

Visual Annotation of Gene List with Functional Enrichment

BY

JING WEN

B.E., Zhejiang University, 2006

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Bioinformatics
in the Graduate College of the
University of Illinois at Chicago, 2011

Chicago, Illinois

Defense Committee:

Hui Lu, Advisor, Chair
Jie Liang
Yang Dai

ACKNOWLEDGMENTS

First and foremost, I would like to express my most sincere thanks my advisor Dr. Hui Lu for his thoughtful guidance through the past three years. His creative ideas, broad knowledge and great passion to research inspire me all the time. It is my honor to be a student of him and learn from him. Thanks.

I would also like to thank my other two committee members Dr. Jie Liang and Dr. Yang Dai. Thank you for offering great lessons from which I have learned a lot of Bioinformatics.

Thanks to all lab mates, Morten Kallberg, Robert E. Langlois, Matthew B. Carson, Georgi Genchev and Wenyi Qin. You are all nice and talented. I had a great time working and studying with you guys.

Special thanks to Dr. Simon Lin who has given me lots of helpful suggestions for my thesis project.

Thanks to my wonderful husband Xishu for supporting me all the time.

TABLE OF CONTENTS

<u>CHAPTER</u>	<u>PAGE</u>
1 INTRODUCTION	1
1.1 Introduction	1
1.2 Organization	2
2 BACKGROUND	4
2.1 Biomedical literature resources	4
2.1.1 Gene Ontology	4
2.1.2 GeneRIF	9
2.2 Knowledge management and information extraction	10
2.2.1 Text mining	12
2.2.2 Text based functional enrichment analysis	14
2.2.3 Hypergeometric Distribution	18
2.3 Web based information visualization	21
2.3.1 Word cloud	21
2.3.2 ASP.NET and PHP	23
3 GENEGIF	27
3.1 Introduction	27
3.2 System and methods	29
3.2.1 Data set	29
3.2.2 Text mining	29
3.2.2.1 Stopwords removing	29
3.2.2.2 Morphological unification	32
3.2.2.3 Phrase recognition	33
3.2.3 Generation of graph	33
3.2.4 Advanced web features	35
3.3 Result and conclusion	40
4 LISTGIF	44
4.1 Introduction	44
4.2 Methods	46
4.2.1 Hypergeometric distribution for enrichment analysis	46
4.2.2 Graphic visualization and web access	49
4.3 Results	50
4.4 Discussion and conclusion	51
CITED LITERATURE	57

TABLE OF CONTENTS (Continued)

<u>CHAPTER</u>	<u>PAGE</u>
VITA	62

LIST OF TABLES

<u>TABLE</u>		<u>PAGE</u>
I	EXAMPLE OF OVERREPRESENTED FUNCTIONS FOR A GROUP OF GENES	15
II	BIOMEDICAL LITERATURE RESOURCES USED BY GENE FUNCTIONAL ENRICHMENT TOOLS	17
III	HYPERGEOMETRIC EXPERIMENT WITH BALL MODEL . . .	20
IV	COMPARISION OF ASP.NET AND PHP	26
V	STOPWORD LIST IN GENEGIF	31
VI	MORPHOLOGICAL RULES USED TO DERIVE BASE FORM OF A WORD	32
VII	PROBABILITY FROM HYPERGEOMETRIC DISTRIBUTION FOR OVER REPRESNETED TERMS IN 50-GENE LIST	52
VIII	PROBABILITY FROM HYPERGEOMETRIC DISTRIBUTION FOR OVER REPRESNETED TERMS IN BREAST CANCER GENE LIST	53

LIST OF FIGURES

<u>FIGURE</u>		<u>PAGE</u>
1	Example of GO term	5
2	Go term in tree view vs. GO term in graph view	7
3	Example of gene annotated by GO term	8
4	GeneRIF of gene "RENBP"	9
5	Number of GeneRIF for different organisms	11
6	Types of output from text mining process	13
7	Most searched topcis in NY Times displayed in tag cloud	22
8	Word cloud for gene "KLF4" using Wordle	24
9	Number of genes vs. number of GeneRIFs	30
10	Data flow for mophological unification	34
11	The GeneGIF of gene "KLF4" and the related GeneRIFs when clicking a specific word. The words related to the important gene functions like "Cell Cycle", "Cancer" etc. stand out in the graph.	36
12	Some web features for better user interaction with GeneGIF	38
13	Discussion board for GeneGIF graph with prepopulated RefSeq records	39
14	GeneGIF system architecture	41
15	Four tiers of information levels in GeneGIF	42
16	ListGIF graph when clicking a specific term showing raw GeneRIF and genes in the input list that present with this term.	50

