0, _b=0;
```
Figure 2.6 Gene expression combine with KEGG. A 3D KEGG network visualization comparing the *E. coli* gene expression values obtained under a treatment condition and a control condition. In addition to coloring and resizing the genes (i.e., node) of the network based on expression changes related to the control, pathway links are also highlighted in green or red depending on up or down expressed genes they connect in a pathway. The highlighted links allow a whole pathway to be easily discerned as up or down regulated.

More than coloring nodes in a network, we can color the links and thereby highlight entire pathways that are up or down-regulated. This is possible because KEGG pathways also contain gene to gene links, not just gene to compound links.

```gel
foreach link in sum where in._r==out._r && in._r>0.5 set _r=1,_width=4;
foreach link in sum where in._g==out._g && in._g>0.5 set _g=1,_width=4;
```

The final network can be saved and reloaded to regenerate the same 3D visualization.

```gel
save "sum.txt",sum;
clear;   // clears all data objects
run "sum.txt";   // reloads the sum network
```

Mango networks are saved natively into Gel commands, thus running the saved code recreates the original graphs in Mango. In addition, the networks can be exported to tabular data using the `export` command. The tabular data can then be read by many other software
programs, e.g., Excel, R, MatLab, Cytoscape, and other graph software or databases. Full
descriptions of the interoperability and other features of Mango are available in the User
Guide.

**Conclusion**

We have developed a powerful new program Mango for multi-network analysis and
visualization. Mango enables scientists to test hypotheses on large heterogeneous networks,
identify crucial features, and extract analysis results all within its integrated environment.
Compared with existing programs, Mango extends the capability and convenience of large
heterogeneous data analysis on a personal computer.

The Mango system was designed to be data agnostic, meaning that any type of
network data can be loaded and analyzed — users have total control on node and link
attribute definitions and their associations within Mango. Mango can load networks with
millions of links, integrate and explore large amounts of data following Gel commands, and
help users deduce predictions or outcomes that can be validated in labs. It is our goal to make
this software widely available to all researchers to promote its use in solving ever more
complex biological research problems. As Mango developers, we will continue to provide
support and further develop the software according to user needs.

**Availability and Requirements**

- **Project name**: Mango 1.23
- **Project home page**: http://www.complex.iastate.edu/download/Mango/
- **Operating system(s)**: Mac OS X 10.9 or later, Windows 7 or later, and Linux variants.
  Both 32- and 64-bit operating systems are supported
- **Programming language**: C++
- **Other requirements**: An internet connection for online database access.
• **License**: Free versions available; specific license agreement included with each distribution.

• **Any restriction to use by non-academics**: Specific restrictions included with each distribution and license agreement.

**List of abbreviations**

- Gel: graph exploration language; RAM: random access memory; GHz: Gigahertz; CSV: comma separated values; 3D: 3-dimensional; 2D: 2-dimensional; WGCNA: weighted gene correlation network analysis; KEGG: Kyoto Encyclopedia of Genes and Genomes; GO: gene ontology; PPI: protein protein interaction

**Ethics approval and consent to participate**

Not applicable.

**Consent for publication**

Not applicable.

**Availability of data and materials**

https://github.com/j23414/Mango_Workshop

**Competing interests**

JC and HHC have founded a software company and have licensed Mango from Iowa State University for further development. A free and functional Mango will always be made available to the public which can be downloaded and used by anyone including commercial entities.

**Funding**

This work is partially supported by the National Science Foundation grant DBI-0850195 and the Iowa State University Plant Sciences Institute Scholar grant to HC. JC is partially supported by the James Cornette Research Fellowship. None of these funding
agencies had any role in the design of the study, data collection, analysis and interpretation, or in writing the manuscript.

Author’s contributions

JC developed the Mango system and drafted the manuscript. HJC carried out the *E. coli* studies and collected the microarray data. HHC developed the Gel language and revised the manuscript. All authors read and approved the final manuscript.

Acknowledgements

We thank Dr. Jo Anne Powell-Coffman, Zebulun Arendsee, and Kannan Sankar for proof-reading the draft manuscript and offering valuable suggestions.
CHAPTER 3 : CAVATICA: A PIPELINE FOR IDENTIFYING AUTHOR ADOPTION TRENDS AMONG SOFTWARE TOOLS OR METHODS FROM LITERATURE

Adopted from a paper submitted to PLOS ONE 2017

Jennifer Chang and Hui-Hsien Chou

Abstract

Cavatica is a pipeline to identify author adoption trends among software tools or methods based on automatic PubMed and PubMed Central literature searches and subsequent multi-network analysis. By employing heterogeneous networks representing both papers and authors, Cavatica can identify collaborating communities and authors who are more devoted to a tool or methodology. Cavatica automatically builds networks from different literature search results, and generates Mango Graph Studio scripts to combine multiple heterogeneous networks into a single network. The scripts enable users to deduce the adoption trend of authors among the tools or methodologies represented by multiple heterogeneous networks obtained from different search terms. We have validated the Cavatica pipeline on two known trends in gene expression measurement methods and genome editing methods where a straight count of papers was usually insufficient to identify them. We then applied Cavatica on two nontrivial cases: the comparison of 9 different network analysis software tools to identify the adoption trends of authors among them, and the comparison of 18 probiotic bacteria commonly found in health supplements to identify their popularity as shown in the scientific literature.

Introduction

Many software tools and methods have been used for biological research. However, choosing and comparing software tools or scientific methods for a specific research purpose
can be a time-consuming task depending on the criteria used. One such criteria is to compare their search results in the literature. Since the number of publications returned may depend on the release date of the software or method, this selection criteria may favor older work. As an example, it is commonly known that the RNA-Sequencing technology (RNA-Seq) based on next-generation sequencing became popular since 2008 and overtook microarrays as the preferred technology for measuring gene expression (25,26). The benefits of RNA-Seq over microarray include the detection of novel transcripts, broader dynamic range, and increased specificity and sensitivity (27). However, when searching on PubMed, as of this writing, there are 56,322 microarray papers and only 9,769 RNA-Seq papers published since 2008, which does not reflect the popularity of the RNA-Seq technology. This example suggests that a method to identify author adoption trends among technologies based on their history of publication may better reflect the trending technology than straight paper counts.

PubMed currently indexes more than 26 million items from MEDLINE, life science journals, and online books (28). Many articles are also assessable on PubMed Central (PMC) (29), which indexes full-text in addition to the metadata indexed by PubMed. A simple PubMed or PMC search can return hundreds to thousands of hits that can easily overwhelm the users. In addition, researchers often search with multiple different but related terms that gradually define their specific research interests, and obtain multiple search results along the way. We hypothesized that an automatic multi-network analysis of multiple search results may provide efficient routes to identify author adoption trends within and across multiple search results.

To test this hypothesis, we have implemented a multi-network analysis pipeline called Cavatica. To validate our pipeline, we conducted searches on PubMed and PMC to identify
the author adoption trends between microarray and RNA-Seq technologies. We also validated Cavatica by comparing TALEN, CRISPR/Cas9, and zinc finger nucleases methods for genome editing.

Subsequently, to demonstrate the broad utility of the Cavatica pipeline, we conducted searches on PubMed and PMC using the names of nine network analysis software tools. While more than nine network analysis software tools are available, we chose Ingenuity Pathway Analysis (IPA) (30), Cytoscape (18), Pathway Studio (31), Gephi (12), iGraph (32), VisANT (33), GraphViz (34), and Neo4j (35). We use the comparison of the nine software tools to explain the Cavatica pipeline. We analyzed these networks to answer the following key questions: 1) are there any author adoption trends among the nine software tools; and 2) Are author adoption trends consistent between PubMed and PMC search results?

Finally, we also compared 18 popular probiotic bacteria strains in human health supplements using Cavatica to show its scalability up to 18 search terms. The Cavatica pipeline code and instructions are provided on GitHub

https://www.github.com/j23414/cavatica. The Cavatica pipeline merges networks using the Mango Graph Studio community edition, which is freely available from


Networks built from the literature have been useful for the understanding of collaboration and publishing productivity among research groups. In one study, the collaboration networks built from 12,170 papers produced by Alzheimer’s Disease centers were analyzed to measure the increase in cross-center collaborations (36). In another study of 888 papers, it was determined that the coauthor degree centrality and betweenness centrality was positively correlated with the citation count of an article, predicting the research impact
of that article (37). While the second study started with an author-paper network which included both authors and papers, both studies employed co-authorship networks which include coauthor information extracted from the literature. Some have also tried to analyze the adoption of new ideas by adding a time component to the co-authorship network analysis, e.g., a 2017 study compared the adoption of two methods for confounder information in the field of pharmacoepidemiology: disease risk score (43 papers) and high-dimensional propensity score (44 papers) (38). They produced a co-authorship network to characterize the adoption trend of the two methods across several years.