LIST OF ABBREVIATIONS

ASP	Active Server Pages
DAG	Directed Acyclic Diagram
IIS	Internet Information Services
KEGG	Kyoto Encyclopedia of Genes and Genomes
GeneRIF	Gene Reference Into Function
GO	Gene Ontology
NCBI	National Center for Biotechnology Information
PHP	Hypertext Preprocessor
RefSeq	NCBI Reference Sequences

SUMMARY

The genes in NCBI databases are currently annotated with itemized text (Gene Reference Into Function, or GeneRIF). A previous work suggests that the visual presentation can be more effective when time and space are under heavy constraints. In this thesis we first report a novel annotation of the genome information using Web 2.0 technologies: GeneGIF (Gene Graphics Into Function). The users can quickly scan through important functions of each gene from a graph, and then go to detailed pages when they find interesting annotations. The modular implementation makes it easily pluggable into other widely used databases without reprogramming. Then we present another web based tool - ListGIF which derives over represented concepts for a list of genes from biomedical literature. ListGIF supports two literature resources: GeneRIF and Gene Ontology. Our strategy is based on the idea that the significance of a feature is associated with the number of literature co-occurrences among each gene's annotation in the list. To reduce the bias that unbalanced GeneRIF distribution among genes might bring to the result, we provide both gene level and GeneRIF level analysis. Result is also presented in a word-cloud like graph with traceability to original published evidence.

CHAPTER 1

INTRODUCTION

1.1 Introduction

Because of the nature of biology and biomedical fields, the amount of information generated from experiments and other computational methods is enormous. Information accumulated is a lot more than it has been interpreted. Traditional manual review for discovering important knowledge from biomedical literature is not sufficient in modern scientific researches. For example in PubMed (1) database, there are 21 million citations for biomedical literature from MEDLINE. And it is expanding at the rate of 500,000 new citations each year (2). It is impossible for a researcher to go through all related articles or even abstracts to find the functionality one is interested in. Thus, there are lots of vocabularies and annotation databases to summarize the important information from biomedical literature, eg. Gene Ontology (3) and GeneRIF (4). But even with this effort, sometimes, the amount of information related to a single gene could still be very large. For example, gene "TP53 tumor protein p53", has 4509 GeneRIFs. It could take one hours or even days to read through all those GeneRIF sentences. And one could easily get lost when going through hundreds or even thousands of annotation sentences. So to better refine gene functional annotation and provide a quick idea of gene functionality more efficiently is in demand.

Also, as bioinformatics entering post genomic era, a new trend of gene research is transition from single gene to a group of genes. Those genes are usually obtained from ESTs (expressed sequence tags) and DNA microarrays experiments. Finding the similarity and difference among those genes can be helpful to uncover new functionalities. To achieve this, functional enrichment analysis are usually employed. With applying statistical tests to Gene Ontology and functional annotation, significance level of functional concepts will be calculated. However, the traditional graph and tree views of the result take a lot of space and usually are lack of information clustering. Thus, we develop graphical visualization to simplify the result and make it easier for discovering over represented concepts among a list of genes.

In this thesis, we introduce two web servers for gene functional annotation using computational methods and statistical analysis. Both of them use a visual presentation with graphs to reduce the amount of time and spaces for discovering important functional features from text. The first tool is mainly focusing on refining useful terms from text based annotation for a single gene with text mining techniques. While the second tool utilizes statistical model to find the over represented functional concepts for a list of genes.

1.2 Organization

The thesis is organized as follows:

Chapter 2 reviews the background knowledge related to this thesis. First, we briefly introduce biomedical literatures - GeneRIF and Gene Ontology applied in thesis project. Then we describe the main problem of the thesis - gene functional enrichment as well as the methods

and statistical models used for the problem. Lastly, we present the web technologies that lead to the idea of visualizing gene functional annotation graphically.

Chapter 3 presents our first work - GeneGIF that applies text mining techniques to biomedical literature resources and presents the important information of functional annotation in a word-cloud like graph. Coupled with web technologies of ASP.net, GeneGIF provides easy access to gene functionality of different levels of details from general overview to original PubMed articles.

Chapter 4 presents the gene functional enrichment tool - ListGIF. It takes a gene list as input and applies hypergeometric test on terms from GeneRIF and Gene Ontology to find out the most significant concepts among the list of genes. Finally it utilizes graphical representation from GeneGIF to add readability and amount of information displayed on the limit space.

CHAPTER 2

BACKGROUND

2.1 Biomedical literature resources

Biomedical literature is a tremendous knowledge base for gene researches. There are various resources that are trying to annotate genes in many comprehensive ways. In this work, we are focusing on two important gene literature systems: Gene Ontology and GeneRIF.

2.1.1 Gene Ontology

One source for gene researches is Gene Ontology. Ontology provides a controlled vocabulary to unify, as well as classifying the representation of genes and gene products. In GO system, each term has a unique identifier and falls into one of the three categories: biological processes, cellular components and molecular functions (5). Each GO term is annotated with predefined format in the system (6). Figure 1 shows an example of GO term.

Cellular component: is a component or substance of a cell. Eg. membrane, protein.

Molecular function: is an elementary activity which occurs at the molecular level of a gene or a gene product.

Biological process: is a series of events or activities with defined beginning and ending time to accomplish some molecular functions.

Figure 1. Example of GO term

id: GO:0008217

name: regulation of blood pressure

namespace: biological_process

def: "Any process that modulates the force with which blood travels through the circulatory system. The process is controlled by a balance of processes that increase pressure and decrease pressure." [GOC:dph, GOC:mtg_cardio, ISBN:0721643949]

synonym: "blood pressure homeostasis" RELATED []

synonym: "blood pressure regulation" EXACT []

synonym: "control of blood pressure" RELATED []

xref: Wikipedia:Blood_pressure#Regulation

is_a: GO:0065008 ! regulation of biological quality

relationship:part_of: GO:0008015 ! blood circulation

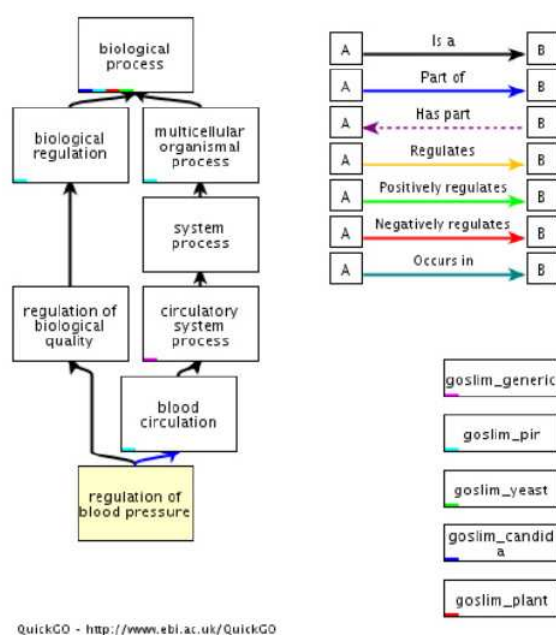
Each GO term is not a static or isolated phrase. It has various relationships to one or more terms in GO system. Those relationships can be represented by direct graph, thus building up the hierarchy of the entire GO system (7).

GO terms can be represented in directed acyclic diagram (DAG) like in Figure 2(a). DAG is a directed graph without any cycles. Two nodes in the graph are connected with a directed edge. The source node acts like a parent node in a tree structure and destination node acts like a child node. But in DAG, each child node can have more than one parent nodes. As a result, DAG also can be transformed into a tree structure with the same child node appears multiple times in different parent-child pairs. GO terms that have more than one parent nodes appear multiple times in the tree under different parent node providing different path ways to the GO term and better illustration of related GO concepts. Figure 2(b) shows inferred tree view for GO terms.

Each gene or gene product can be mapped onto one or more GO term which describes some of the features of them. Figure 3 is an example of how GO term annotating a protein. Gene Ontology also provides many tools to analyze genomic data. (7)

GO terms are well organized and standardized to describe the basic characteristics of a gene or gene product, but GO terms limit the analysis to the existing phrases which do not cover some functionalities researchers are interested in. Also, associating one gene to a GO term requires lots of effort from a biologist who may needs to go through all the GO terms and pick up those related with the genes or gene products.