Previously, most co-authorship networks were created with homogeneous nodes to represent the same type of entity—authors. As we will show in the following, a combined co-author publication network (also called author-paper network) that contains heterogeneous nodes—both authors and their publications—makes it easier to identify if some authors have collaborated on several papers and if they have collaborated with the same or different group of co-authors on each paper. Furthermore, we will show that combining and analyzing multiple heterogeneous networks produced by different searches may also reveal if certain authors are more devoted to certain ideas by bringing those ideas to new groups.

Results

Validating Cavatica results by confirming two known trends

Cavatica fetched microarray and RNA-Seq papers from PubMed and PMC to identify the author adoption trends between them. An author who published a microarray paper and subsequently a RNA-Seq paper is counted as one author adoption of RNA-Seq. An author may switch back and forth between methods, and each transition is counted separately. When
a paper mentions multiple tools or methods, Cavatica counts author adoption by comparing it to subsequent publications. The total publication count of each method since 2008 on PubMed and PMC and the author adoption trends between microarray and RNA-Seq are shown in Table 3.1. Even though RNA-Seq is popular nowadays, there are still more microarray papers than RNA-Seq papers since 2008. It is not until we used Cavatica to compare the author adoption between the two methods that we can clearly see RNA-Seq adoption leading microarray, which also validates Cavatica results.

Table 3.1 Total paper count and author adoption trend for microarray and RNA-Seq

<table>
<thead>
<tr>
<th></th>
<th>PubMed Author Adoption</th>
<th>PMC Author Adoption</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Microarray RNA-Seq</td>
<td>Microarray RNA-Seq</td>
</tr>
<tr>
<td>Microarray</td>
<td>9856</td>
<td>9835</td>
</tr>
<tr>
<td>RNA-Seq</td>
<td>4503</td>
<td>4732</td>
</tr>
<tr>
<td>Net Change</td>
<td>-5353</td>
<td>-5103</td>
</tr>
<tr>
<td>Papers (2008-2016)</td>
<td>56,322</td>
<td>160,272</td>
</tr>
</tbody>
</table>

Genome editing is a very important technique for modern biological and medical research. The oldest genome editing method is based on the Zinc Finger enzyme. A few years earlier when TALEN genome editing was first discovered, it was the popular genome editing method until the discovery of the CRISPR/Cas9 method, which is currently the most popular genome editing method. Therefore, we expect the order of popularity of genome editing tools to be CRISPR, TALEN, and Zinc Finger. To validate Cavatica, we used it to generate the author adoption trend table comparing CRISPR, TALEN, and Zinc finger search results. In Table 3.2, we summarize the adoption trend and compared it with the PubMed and PMC paper counts for each method. The net change row in Table 3.2 represents the total author influx to a method minus its total author outflux to the other methods based on publication years; it does show the expected popularity order of the three methods.
Nevertheless, the straight paper count does not, which shows the Cavatica generated author adoption table may better reflect a preferred tool or method in a field over a straight PubMed or PMC paper count.

**Table 3.2 Total paper count and author adoption trend for CRISPR, TALEN, and Zinc Finger genome editing methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>PubMed Author Adoption</th>
<th>PMC Author Adoption</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRISPR</td>
<td>TALEN</td>
</tr>
<tr>
<td>CRISPR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TALEN</td>
<td>660</td>
<td>187</td>
</tr>
<tr>
<td>Zinc Finger</td>
<td>3190</td>
<td>716</td>
</tr>
<tr>
<td>Net Change</td>
<td>3403</td>
<td>10</td>
</tr>
<tr>
<td>Papers (1996-2016)</td>
<td>5759</td>
<td>697</td>
</tr>
</tbody>
</table>

*The net change row is added since there are more than two methods compared.*

**Applying Cavatica to two nontrivial cases**

After we validated Cavatica pipeline by confirming its computed author adoption tables are consistent with known popularities of different method in two application domains, we set out to test Cavatica on two additional application domains regarding network analysis software and probiotic bacteria and yeast.

**Analyzing the adoption trend of nine network analysis software tools**

Cavatica fetched papers from PubMed and PMC containing the names of at least one of nine different network analysis software tools. The nine software tools were listed earlier, and their author adoption tables based on PubMed and PMC results are shown in Table 3.3 and Table 3.4, respectively. A detailed description of the analysis pipeline with filtering and preprocessing steps is given in the Methods section.
According to PubMed search results, Cytoscape and Neo4j are the only tools with a net author gain. Cytoscape is a known popular network analysis tool in bioinformatics and Neo4j is just starting to get some traction in bioinformatics as a large graph database, thus it is reasonable that these two tools are popular. Surprisingly, the PMC results showed that Ingenuity Pathway Analysis, Gephi, and iGraph also have a net gain of authors. There was a larger number of iGraph papers returned from PMC (see Table 3.6 below) which may have
been due to iGraph being listed in the methods section and not in the abstract or title of the papers; this can explain the popularity of iGraph in PMC analyses. However, there are no obvious reasons why Ingenuity Pathway Analysis and Gephi are also popular among the network analysis software tools. The more detailed adoption trend observations are possible with PMC searches as PMC stores not only the metadata of each paper but also its full text.

Comparing the popularity of 18 probiotic bacteria and yeast strains in the literature

Names of 18 bacterial and yeast strains from three probiotic supplements available in the market (Pro-15, Garden of Life, and Schwartz) were searched on PubMed and PMC and analyzed by Cavatica; the resulting author adoption table was automatically generated. This dataset was challenging because Cavatica was dealing with 18 different search terms and many PMC or PubMed papers that were returned to Cavatica contain multiple search terms. Cavatica can deal with multi-term papers and count author adoption for each term based on a comparison to the subsequent publication of that same author.

The Cavatica pipeline could fetch all 18 search results, construct networks, and generate the corresponding author adoption table. Only the net change columns instead of the entire author adoption table are shown in Table 3.5. We can observe from the table that strains in the *Bifidobacterium* genus were gaining research attention, and the traditional strains in the *Lactobacillus* genus for making yogurts were losing research attention in recent years. This shows how Cavatica can be scaled up to search and compare 18 terms and deal with situations where papers contain multiple search terms.
Table 3.5 PubMed and PMC author adoption trends among probiotics

<table>
<thead>
<tr>
<th>Probiotic</th>
<th>PubMed Net Change</th>
<th>PMC Net Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bacillus subtilis</td>
<td>71</td>
<td>2859</td>
</tr>
<tr>
<td>Bifidobacterium bifidum</td>
<td>557</td>
<td>1027</td>
</tr>
<tr>
<td>Bifidobacterium breve</td>
<td>652</td>
<td>534</td>
</tr>
<tr>
<td>Bifidobacterium infantis</td>
<td>231</td>
<td>-177</td>
</tr>
<tr>
<td>Bifidobacterium lactis</td>
<td>527</td>
<td>196</td>
</tr>
<tr>
<td>Bifidobacterium longum</td>
<td>244</td>
<td>346</td>
</tr>
<tr>
<td>Lactobacillus acidophilus</td>
<td>-194</td>
<td>62</td>
</tr>
<tr>
<td>Bactobacillus brevis</td>
<td>-88</td>
<td>067</td>
</tr>
<tr>
<td>Lactobacillus casei</td>
<td>-166</td>
<td>-114</td>
</tr>
<tr>
<td>Lactobacillus fermentum</td>
<td>18</td>
<td>22</td>
</tr>
<tr>
<td>Lactobacillus gasseri</td>
<td>-100</td>
<td>-81</td>
</tr>
<tr>
<td>Lactobacillus paracasei</td>
<td>222</td>
<td>36</td>
</tr>
<tr>
<td>Lactobacillus plantarum</td>
<td>-247</td>
<td>-117</td>
</tr>
<tr>
<td>Lactobacillus reuteri</td>
<td>250</td>
<td>-93</td>
</tr>
<tr>
<td>Lactobacillus rhamnosus</td>
<td>-371</td>
<td>-144</td>
</tr>
<tr>
<td>Lactobacillus salivarius</td>
<td>-236</td>
<td>-149</td>
</tr>
<tr>
<td>Saccharomyces boulardi</td>
<td>-115</td>
<td>393</td>
</tr>
<tr>
<td>Streptococcus thermophilus</td>
<td>-800</td>
<td>-108</td>
</tr>
</tbody>
</table>

Methods

Fetching PubMed and PMC search results

The entire Cavatica data processing pipeline is shown in Figure 3.1. The dotted arrows indicate steps that are not required to generate the final transition table, but can be used by users for an optional manual curation step. Shown in Figure 3.1 as examples, PubMed and PubMed Central (PMC) search results for the names of the nine network analysis tools were fetched using modified Perl scripts generated from NCBI Ebot website (39). The PubMed IDs were independently validated by completing the same search using RISmed (40). However, as far as we could find, RISmed did not fetch PMC IDs so PMC IDs were not independently verified. To further examine the downloaded search results and refine the returned IDs, Perl scripts were written to pull out sentences containing the search terms and summarize them into sentence-highlighting html files for inspection. For PubMed data, the sentences were pulled from the title and abstract of each article, but for PMC data, the sentences were pulled from the text body of an article.
Figure 3.1 Cavatica Pipeline to fetch PubMed and PubMed Central (PMC) results. The dotted arrows indicate optional computer-aided human curation steps of search results using automatically generated html files that highlight sentences containing the search keywords. The curation may result in manual removal of some search results.
The sentence-highlighting HTML file lists each article live link ID and title in a text block. Under the title is a bullet list of sentences containing the search terms in bold typeface. In this example, we have identified a false positive hit for the Gephi software that is boxed in red.

The sentence-highlighting html files improved our pipeline and filter out more false-positive search results. For example, “visant” means the particle “to” in French, thus many French articles were falsely matched to the network software VisANT until we enacted case sensitive matches to “VisANT” or “VisAnt” to exclude them from its search results (note that PubMed and PMC searches are generally case insensitive). Sometimes the query terms were only found in the references but not in the main text body of some PMC articles, which were also filtered out. The html files can also be optionally inspected by users to further improve the search results. For example, an article containing the sentence “Genetically encoded pH-indicators (GEpHIs) …” was falsely matched to the network software Gephi (see Figure 3.2) and removed by us. The number of pre- and post-filtered results from the nine network analysis software tools are listed in Table 3.6 for PubMed and PMC while the growth of publications for each tool are shown in Figure 3.6 and Figure 3.7 in supplementary materials.
Multi-network analysis using Mango Graph Studio

The PubMed and PMC parse code both took the respective XML search results as input and produced two separate tabular data summarizing the paper and author information of returned publications for each network analysis software tool. A combined co-author publication network was then created from the two tabular data files using Gel mathematics within Mango Graph Studio (Gel stands for ‘Graph Exploration Language’) (1). In each co-author publication network, nodes can represent either a publication or an author, and edges connect authors to their publications. Note that author nodes and paper nodes were not directly connected to nodes of the same types. Taking the VisANT co-author publication network as an example. The automatically generated Mango Graph Studio script for VisANT to accomplish the following steps is included below and visualized in Figure 3.3.
The network was created after its paper and author network files VisANT-papers.tsv and VisANT-authors.tsv (available with Cavatica) were imported into Mango using the import function and added together using the native Gel graph addition command (1). Subsequent Gel statements helped visualize this co-author publication network in Mango’s 3D display: papers were represented by black nodes, some authors were represented by green nodes, and authors who had produced more than one VisANT publication within the past twenty years were represented by yellow nodes. A closer examination of the yellow nodes in Mango Graph Studio reveals that this group of authors are the original VisANT developers and the papers include version releases of the VisANT software. We want to emphasize that the visualization steps are optional for generating the final author-adoption trend and users may use the interactive Mango Graph Studio visualization for further data exploration. A similar set of scripts processed the other 8 networks, and all 9 networks are visualized in Figure 3.3.
Mango allows further exploration of the data. For example, we can group similar papers and their authors together and more easily recognize key papers and key authors by identifying co-author communities within the larger networks. A co-author community is defined as a disjoint cluster (i.e., disjoint subnetwork) of more than one papers and their co-authors. Co-authors in the same community may follow a common research trajectory, thus
reading their recent papers may help us understand that trajectory. For example, a recent paper from the 3 largest co-author communities of each software is listed in the Supplementary Materials as Table 3.8 and Table 3.9, which may help readers qualitatively understand the current state-of-the-art research in each co-author community. Quantitatively, the number of distinctive communities of each software may help us understand its breadth in the whole research field and incorporating a time component may help us identify the fastest growing collaborating communities.

We may define the growth factor (GF) of each co-author community as the summation of age weighted paper count in each community to quantify their influence:

\[
GF_C = \sum_{\text{over years}} \frac{\text{paper count on a particular year}}{\text{years back from 2017}}
\]

The paper count and computed growth factor for up to the three largest communities of each software are shown in Table 3.7. Generally, larger communities have higher growth factors, which is reasonable as more papers were published in more years, pushing up the growth factor. Nevertheless, we can see that the 2 largest Pathway Studio communities have slower growth factors compared to its third largest community, suggesting that growth factor may provide an alternative measurement of the importance of each co-author community than community size. Our rationale is that older papers may not be as relevant as recent papers to indicate which community is quickly expanding. For example, papers in the Pathway Studio community that has a growth factor of 0.58 was published between 2003 to 2010, but papers in the community that has a growth factor of 2.39 was published between 2010 to 2016, which suggests that the latter community may be more active recently, despite its slightly lesser paper count.
Table 3.7 Growth factors for up to the 3 largest PubMed communities of each software.

<table>
<thead>
<tr>
<th>Network Software</th>
<th>Paper count (growth factor)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPA</td>
<td>79 (38.07) 63 (27.26) 30 (9.77)</td>
</tr>
<tr>
<td>Cytoscape</td>
<td>141 (49.62) 10 (5.42) 7 (5.33)</td>
</tr>
<tr>
<td>Pathway Studio</td>
<td>5 (2) 5 (0.58) 4 (2.39)</td>
</tr>
<tr>
<td>GraphViz</td>
<td>2 (0.21) 2 (0.21) -</td>
</tr>
<tr>
<td>VisANT</td>
<td>12 (2.64) - -</td>
</tr>
<tr>
<td>Neo4j</td>
<td>3 (3) - -</td>
</tr>
</tbody>
</table>

* Gephi, iGraph and GraphLab have no co-author communities thus they are not listed.

To study author adoption trends among the network analysis software tools, all co-author publication networks were combined using Gel commands. Cavatica generated the Mango Gel script in the following box to combine and layout these co-author publication networks. The combined PubMed co-author publication network is visualized in Fig. 4; links that cross between subnetworks are highlighted in blue. The presence of links connecting the subnetwork of one software tool to the subnetwork of another software tool indicates that some authors have used multiple tools. Since Mango Graph Studio visualization is dynamic, users can easily mouse over each node to see author or paper information, which facilitate exploration of the data, although the author adoption table can be automatically generated without any visualization.

```graphlab
graph(c_node,lt) c; // c_node type contains all data fields in 9 networks
c=GraphLab;
foreach node in c set _z=_z+9; // offset existing network
c.+=iGraph; // add another network
foreach node in c set _z=_z+9; // repeat for all networks
  c.+=Neo4j;
foreach node in c set _z=_z+9;
  c.+=VisANT;
foreach node in c set _z=_z+9;
  c.+=GraphViz;
```
foreach node in c set _z=_z+8;
c.+=Gephi;
foreach node in c set _z=_z+8;
c.+=Pathway;
foreach node in c set _z=_z+8;
c.+=Cytoscape;
foreach node in c set _z=_z+8;
c.+=Ingenuity;
foreach node in c set _z=_z-16; // center the combined network
// highlight links between the original nine subnetworks
foreach link in c where in._z!=out._z set _b=0.8,_width=0.2;

Additional Cavatica generated Gel code is shown in the following box, which exports a subgraph of authors who have published about multiple software tools and their corresponding publications. This subnetwork is a horizontal slice into the combined network shown in Figure 3.4 along the blue links. This code is interesting because it first selected the subnetwork based on the author nodes, and then annotated their paper nodes to discover the author adoption trends. This selection is possible because both authors and papers were represented in our heterogeneous co-author publication networks, and the publication year and associated software were encoded in each paper node.
Figure 3.4 Side view of merged 3D PubMed co-author publication networks. Each vertical slice is a different co-author network resulting from a different software name search term. Blue links between different subnetworks indicate authors who have used multiple tools.
Figure 3.5 highlights the author adoption trend among the network software analysis tools throughout the years. All paper nodes are labeled by their year of publication and one-letter codes of the network software they are associated with. Some papers involved more than one tool, thus are labeled with more than one code. In the blue box of Figure 3.5, we can see two example authors: one author published a Cytoscape (C) paper in 2012 and two Ingenuity Pathway Analysis (J) papers in 2014 and 2015, while his coauthor published the same Ingenuity Pathway Analysis (J) paper in 2014 but a new Cytoscape (C) paper in 2015. Therefore, based on this simple example, both Cytoscape and Ingenuity Pathway Analysis have an influx and outflux of 1 author. The entire author adoption subnetwork can be exported to a tabular file and automatically summarized by a Perl script which produced Table 3.3 in the Results section.
Figure 3.5 **Author adoption trends among software tools.** The black paper nodes are labeled with year of publication and single-letter code(s) representing the software tool(s) it is associated with. The blue box is a close-up example of two co-authors: one published a Cytoscape paper in 2012 and two IPA papers in 2014 and 2015, while the other published the same 2014 IPA paper but a new Cytoscape paper in 2015.
The PMC author adoption trends of the nine network software tools are computed similarly and listed in Table 3.4. The entire Cavatica pipeline can be run without generating visualizations. Cavatica provides default scripts to generate the final author adoption tables. Users just need to list search terms representing the methods (e.g. “microarray”, “RNA-Seq”) in the config.txt file, and type basicrun1.sh in a terminal to fetch PubMed and PMC results and generate the Mango Graph Studio Gel scripts. User can then open Mango Graph Studio and types ‘run “pubmed.gel”; run “pmc.gel”;’ into the Mango console to generate and combine all co-author publication networks. The final author adoption table is generated when users run the basicrun2.sh shell script on the Mango exported subnetwork file.