Figure 2. Go term in tree view vs. GO term in graph view



(a) Go term graph view

- P** [GO:0008150 biological_process](#) [406428 gene products]
- P** [GO:0032501 multicellular organismal process](#) [42697 gene products]
- P** [GO:0003008 system process](#) [10697 gene products]
- I** [GO:0065007 biological regulation](#) [88165 gene products]
- P** [GO:0003013 circulatory system process](#) [1867 gene products]
- P** [GO:0008015 blood circulation](#) [1860 gene products]
- I** [GO:0065008 regulation of biological quality](#) [17892 gene products]
- ▼** [GO:0008217 regulation of blood pressure](#) [627 gene products]
 - I** [GO:0010850 chemoreceptor signaling pathway involved in regulation of blood pressure](#) [3 gene products]
 - I** [GO:0045776 negative regulation of blood pressure](#) [153 gene products]
 - I** [GO:0045777 positive regulation of blood pressure](#) [161 gene products]
 - I** [GO:0014916 regulation of lung blood pressure](#) [14 gene products]
 - I** [GO:0003073 regulation of systemic arterial blood pressure](#) [235 gene products]

(b) Go term in inferred tree view

Figure 3. Example of gene annotated by GO term

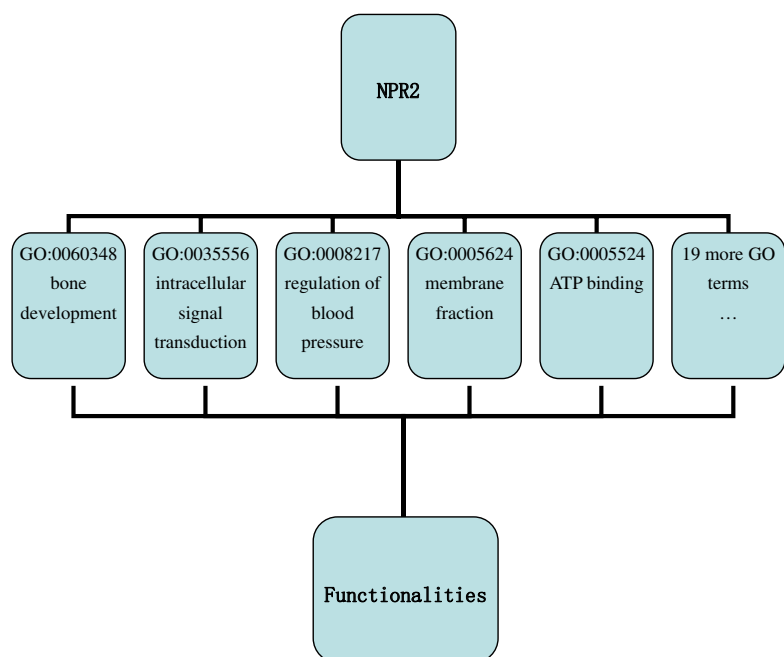


Figure 4. GeneRIF of gene "RENBP"

GeneRIFs: Gene References Into Functions [What's a GeneRIF?](#)

1. [Observational study of gene-disease association, gene-environment interaction, and pharmacogenomic / toxicogenomic. \(HuGE Navigator\)](#)
2. [Clinical trial of gene-disease association and gene-environment interaction. \(HuGE Navigator\)](#)
3. [Observational study of gene-disease association. \(HuGE Navigator\)](#)
4. [determination of catabolic role in sialic acid metabolism](#)
5. [domain structure of RBP](#)
6. [gene and protein expression were selectively activated in left ventricular myocytes from end-stage failing human hearts](#)

Submit: [New GeneRIF](#) [Correction](#)

2.1.2 GeneRIF

The other source is functional annotation which lies in literature database such as GeneRIF (Gene Reference Into Function) or PubMed. GeneRIF is a database which allows gene researchers and scientists to write functional description of a gene based on the related PubMed articles. It provides a simple but powerful way to gene functional annotation: each GeneRIF is a textual statement up to 255 characters describing the function of a gene, which gives more detailed description of genes' function compared to GO terms. Figure 4 is an example of gene "RENBP" annotated by GeneRIF sentences.