Discussion

We have developed the automatic Cavatica pipeline to fetch, filter, integrate and analyze PubMed and PubMed Central (PMC) literature search results to identify author adoption trends among different software tools or methods. Cavatica merges author and paper information into a co-author publication network, which allows individual authors to be the focus for tracing software tool or method adoption trends. Finally, Cavatica reports the author adoption trend into tables, showing the net gains of authors adopting or leaving each tool.

We validated our pipeline by using it to confirm known trends among gene expression measurement technology and gene editing technology. Then we used Cavatica to identify trends among nine network analysis software tools and among 18 different probiotic strains. We must emphasize that this automatic analysis is solely based on biomedical literature searches and does not necessarily reflect the actual popularity of each software or probiotic in the broader market.
Cavatica searches on both PubMed and Pubmed Central, even though other automatic biomedical literature search tools tend to only work with PubMed. A PMC search returns more results than a PubMed search because the search keyword is matched against the full text, not just the metadata, of each publication, and many software tools were mentioned only in the methods sections but not in the abstracts of publications. PMC data is much more complex to parse and analyze, therefore it took great efforts to make Cavatica also work with PMC. Nevertheless, PMC searches allow the identification of more detailed author adoption trends as evidenced in the nine network analysis results.

Many Cavatica steps allow visualizations. Visualizing individual co-author publication networks inspired us to create automatic Mango Graph Studio scripts to identify co-author communities and define a growth factor measurement for such communities. Since authors within the same community may have a similar research trajectory, reading the most recent papers of each community may help us select the most interesting papers to investigate further. Surveying a few larger communities, which may focus on different research subjects, may also provide a quick glance at the whole research field related to the searches. All visualizations are optional, however, and just provide users the means to further explore the data.

Challenges remain, including the fact that search keywords must be carefully chosen to better define a software or method. For example, terms like “VisANT” which unexpectedly matches to a foreign language grammar particle can significantly distort the search results. For this and other purposes, Cavatica automatically produces HTML summaries of each returned paper to help users identify false positives. The computer-aided human curation of search results can help balance false positive pruning and filtering out too
much useful information by giving very restrictive search terms, but is optional. We hope that this automatic literature analysis pipeline can be useful to researchers who wishes to conduct surveys for some research fields.

**Supplementary Materials**

Table 3.8 Sample recent PubMed papers from up to the three largest communities of each software if available.

<table>
<thead>
<tr>
<th>Network Software</th>
<th>Most Recent Paper of Community Year</th>
<th>Title</th>
<th>Papers in Year*</th>
<th>All papers in Community</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPA</td>
<td>2016</td>
<td>Altered DNA methylation in neonates born large-for-gestational-age is associated with cardiometabolic risk in children.</td>
<td>20</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Differential Amino Acid, Carbohydrate and Lipid Metabolism Perpetuations Involved in a Subtype of Rheumatoid Arthritis with Chinese Medicine Cold Pattern.</td>
<td>12</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Effects of Lithium Monotherapy for Bipolar Disorder on Gene Expression in Peripheral Lymphocytes.</td>
<td>10</td>
<td>30</td>
</tr>
<tr>
<td>Cytoscape</td>
<td>2016</td>
<td>Comprehensive gene and microRNA expression profiling reveals a role for miRNAs in the oncogenic roles of SphK1 in papillary thyroid cancer.</td>
<td>25</td>
<td>141</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Critical genes of hepatocellular carcinoma revealed by network and module analysis of RNA-seq data.</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Mechanisms of CCl4-induced liver fibrosis with combined transcriptomic and proteomic analysis.</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>Pathway Studio</td>
<td>2015</td>
<td>Bioinformatics Annotation of Human Y Chromosome-Encoded Protein Pathways and Interactions.</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2010</td>
<td>Analysis and construction of pathogenicity island regulatory pathways in Salmonella enterica serovar Typhi.</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Network analysis of human post-mortem microarrays reveals novel genes, microRNAs, and mechanistic scenarios of potential importance in fighting huntington's disease</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>GraphViz</td>
<td>2008</td>
<td>Managing an emergency department by analysing HIS medical data: a focus on elderly patient clinical pathways.</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2008</td>
<td>HBAT: a complete package for analysing strong and weak hydrogen bonds in macromolecular crystal structures.</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>VisANT</td>
<td>2016</td>
<td>Visualization of Metabolic Interaction Networks in Microbial Communities Using VisANT 5.0.</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>Neo4j</td>
<td>2016</td>
<td>Recon2Neo4j: Applying graph database technologies for managing comprehensive genome-scale networks.</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

* Number of papers in the most recent year that has publications from that community
Table 3.9 Sample recent PubMed Central papers from the three largest communities of each software if available.

<table>
<thead>
<tr>
<th>Network Software</th>
<th>Most Recent Paper of Community</th>
<th>Title</th>
<th>Papers in Year*</th>
<th>All papers in Community</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPA</td>
<td>2016</td>
<td>IPA</td>
<td>368</td>
<td>1165</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Analysis of Microarray Data on Gene Expression and Methylation to Identify Long Non-coding RNAs in Non-small Cell Lung Cancer</td>
<td>5</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Comparative miRNAome analysis revealed different miRNA expression profiles in bovine sera and exosomes</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td>Cytoscape</td>
<td>2016</td>
<td>Cytoscape</td>
<td>678</td>
<td>3219</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Modular transcriptional repertoire and MicroRNA target analyses characterize genomic dysregulation in the thymus of Down syndrome infants</td>
<td>2</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Qualitative dynamics semantics for SBGN process description</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>Pathway Studio</td>
<td>2016</td>
<td>Pathway Studio</td>
<td>2</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Distinct lymphocyte antigens 6 (Ly6) family members Ly6D, Ly6E, Ly6K and Ly6H drive tumorigenesis and clinical outcome</td>
<td>2</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Neuroproteomics and Systems Biology Approach to Identify Temporal Biomarker Changes Post Experimental Traumatic Brain Injury in Rats</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Characterizing Transcriptional Networks in Male Rainbow Darter (Etheostoma caeruleum) that Regulate Testis Development over a Complete Reproductive Cycle</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>Gephi</td>
<td>2016</td>
<td>Gephi</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Increased DNA methylation variability in type 1 diabetes across three immune effector cell types</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Research and Development of Hepatitis B Drugs: An Analysis Based on Technology Flows Measured by Patent Citations</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>GraphViz</td>
<td>2016</td>
<td>GraphViz</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Analysis of Mycobacterium tuberculosis Genotypic Lineage Distribution in Chile and Neighboring Countries</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>2013</td>
<td>CellH5: a format for data exchange in high-content screening</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2015</td>
<td>Network Reconstruction Based on Proteomic Data and Prior Knowledge of Protein Connectivity Using Graph Theory</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>VisANT</td>
<td>2016</td>
<td>VisANT</td>
<td>4</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Pathway collages: personalized multi-pathway diagrams</td>
<td>4</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Pathways and gene networks mediating the regulatory effects of cannabidiol, a nonpsychotomimetic cannabinoid, in autoimmune T cells</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Global Membrane Protein Interactome Analysis using In vivo Crosslinking and Mass Spectrometry-based Protein Correlation Profiling FN1 FN2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>Neo4j</td>
<td>2016</td>
<td>Neo4j</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>PubChem Substance and Compound databases</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>STON: exploring biological pathways using the SBGN standard and graph databases</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>2015</td>
<td>cyNeo4j: connecting Neo4j and Cytoscape</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>iGraph</td>
<td>2016</td>
<td>iGraph</td>
<td>8</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Environmental and genetic effects on tomato seed metabolic balance and its association with germination vigor</td>
<td>8</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>Ribosome quality control is a central protection mechanism for yeast exposed to deoxynivalenol and trichothecin</td>
<td>2</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>2016</td>
<td>ccmGDB: a database for cancer cell metabolism genes</td>
<td>1</td>
<td>11</td>
</tr>
</tbody>
</table>

* Number of papers in the most recent year that has publications from that community
Figure 3.6 Counts of PubMed publications by year from 1996 to 2016 for all network analysis software tools except GraphLab, whose barchart looked much like iGraph, with only two publications, one in 2013 and the other in 2014.
Figure 3.7 Counts of PubMed Central publications by year from 1996 to 2016 for all network analysis software tools except GraphLab, whose barchart only showed three publications, one in 2015 and two in 2016.
CHAPTER 4: EVALUATION AND PREDICTION OF PROBIOTICS BASED ON HUMAN MICROBIOME PROJECT DATASETS

Manuscript in preparation

Jennifer Chang and Hui-Hsien Chou

Abstract

Since the inception of the Human Microbiome Project (HMP) in 2008, there has been an increased interest in identifying and characterizing the microorganisms on different parts of the body to better understand their association with human health or diseases. The available datasets from the HMP website were generated from 18 different body sites on more than 200 healthy individuals to characterize the healthy human microbiome, and include reference sequences, microbe abundance counts, and metagenomic shotgun sequences. Combining the datasets to test biological hypotheses is a challenge. The microbiome influence on digestive trait health is most prominent, thus over-the-counter probiotic health supplements are a growing business nowadays. Although fecal transplants have been proven to be effective in treating digestive disorders like Clostridium difficile infection, eating probiotics is more acceptable by the consumers.

Probiotics is a term commonly attributed to strains believed to be beneficial to gut health. Many strains of probiotics are sold on the market today. Examining the commonly marketed 18 probiotic strains in the HMP sequence dataset can 1) provide confirmation if these commercial ‘probiotic’ strains are overly represented compared to the other microbial strains in healthy individuals, 2) suggest novel potential probiotic strains, and 3) imply their overall microbiome community function through KEGG pathway analyses.
Introduction

The Human Microbiome Project (HMP) was launched in 2008 (41), spurring research to identify and characterize the microbiome across body sites and their association with human health or diseases. The available datasets from the HMP website were generated from 18 different body site samples on more than 200 healthy individuals to characterize the healthy human microbiome. The datasets contain reference sequences, microbe abundance counts, and raw metagenomics shotgun sequences. Due to the heterogeneity and large sizes of the datasets, combining them to test biological hypotheses is a challenge. The microbiome influence on digestive trait health is most prominent, and over-the-counter probiotic health supplement business continues to grow. While fecal transplants have proven to be effective in treating digestive disorders like *Clostridium difficile* infection (42), taking probiotics is more acceptable by the consumers.

Probiotics is a term commonly attributed to strains believed to be beneficial to gut health, and the term is often attributed to R.B. Parker since his 1971 paper (43). Probiotics have been shown to be effective in treating gastrointestinal problems such as constipation in children (44) and adults (45). Many strains of probiotics are sold on the market today. Examining the commonly marketed 18 probiotic strains in the HMP sequence datasets can 1) provide confirmation if these commercial ‘probiotic’ strains are overly represented compared to the other microbial strains in healthy individuals, 2) suggest novel potential probiotic strains, and 3) reveal their microbiome community function through KEGG pathway analyses. HMP provides several datasets of bacterial genome references, 16S rRNA microbial community analysis results, and shotgun sequences.
The 16S rRNA is often used in numeric taxonomy to determine the presence and relative abundance of taxa in a sample (46). As part of the prokaryotic 30S RNA complex, the 16S rRNA has a mostly conserved sequence along with 9 hypervariable regions. Usually reads across a variable region are matched to a reference sequence and grouped into Operational Taxonomy Units (OTUs) according to sequence similarity. OTUs consists of reads that are grouped based on ~95% sequence similarity. Depending on the sequence variation within a species and the sequence similarity threshold chosen, multiple OTUs may map to one species or one OTU may include multiple species. A novel consensus sequence is generated for unmappable reads that form a novel OTU. HMP provides 16S rRNA analysis results from two programs QIIME (47) and MOTHUR (48).

For this study, we mainly focus on the QIIME dataset. The OTU tables generated by QIIME contain the abundance counts of OTUs across all samples and the corresponding taxa. However, the taxa composition of a sample is usually identified at the genus level, not the species level. The genus of the 18 commercial probiotic strains are mapped to their corresponding OTUs to determine if those OTUs are significantly abundant in GI samples than in other body location. We also check if the 18 probiotic strains were represented in the HMP gastrointestinal (GI) reference genomes. Not finding a GI reference genome for a probiotic strain is not necessarily a negative result, as it is possible that the probiotic strain has not yet been assembled. Significantly abundant OTUs in GI where none of the probiotic genus are mapped are used to suggest novel probiotic microbes. Mapping between OTU counts, taxa, and probiotic strains are done in Mango Graph Studio. A scoring scheme from body locations and OTU significances to taxa are computed using the Graph Exploration Language in Mango Graph Studio.
HMP also provides KEGG enzyme, pathway, and module representation counts from shotgun sequencing data. To characterize the community function of the 18 strains, significantly represented pathways and their corresponding enzymes in GI vs other body sites are identified by comparing the counts. From there, we determine the most abundant microbial genomes and check if the 18 probiotic strain genomes are among them. Last, we identify genes that are specific to the 18 probiotic microbes and newly identified potential probiotic strains, and list their corresponding biological function. This listing of species-specific genes suggests their unique microbiome community functions and presents a broader understanding of microbe effects on human GI health.

**HMP Datasets**

Datasets fetched from HMP website (www.hmpdacc.org) include final OTU tables for the v35 region of the 16S rRNA from QIIME analysis, final KEGG biological counts, and Genbank files of all GI bacterial species genomes. Network files were also fetched from KEGG (kegg.jp) that include linkages of organism to enzyme, enzyme to pathway, and pathway to module that allow us to map the KEGG biological counts data to the 18 probiotic strains. From the HMP datasets, we derived several different biological networks, merged them and analyzed them within Mango Graph Studio.

We focused on merging 3 types of datasets: 1) the 16S OTU counts to identify significantly different microbe abundances and their corresponding correlation networks in GI vs other body locations to identify other potential probiotics; 2) the KEGG counts derived from shotgun sequencing to identify significantly represented pathways in the GI vs other body location and any connections to the 18 probiotics; and 3) the HMP references to determine if the 18 commercial probiotics are present in HMP GI references. The biological
network files created on the fly were loaded and merged within Mango Graph Studio to provide a multi-layered approach of exploring the HMP datasets from the context of the 18 probiotic strains. The pipelines are shown in Figure 4.1.

**Figure 4.1 HMP data analysis pipeline.** HMP provided datasets are boxed in blue. The various heterogeneous networks are combined in Mango Graph Studio to answer biological questions.

### Mapping the 18 probiotic strains to significantly different OTUs

As stated earlier, OTUs within the QIIME microbe abundance counts are mapped to microbial taxa. An examination of the data revealed that the OTUs are only mapped to genus level taxa. Therefore, 4 out of 5 genus names of the 18 probiotics were matched to their corresponding taxonomy names in the HMP dataset. Note that multiple OTUs can map to the same taxa. The 5 probiotic genus names, their corresponding QIIME taxa names and the counts of mapped OTUs are listed in Table 4.1.
Table 4.1 The probiotic genes and their corresponding microbe names in the HMP microbe counts

<table>
<thead>
<tr>
<th>Probiotic genus</th>
<th>QIIME microbe name</th>
<th># of OTU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bacillus</td>
<td>Root;p__Firmicutes;c__Bacilli;o__Bacillales;f__Bacillaceae;g__Bacillus</td>
<td>23</td>
</tr>
<tr>
<td>Bifidobacterium</td>
<td>Root;p__Actinobacteria;c__Actinobacteria;o__Bifidobacteriales;f__Bifidobacteriaceae;g__Bifidobacterium</td>
<td>39</td>
</tr>
<tr>
<td>Saccharomyces</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>Lactobacillus</td>
<td>Root;p__Firmicutes;c__Bacilli;o__Lactobacillales;f__Lactobacillaceae;g__Lactobacillus</td>
<td>2937</td>
</tr>
<tr>
<td>Streptococcus</td>
<td>Root;p__Firmicutes;c__Bacilli;o__Lactobacillales;f__Streptococcaceae;g__Streptococcus</td>
<td>3562</td>
</tr>
</tbody>
</table>

The HMP made it a goal to sequence the microbe genomes in the gut and other bodily locations to provide standard datasets for human microbiome research. As of 2017, there are 457 microbial strains that are sequenced, assembled and made available on the HMP website. The 18 probiotic strain names were matched against the HMP gastrointestinal (GI) reference GenBank files. Of the 18 probiotic strains, 10 are mapped to at least one of the 457 microbial strains. Some strains were not mapped to any reference (e.g. *Bacillus subtilis*, *Bifidobacterium bifidum* and *Bifidobacterium lactis*), while other probiotic strains were matched to multiple reference sequence (e.g. *Lactobacillus reuteri*, etc.). Table 4.2 below lists the 18 probiotic strains and their corresponding GI reference names in the HMP dataset if available.

The QIIME generated OTU tables contain the abundance counts of 4856 samples from 235 individuals (124 male, 111 female) collected at 18 different body sites at up to 3 different timepoints. A summary of the number of samples by body location in the QIIME dataset is included in Table 4.3 below.