Everyone can contribute to the addition of GeneRIFs by providing the linked a PubMed ID where the GeneRIF sentence comes from. Genes that are on research hotspot usually have

a fast growth of GeneRIFs. Those GeneRIFs provide a comprehensive annotation of gene functionality. By September 2011, there are 668571 GeneRIF and 62802 genes with at least one GeneRIF. For those genes in the important research areas usually have significantly large numbers of GeneRIF. For example, gene "TP53 tumor protein p53" has 4509 GeneRIFs. And because of the rapid progress scientist have made, the number of GeneRIFs is incrementing every day. Figure 5 shows the distribution of GeneRIFs among different organisms.

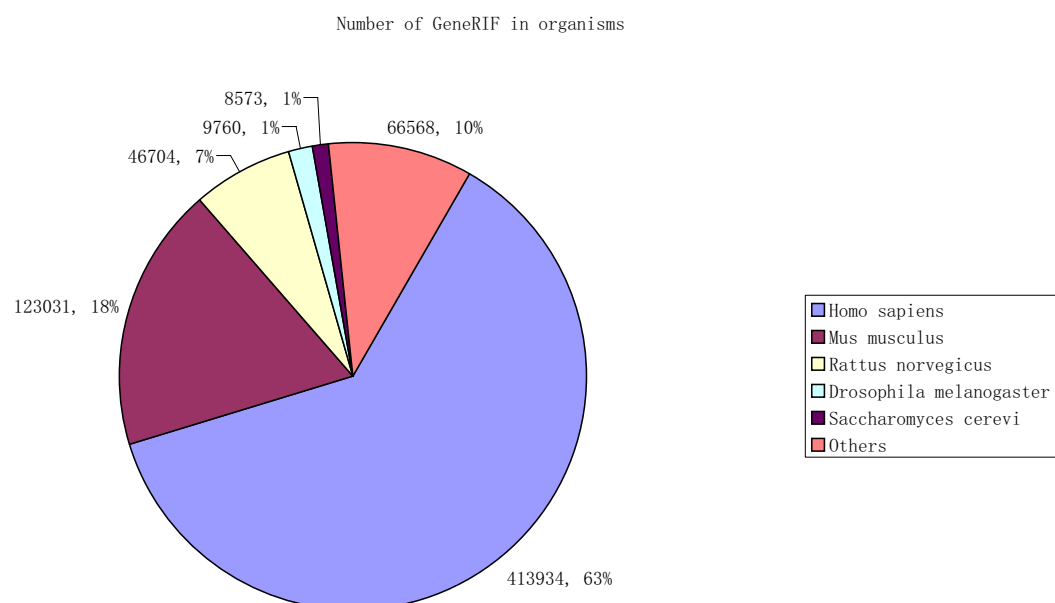
GeneRIF provides an important and relatively fast entrance to gene functionality studies. It extracts key information of gene functions from PubMed abstract or articles. So that it would be faster to get a general idea of functionality of genes. But sometimes, the number of GeneRIF for a single gene could be dozens or even hundreds. It will be time-consuming to go through all of them.

Because of the massive amount of information in biological literatures, development of methods to analyze the information and extract the most important features and relationships from it is an area that has drawn a lot of attentions from scientists.

2.2 Knowledge management and information extraction

With modern techniques, information is accumulated at a much faster rate than it has been interpreted. Generation of information can be achieved by different computational methods. But discovering important information and analyzing relationships between information entities still requires lots of human input. Particularly, in biological and biomedical areas, large amount of data is generated from micro array experiments, clinic trials and other lab reports. Using computational methods to extract and analyze information is an urgent need for biomedical

Figure 5. Number of GeneRIF for different organisms



researches. Some of the techniques for information retrieval have been developed and applied to scientific domain. Those applications have been widely used and changing the ways that how information is discovered, organized, analyzed and represented.