### Table 4.2 The 18 probiotics and their corresponding organism name in the HMP reference

<table>
<thead>
<tr>
<th>Probiotic</th>
<th>HMP Gastrointestinal Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bacillus subtilis</td>
<td>N/A</td>
</tr>
<tr>
<td>Bifidobacterium bifidum</td>
<td>Bifidobacterium bifidum NCIMB 41171</td>
</tr>
<tr>
<td>Bifidobacterium breve</td>
<td>Bifidobacterium breve DSM 20213 = JCM 1192</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium breve HPH0326</td>
</tr>
<tr>
<td>Bifidobacterium infantis</td>
<td>N/A</td>
</tr>
<tr>
<td>Bifidobacterium lactis</td>
<td>N/A</td>
</tr>
<tr>
<td>Bifidobacterium longum</td>
<td>Bifidobacterium longum subsp. infantis 157F</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. infantis ATCC 15697 = JCM 1222 = DSM</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. infantis CCUG 52486</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. longum 2-2B</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. longum 44B</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. longum ATCC 55813</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. longum F8</td>
</tr>
<tr>
<td></td>
<td>Bifidobacterium longum subsp. longum JCM 1217</td>
</tr>
<tr>
<td>Lactobacillus acidophilus</td>
<td>Lactobacillus acidophilus ATCC 4796</td>
</tr>
<tr>
<td>Lactobacillus brevis</td>
<td>Lactobacillus brevis subsp. gravesensis ATCC 27305</td>
</tr>
<tr>
<td>Lactobacillus casei</td>
<td>N/A</td>
</tr>
<tr>
<td>Lactobacillus gasseri</td>
<td>N/A</td>
</tr>
<tr>
<td>Lactobacillus paracasei</td>
<td>Lactobacillus paracasei subsp. paracasei ATCC 25302</td>
</tr>
<tr>
<td>Lactobacillus plantarum</td>
<td>Lactobacillus plantarum subsp. plantarum ATCC 14917 = JCM 1149 =</td>
</tr>
<tr>
<td>Lactobacillus reuteri</td>
<td>Lactobacillus reuteri CF48-3A</td>
</tr>
<tr>
<td></td>
<td>Lactobacillus reuteri JCM 1112</td>
</tr>
<tr>
<td></td>
<td>Lactobacillus reuteri MM2-3</td>
</tr>
<tr>
<td></td>
<td>Lactobacillus reuteri MM4-1A</td>
</tr>
<tr>
<td></td>
<td>Lactobacillus reuteri SD2112</td>
</tr>
<tr>
<td>Lactobacillus rhamnosus</td>
<td>Lactobacillus rhamnosus ATCC 21052</td>
</tr>
<tr>
<td></td>
<td>Lactobacillus rhamnosus GG</td>
</tr>
<tr>
<td></td>
<td>Lactobacillus rhamnosus LMS2-1</td>
</tr>
<tr>
<td>Lactobacillus salivarius</td>
<td>N/A</td>
</tr>
<tr>
<td>Saccharomyces boulardii</td>
<td>N/A</td>
</tr>
<tr>
<td>Streptococcus thermophilis</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### Table 4.3 Summary of the v35 Microbe samples by body site

<table>
<thead>
<tr>
<th>Samples</th>
<th>Location</th>
<th>GI</th>
<th>Skin</th>
<th>Mouth</th>
<th>Vaginal</th>
</tr>
</thead>
<tbody>
<tr>
<td>282</td>
<td>Anterior nares</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>313</td>
<td>Attached keratinized gingiva</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>312</td>
<td>Buccal mucosa</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>308</td>
<td>Hard palate</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>225</td>
<td>Left antecubital fossa</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>Left retroauricular crease</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>134</td>
<td>Mid vagina</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>315</td>
<td>Palatine Tonsils</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>133</td>
<td>Posterior fornix</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>241</td>
<td>Right antecubital fossa</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>305</td>
<td>Right retroauricular crease</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>292</td>
<td>Saliva</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>321</td>
<td>Stool</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>309</td>
<td>Subgingival plaque</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>314</td>
<td>Supragingival plaque</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>310</td>
<td>Throat</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>316</td>
<td>Tongue dorsum</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>127</td>
<td>Vaginal introitus</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
From the OTU abundance datasets, samples containing less than 1000 reads were filtered out and the remaining OTU counts were normalized by samples, and then the samples were analyzed by one-way ANOVA followed by Tukey HSD to determine if OTUs between body sites were significant and if so which OTUs were significantly abundant when comparing Stool (GI) with other body locations. Out of a total of 45,383 OTUs, 9349 were more abundant in GI than other body locations. The number of OTUs that were more prevalent in GI sample than other body sites are listed in Table 4.4.

<table>
<thead>
<tr>
<th>Body Site</th>
<th>16S rRNA OTU</th>
<th>Enzyme</th>
<th>KEGG Pathway</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anterior nares</td>
<td>9057</td>
<td>2288</td>
<td>67</td>
<td>57</td>
</tr>
<tr>
<td>Attached Keratinized gingiva</td>
<td>9225</td>
<td>1213</td>
<td>38</td>
<td>34</td>
</tr>
<tr>
<td>Buccal mucosa</td>
<td>9216</td>
<td>2802</td>
<td>77</td>
<td>60</td>
</tr>
<tr>
<td>Hard Palate</td>
<td>9189</td>
<td>468</td>
<td>15</td>
<td>16</td>
</tr>
<tr>
<td>Left Antecubital fossa</td>
<td>8514</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Left Retroauricular crease</td>
<td>9131</td>
<td>1418</td>
<td>44</td>
<td>41</td>
</tr>
<tr>
<td>Mid vagina</td>
<td>8162</td>
<td>903</td>
<td>38</td>
<td>30</td>
</tr>
<tr>
<td>Palatine tonsils</td>
<td>9216</td>
<td>975</td>
<td>29</td>
<td>32</td>
</tr>
<tr>
<td>Posterior fornix</td>
<td>8167</td>
<td>2740</td>
<td>92</td>
<td>65</td>
</tr>
<tr>
<td>Right antecubital fossa</td>
<td>8643</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Right Retroauricular crease</td>
<td>9115</td>
<td>1681</td>
<td>50</td>
<td>41</td>
</tr>
<tr>
<td>Saliva</td>
<td>9142</td>
<td>950</td>
<td>32</td>
<td>28</td>
</tr>
<tr>
<td>Supragingival plaque</td>
<td>9198</td>
<td>950</td>
<td>31</td>
<td>32</td>
</tr>
<tr>
<td>Supragingival plaque</td>
<td>9222</td>
<td>2105</td>
<td>51</td>
<td>54</td>
</tr>
<tr>
<td>Throat</td>
<td>9192</td>
<td>1081</td>
<td>33</td>
<td>32</td>
</tr>
<tr>
<td>Tongue dorsum</td>
<td>9233</td>
<td>2455</td>
<td>67</td>
<td>63</td>
</tr>
<tr>
<td>Vaginal introitus</td>
<td>8030</td>
<td>1056</td>
<td>44</td>
<td>36</td>
</tr>
<tr>
<td><strong>Unique Sig. Diff. at 95% CI</strong></td>
<td><strong>9349</strong></td>
<td><strong>3525</strong></td>
<td><strong>123</strong></td>
<td><strong>80</strong></td>
</tr>
<tr>
<td><strong>Total Possible</strong></td>
<td><strong>45,383</strong></td>
<td><strong>9660</strong></td>
<td><strong>297</strong></td>
<td><strong>250</strong></td>
</tr>
</tbody>
</table>

Spearman Correlation links 25,619,186*

* Only the correlation links between significant OTUs were created and counted.

A network was created connecting body sites to OTU sequences that were more prevalent in stool (GI) then themselves using Mango Graph Studio analysis. As an example, from Table 4.4, 9142 OTUs were significantly more abundant in GI vs Saliva. Therefore, in the body-to-OTU network, there are 9142 links connecting the node "Saliva" to the 9142 individual OTU nodes that were more abundant in GI than in saliva. The links were weighted
by the computed sample mean differences and p-values. The weighted links allow flexible thresholding of significantly different OTUs and their corresponding taxa within the Mango Graph Studio. Additionally, a OTU spearman correlation network was built across samples for only the 9349 significantly abundant OTUs using the Hmisc R package (49) and an OTU-to-taxa network was built to map OTUs to probiotic genus.

The HMP provided KEGG pathway, enzyme and module representation counts that were computed from shotgun sequencing data. The HMP KEGG dataset contains 686 samples from 103 individuals (56 male, 46 female) sampled at up to 16 body locations during up to 3 timepoints. The number of samples separated by body locations is listed in Table 4.5.

### Table 4.5 Summary of the HMP KEGG pathway data by body location

<table>
<thead>
<tr>
<th>Samples</th>
<th>Location</th>
<th>GI</th>
<th>Skin</th>
<th>Mouth</th>
<th>Vaginal</th>
</tr>
</thead>
<tbody>
<tr>
<td>87</td>
<td>Anterior nares</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>109</td>
<td>Buccal mucosa</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Hard palate</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Keratinized_gingiva</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>L Retroauricular crease</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Mid vagina</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>6</td>
<td>Palatine tonsils</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>Posterior fornix</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>17</td>
<td>R Retroauricular crease</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>5</td>
<td>Saliva</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>136</td>
<td>Stool</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>7</td>
<td>Subgingival plaque</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>7</td>
<td>Throat</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>123</td>
<td>Tongue dorsum</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>Vaginal introitus</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The HMP KEGG datasets were already normalized. One-way ANOVA followed by Tukey HSD were used to identify significantly represented enzymes, pathways and modules in GI vs other body locations. The number of resulting enzymes, pathways and modules that were significantly more represented in stool vs other body site samples are listed in Table 4.4. Spearman correlation networks were built using the Hmisc R package for KEGG
pathways and modules. The correlation networks were built across pathways and modules.