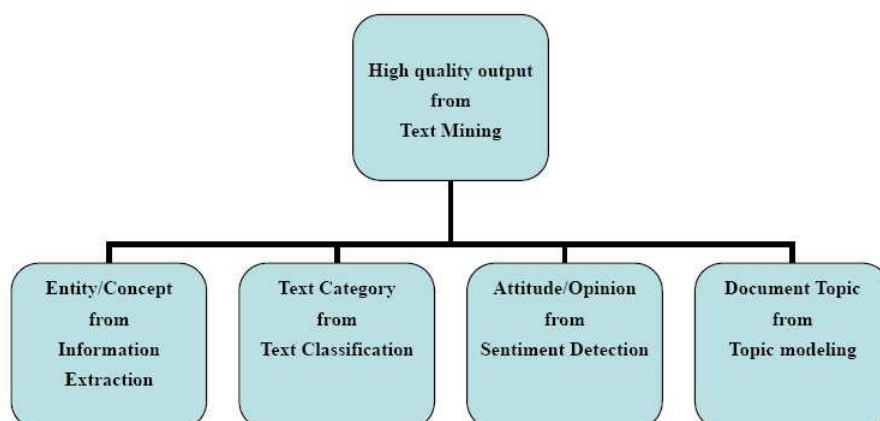
2.2.1 Text mining

One popular way for biomedical information retrieval is text mining. Text mining is the also called text data mining and can be considered as a branch of data mining. As indicated in its name, it is a process to derive high quality information from large amount of text (8). By parsing the input text data, text mining can discover the important information such as entity names, concepts and hidden patterns(9). Figure 6 shows different types of output that text mining can generate.

Calling for text mining techniques in biomedical area is a recent trend in scientific research. There are two major reasons for the need of this technique. First, fast increasing of electronic data available in private and public databases makes manual analysis of information impossible and inefficient. One major database is PubMed. There are huge amount of records in PubMed. Even with looking for a specific research subject, a large number of records could be retrieved from PubMed. And for each of those items, it is a publication with thousands of words. Even by reading only the abstract, it will be significant amount of time spending.

Apart from the considerable amount of information, another advantage that text mining can bring to biomedical research is it can discover the connections between related genes and gene products from the massive literature resource. Because of the enormous amount of publications and articles in PubMed and diversity of biomedical research, one scientist usually only works

Figure 6. Types of output from text mining process



with a small sub group of them. It is possible that they encounter some genes they have no idea of and need to get a simple and general idea of the functionality of them without spending too much time reading through all the articles and annotations for the genes.

Text mining usually starts with the process of structuring the input text by adding the derived linguistics features as well as removing non-informative words and inserting into database structure. Then it derives patterns for the structured text and finally result validation will be performed to see if those derived patterns apply to general wild text.

Text mining is the kind of technique that can provide a broader and quick view of genes and sequences in one's interest. Many text mining applications in bioinformatics area have been

developed (10). The main development of text mining in biological and biomedical area focus on the subjects like entity identification, such as finding gene or protein name from free text. After the gene and protein name has been identified successfully, the next step is to analyze the relationship of the gene and gene product, such as protein interactions (11).

2.2.2 Text based functional enrichment analysis

Gene functional enrichment analysis is aim to find the most significant functionalities of a list of genes generated from micro-array experiments. It exams some specific functionalities and determines whether they are over represented in this group of genes. Thereby it obtains the shared features from the functional side of the group of genes and uncovers the relationships between those genes and gene products. Traditionally, this was done by reviewing all the functional annotations lying in different literatures. The disadvantages of manual review are: first, there are tremendous articles in PubMed and other databases which requires significant amount of time to summarize the functionalities for the list of genes if scientist has no idea of some of them. Second, even scientist is familiar with all the genes in the list, some personal understanding could be brought into the analysis which leads to inaccuracy and bias for the result.

To solve the two problems in manual review of finding the over represented concepts in gene list, some statistical methods are employed and known as functional enrichment analysis. The most common statistical model used for over representation analysis is hyper-geometric tests, which will be explained in detail in the next section.

TABLE I
EXAMPLE OF OVERREPRESENTED FUNCTIONS FOR A GROUP OF GENES

num of genes in population: 1000
num of sub group of genes: 100

	# of expected genes in study group	# of observed genes in study group
Feature A:	50	50
Feature B:	10	10
Feature C:	50	30
Feature D:	5	20

To introduce over-represented analysis, first we will explain how features can be determined as overly represented within a group of features for a gene list. For example, we have a population of genes which acts as a background gene list and a sub set of genes which is the study group of genes of our research interests. There are several attributes we want to exam for the study group and decide which ones are overly represented in this study group. Table I is an example showing how genes are presented with four different features in gene population and in study group.