The number of correlation links are listed along the bottom of Table 4.4.

**Merging heterogeneous graphs in Mango Graph Studio**

Mango Graph Studio enables the analysis of multiple heterogeneous networks via the Graph Exploration Language (Gel) and a 3D interactive visualization environment. A full discussion of Mango features and applications examples are available on GitBook ([https://www.gitbook.com/book/j23414/mango-user-guide](https://www.gitbook.com/book/j23414/mango-user-guide)). The networks built in the last step were loaded and merged in Mango via the following Gel script. Because nodes in those networks may represent a body site, an OTU sequence, a Taxa or a probiotic strain, a layered or segmented network visualization might work best for their analysis. Conceptually, from left to right, we want segregated layers of nodes that representing body sites, significantly abundant OTUs, taxa and non-significant OTUs with horizontal links among these layers.

Since many networks may be generated and updated, it is more efficient to store network names as strings at the top of a Gel script. We declared a *node type* that contains an extra string attribute *type* in addition to their string *id* to indicate the different types of nodes "body", "otu", "taxa", etc. Since the body-site-to-OTU network contains a mean difference and a p-value per each link, its *link type* is directional (<>) and contains the float *diff* and float *p_val* attributes. The following Gel script loads the three networks into Mango: body-site-to-OTU network where links are weighted by mean differences and p-values, OTU-to-taxa network where multiple OTUs are mapped to each taxa, and a network of OTUs and taxa nodes that are related to the 18 probiotic strains. Note that the taxa nodes list genus level but not strain level information. The resulting networks are shown in Figure 4.2.
string ofile = "./.Rscripts/otu.net";          // body to OTUs, sig diff
string tfile = "./.DATA/QIIME/otu_taxa.tsv";   // maps OTU to taxa
string pfile = "./.DATA/QIIME/probiotic.net"; // probiotic OTUs and Taxa

node(string id, string type) nt;       // node type
link<float diff, float p_val> lt; // link type

/* Body-site-to-OTU network */
graph(nt,lt) g=import(ofile," ",1);
foreach link in g set in.type="body", out.type="otu";
foreach node in g where type=="body" set _x=-5, _y=rand(-8,8), _text=id, _g=0.5;
foreach node in g where type=="otu" set _x=1, _y=rand(-8,8), _r=0.5;
foreach link in g set _width=0.1, _g=(0.05-p_val)*20;

/*OUT-to-Taxa (genus) */
link<> lt2;   // this network needs no link attributes
graph(nt,lt2) g2=import(tfile,\"t\",1);
foreach link in g2 set in.type="otu", out.type="taxa";
foreach link in g2 set in._x=8, in._y=rand(-8,8), out._x=5, out._y=in._y;
foreach link in g2 set _width=0.1;
foreach link in g2 set in._y=out._y;
foreach node in g2 where type=="taxa" set _b=0.5;

/*Probiotic Taxa and OTUs */
graph(nt,lt2) g3=import(pfile,\" ",1);
foreach node in g3 set _radius=0.5;

/*Merge all three networks into sum, saving their attributes*/
auto sum = g .+ g2; // adds the taxa to OTU graph and save to new graph sum
sum.+g3 // increases the radii of probiotic OTU and taxa
Figure 4.2 The three biological networks generated from the HMP datasets. In graph g, the body site is listed on the left and are linked to OTUs on the right that were significantly more abundant in GI. In graph g2, blue nodes representing taxa are linked to their corresponding OTUs in black. In graph g3, all probiotic related taxa and OTUs are left at default xyz position (0,0,0) but with a larger non-default radius. There were 9349 significantly abundant OTUs out of a total of 45383 in GI versus other body locations.

The three graphs in Figure 4.2 are slightly different from one another in purpose and presentation. The first graph (g) provides a summary of OTUs that are significantly more abundant in GI than other body sites. The second graph (g2) provides a mapping from taxa (blue nodes) to OTUs (black nodes). If many OTUs were mapped to the same taxa, they were plotted in the same location for clarity, meaning that some OTUs overlap in this network. In the same vein, the third graph (g3) looks like one single node but it contains many taxa nodes as well as OTU nodes. All the nodes in the third graph were plotted at the default xyz position (0,0,0) but their radius was changed to a larger non-default value.
Mango Graph Studio contains the Graph Exploration Language with standardized graph mathematics that can take care of merging non-default attributes and resolving data conflicts automatically. Given two sample graphs $G_A$ and $G_B$, Mango Graph Studio provides graph mathematics like the following. Therefore, to merge the three networks we used the dot additions listed in the previous script that result in the graphs shown in Figure 4.3.

$G_A = \{V_A, E_A\}$

$G_B = \{V_B, E_B\}$

$G_A + G_B = \{V_A \cup V_B, E_A \cup E_B\}$

$G_A + G_B = \{V_A, E_A \cup \{(v_i, v_j) | v_i, v_j \in V_A, (v_i, v_j) \in E_B \}\}$

$G_A - G_B = \{V_A \setminus V_B, (v_i, v_j) | (v_i, v_j) \in E_A \setminus E_B, v_i, v_j \in [V_A \setminus V_B] \}$

$G_A - G_B = \{V_A, E_A \setminus E_B\}$
In Figure 4.3, the graph on the left is the result of \( g + g_2 \) where OTUs that are significantly different in GI are shown in the red column while non-significant OTUs are shown in black on the right. Taxa nodes are represented by blue nodes. In the original graph \( g \), the significant OTUs have an x position of 1 which should have conflicted with the x position of 7 in \( g_2 \). However, when the taxa to OTU network is added in, the significantly different OTUs remain to the left (in red) while the non-significantly different OTU's remain on the right (in black). This is left operand preference and we shall take advantage of this property when merging and comparing two graphs. The graph on the right of Figure 4.3 is the result of adding in the probiotic network (\( g_3 \)). Since node radius values in the sum network were at default value of 0.2, the non-default radius values of 0.5 overwrote the default radius values, highlighting the probiotic relevant parts of the network. Additionally,
Gel allows us to count how many probiotic OTUs are significantly different and not significantly different based on radius and layer.

```csharp
/* count occurrence of significant probiotic OTUs and non sig probiotic OTUs */
int i=0; foreach node in sum where _radius>0.2 && _x==1 set i++; print i;
int j=0; foreach node in sum where _radius>0.2 && _x==8 set j++; print j;
```

After loading into Mango Graph Studio, we could determine that 11 out of 6561 OTUs that mapped to probiotic genus were significantly different. Those 11 OTUs mapped to *Lactobacillus* and *Bifidobacterium* which are two out of the four probiotic genera.

Viewing layered networks enable the user to see connection between body site, significant OTUs, taxa and insignificant OTUs. Mango Graph Studio enables propagation of values between those levels so we can program various criteria for identifying significant taxa. For example, we could define a probiotic score for a genus summing the number of significant OTUs mapping to that genus, normalize by total number of OTUs for that genus.

The equation and corresponding Gel script is shown below:

\[
\text{ProbioticScore} = \frac{\# \text{ of sig. OTU's for Genus}}{\# \text{ of total OTU's for Genus}}
\]

```csharp
// Number of significant OTUs for a taxa (genus)
foreach link in sum where out.type=="taxa" && in._x<out._x set out._text++;
// Normalized by total number of OTUs for a taxa (genus)
float f; foreach node in sum where type=="taxa" set f=_text, _text=f/(in+out);
// Export Mango computed values for further analysis
auto out=select node from sum where type=="taxa" && _text>0.3;
foreach node in out set _text=_text."":".id;
export ("out.tsv","tsv",out);
```
The tabular output can then be sorted to identify novel probiotics based on the scoring function and is listed below. Other scoring functions tested by changing the Gel script.

Layering the network structure can help inspire new scoring functions that can be propagated layer to layer. New layers can be added and organized individually. For example, the network in Figure 4.4 has been reorganized to group similar body sites together and where taxa have been reordered such that related species are placed near each other. Certain regions of the taxa tree seem to be more prevalent in Stool vs other body location and can be viewed as the clustered parallel lines between the red OTU nodes and the blue taxa nodes. The reordering by location also allows subnetworks to be fetched and shown in Figure 4.5.
Figure 4.4 Regrouping of layers in the graph. The body sites are grouped into Mouth, Skin, and Vaginal. The taxa are rearranged to follow the taxonomy structure.

Figure 4.5 Subnetworks by location
Mapping the 18 probiotic strains to significantly different KEGG pathways

Organisms listed on KEGG have short letter codes (org IDs) that are required to fetch corresponding organism to enzyme or organism to pathway data. Of the 18 probiotics, 15 matched to at least one KEGG org ID. We wrote a script that fetched all orgs associated with listed strains and then pools the KEGG enzymes available for that strain. The number of KEGG org IDs matching to each of the 18 probiotics and the number of unique enzymes associated with that organism are listed in Table 4.6.

<table>
<thead>
<tr>
<th>Probiotic</th>
<th>KEGG org ID match</th>
<th>Unique Enzymes with KO KEGG ids</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bacillus subtilis</td>
<td>15</td>
<td>2063</td>
</tr>
<tr>
<td>Bifidobacterium bifidum</td>
<td>3</td>
<td>835</td>
</tr>
<tr>
<td>Bifidobacterium breve</td>
<td>9</td>
<td>960</td>
</tr>
<tr>
<td>Bifidobacterium infantis</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bifidobacterium lactis</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Bifidobacterium longum</td>
<td>12</td>
<td>1050</td>
</tr>
<tr>
<td>Lactobacillus acidophilus</td>
<td>4</td>
<td>928</td>
</tr>
<tr>
<td>Lactobacillus brevis</td>
<td>2</td>
<td>1003</td>
</tr>
<tr>
<td>Lactobacillus casei</td>
<td>7</td>
<td>1200</td>
</tr>
<tr>
<td>Lactobacillus gasseri</td>
<td>1</td>
<td>798</td>
</tr>
<tr>
<td>Lactobacillus paracasei</td>
<td>4</td>
<td>1202</td>
</tr>
<tr>
<td>Lactobacillus plantarum</td>
<td>7</td>
<td>1244</td>
</tr>
<tr>
<td>Lactobacillus reuteri</td>
<td>5</td>
<td>1010</td>
</tr>
<tr>
<td>Lactobacillus rhamnosus</td>
<td>6</td>
<td>1190</td>
</tr>
<tr>
<td>Lactobacillus salivarius</td>
<td>3</td>
<td>990</td>
</tr>
<tr>
<td>Saccharomyces boulardii</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Streptococcus thermophiles</td>
<td>8</td>
<td>1036</td>
</tr>
</tbody>
</table>

The new networks were combined into the layered network creating the "Probiotic" and "Enzyme" columns in Figure 4.6. Several layers were combined and values can be propagated from any layer to another layer using the foreach loops.
Figure 4.6 Layered network of several different biological entities grouped by column

Layering the network in a hierarchical structure also allows propagation of values or computed scores along the network from one layer to another.

Figure 4.7 Network analysis of layered biological entities.

To keep track of the heterogeneous biological entities, we segregated entities into layers grouped by similar type. In Figure 4.7, the left to right layers contains general body location, specific body location, significant OTUs, Taxa, non-significant OTUs, probiotic
strains, enzymes, and KEGG pathways. This eases the mental burden of looking at a complicated graph with multiple biological entity types. Certain layers have their own ordering from top to bottom depending on their type. For example, the specific body locations are grouped by their general body location. The Taxa are organized so related taxa are grouped together. Each layer can be rearranged independently.

Visualizing layered networks separated by different biological entities may inspire users to test new questions. Since values can be propagated along links in Mango Graph Studio, new scoring functions can be tested that cross different biological entities. For example, in Figure 4.7, the significant difference among OTUs are summed to the right and stored in the taxa nodes. The taxa node sums are then summed to the right to their probiotics nodes, then their enzymes, then finally KEGG pathway nodes. The final column of KEGG pathway nodes is isolated into a subnetwork and to be viewed in a separate panel where computed values are mapped to height. Mango provides an integrated environment to layer complex networks, propagate values through complex network, and provide basic plotting to inspire and test biological questions.

**Identify potential strain specific functions in a community**

To preliminary determine the unique function of a microbe in a community we used the microbe to enzyme network to identify enzymes and their corresponding pathways that were unique to one or two microbes and not to any other microbe in the group. For a preliminary study, we looked at the 18 probiotic strains. From the microbe to enzyme network, we removed any enzymes that were connected to more than two of the 18 probiotics. The resulting network with enzymes labeled by number of microbes connected to it is shown in Figure 4.8.
Figure 4.8 Probiotic to enzyme network. The enzymes are labeled by number of probiotic strains connected to it.

Assuming there is a group of bacteria in a niche and the entire community is working together, enzymes unique to a microbial strain may suggest the strains function within a group. For example, if bacteria A is the only strain capable of converting an otherwise toxic environmental substance to a beneficial substance, then bacteria A is necessary for the entire community to thrive despite the environmental toxin. To identify the unique functions of a community containing the 18 probiotic strains, we used Mango Graph Studio to isolate a subgraph containing only enzymes that are present in up to two bacteria strains. The subgraph is shown in Figure 4.8.
Discussion

More varied data are becoming available from endeavors such as the Human Microbiome Project and many challenges remain. In recent years, there has been an increased interest in identifying, and characterizing the microorganisms on different parts of the body to better understand their association with human health or disease and seems to be a valuable resource for validation or invalidation of popular probiotics. Examining the commonly marketed 18 probiotic strains in the HMP sequence dataset presents a challenge in that the microbe counts are only listed down to a genus level which may or may not provide enough resolution in the final microbe count to confirm or refute the use of these 18 probiotics. In the same way, suggesting novel probiotic strains may be premature until resolution can go further than genus level.

However, if a set of microbes are known to inhabit a community, a KEGG pathway analysis may suggest their unique role in that community. If a small subset of microbes has the genetic material for certain enzymes, that could suggest their needed role in a community. These pathways or modules that are represented a small sample of the community may suggest their overall microbiome community function.
CHAPTER 5 : DISCUSSIONS AND CONCLUSIONS

We have developed a new software platform for multi-network analysis and visualization: Mango Graph Studio enables scientists to test hypotheses on large heterogeneous networks, to identify crucial features, and to extract analysis results all within its integrated analysis and visualization environment. Users may flexibly define node and link attributes and their associations using its Graph Exploration Language (Gel). Mango extends the capability and convenience of large heterogeneous data analysis beyond existing programs; Mango can load networks with millions of links, allow users to integrate and explore large amounts of data with Gel commands, and help users deduce predictions or outcomes that can be validated in labs. We hope Mango Graph Studio will become an essential tool for future biological research.

We used Mango Graph Studio to build a pipeline called Cavatica to fetch, filter, integrate and analyze PubMed and PubMed Central (PMC) literature search results to identify author adoption trends among different software tools or methods. Cavatica merges author and paper information into a co-author publication network, which allows individual authors to be the focus for tracing software tool or method adoption trends. We validated our pipeline by using it to confirm known trends among gene expression measurement technology and gene editing technology, and then we used Cavatica to identify novel trends among nine network analysis software tools and among 18 different probiotic strains.

Many other automatic biomedical literature search tools only work with PubMed searches, but Cavatica works with both PubMed and PubMed Central (PMC) searches. A PMC search returns more results than a PubMed search because the search keyword is
matched against the full text, not just the metadata, of each publication. PMC data is much more complex to parse and analyze, therefore it took great efforts to make Cavatica also works with PMC. Nevertheless, PMC searches allow the identification of more detailed author adoption trends as evidenced in the nine network analysis results.

Finally, we used Mango Graph Studio to integrate differing types of biological datasets from the Human Microbiome Project (HMP) and integrated them into hierarchical networks to evaluate the prevalence of 18 probiotic strains available on the consumer market. The hierarchical layers of the datasets included body sites, taxa, OTUs, KEGG enzymes, KEGG pathways, and correlation networks. From the layered networks, we could subset the networks based on any of the levels or interactions between the levels. Here we benefited from having the Gel graph mathematics that merges non-default values from node and links and manages data conflict automatically. We did not find that the HMP datasets provide sufficient detail to support the efficacy of the 18 commercially available probiotic strains, although we do suggest some novel probiotic strains based on our scoring criteria.

**Recommendations for Future Research**

The Mango Graph Studio software has been commercially distributed since 2016 and is continuously being developed and maintained by a company. Many user features requests are being added to Mango, e.g., to enable more import-export options including visual media formats, to add hashes and arrays to the Gel to enable complex analysis algorithms to be conveniently implemented, and to implement Graphics Processing Unit (GPU) accelerations. Although we have found many applications of Mango features, we have not explored them all. For example, the simulation capabilities of Mango Graph Studio such as allowing the propagation of functions through networks have yet to be explored for biological studies.
Preliminary forays into cellular automata simulation using Mango have been written and documented on GitHub, proving that Mango Graph Studio can also be used for simulation, but more is possible. Despite the availability of Mango Graph Studio, there are still challenges in multi-network analysis as more diverse datasets continue to be created, which may influence the future direction of Mango development.
REFERENCES


32. Csárdi G, Nepusz T. The igraph software package for complex network research.


40. Kovalchik S. R: Download content from NCBI databases. [Internet]. 2016 [cited 2017 Jan 27]. Available from: https://cran.r-project.org/package=RISmed


49. CRAN - Package Hmisc [Internet]. [cited 2017 Jun 19]. Available from: https://cran.r-project.org/web/packages/Hmisc/index.html