In the example above, for feature 'A', there are 50 genes are associated with it and for feature 'D', there are only 8 genes are associated with it. But we still consider feature 'D' is over represented more significantly than feature 'A' within study group of genes. Because for feature 'D', there are 20 genes out of 100 are expressed with it, which has much higher

proportion than the expected ratio 5/100 by chance. But for feature 'A', the ratio of genes expressed is 50/100 which is equal to the expected ration by chance.

This example uses the simplest way by calculating the proportion of genes associated with feature out of total genes in study group. If this proportion is significantly larger than the expected proportion by chance, we consider this feature is over represented in this study group of genes.

As there is lots of information hidden in biomedical literature, a major part of functional enrichment analysis is text based with finding the over represented features from publications or other text resources. In this case, usually a statistical model is used for sampling the genes. Different statistical models such as hyper geometric distribution, binomial distribution, χ^2 , and fisher's exact (12) test are used in different over representation tools. And there are lots of debates on which test would be best (13).

In functional enrichment analysis, another concept is p value, which represents the significant level of the concept in the given study gene group. P value is probability of obtaining such a gene list with as many genes related with some features as the study group from the background gene population. If the p value is small, it reflects the possibility of having this amount of genes related with the target feature by chance is very small, which proves the study group over represents the target feature.

Table II shows the literature resources used by different text based enrichment analysis tools. We can easily find that most of the current tools for functional enrichment analysis are based on the pre-defined Ontology (GO) or pathway annotation (KEGG) (14). The advantage of using

TABLE II
BIOMEDICAL LITERATURE RESOURCES USED BY GENE FUNCTIONAL
ENRICHMENT TOOLS

Tool	Literature resources
CLENCH (17):	Gene Ontology
ClueGO (18):	Gene Ontology, KEGG
DAVID (19):	Gene Ontology, KEGG, PFAM, NCBI
eGOn (20) :	UniGene, EntrezGene, SwissPort, Gene Ontology
FuncAssociate (21):	Gene Ontology
FunNet (22):	Gene Ontology, KEGG
GOEAST (23):	Gene Ontology
GOSurfer (24):	Gene Ontology
GORILLA (25):	Gene Ontology
Martini (26):	PubMed
MILANO:	GeneRIF, Medline
The Ontologizer (27; 28):	Gene Ontology

those pre-defined resources is they are well organized and standardized to describe the basic characteristics of a gene or gene product. On the other hand, generating concepts of ontology and pathway requires a lot of manual review from literature which will limit the scope and diversity of research area. Because of the limitation that ontology brings, some researches are trying to expand the resources for over representation analysis to a more general and broader scope (15). For example, MILANO (16) uses GeneGIF and Medline as literature resources for enrichment analysis.

There are many over representation analysis tools that have been developed. They vary from reference literatures, statistical models, visualization of result and have their own limitations (29).

Among those tools, DAVID and MILANO are two of the most popular ones. DAVID was first developed in 2003, which is one of the earliest over representation tools. One of the most useful aspects of DAVID is that it provides different graphs and charts to present the results within different biological context such as GO, KEGG Pathways and PFAM protein domains.

Another popular tool is MILANO. It was developed in 2004. It uses GeneRIF and Medline as the background literature source and allows users to specify the free text to customize the features of over representation analysis.

2.2.3 Hypergeometric Distribution

In probability theory and statistics, the hypergeometric distribution is a discrete probability distribution that describes the number of successes in a sequence of n draws from a finite population without replacement (30). Similar to Binomial distribution, hypergeometric distribution also describe the probability distribution of number of successes out of a serial of experiments. But after each draw from population, the selected item is not returned to the population, which is defined as without replacement. In binomial distribution, every draw is performed against the same population which means each selected item will be put back to the population for the next draw which is defined as with replacement.

Hypergeometric experiment is the experiment that has two properties:

1. A sample of size n is selected randomly from a population N without replacement.

2. There are only two outcomes of the result: k items are classified as success and $N - k$ are classified as failure.

Because the experiment is without replacement, each time the chance of drawing an item marked as success is different. In the model above, chance of getting a success for the first draw is k/N . But chance of success becomes $(k - 1)/(N - 1)$ for the second draw if the first draw is a success; or becomes $k/(N - 1)$ if the first draw is a failure. But in binomial distribution, the chance of success is equal in each draw which is always k/N .

In hypergeometric distribution, usually a random variable X is used to describe the number of successes of outcomes from a serial of hypergeometric experiments. Several parameters used in hypergeometric distribution as follows:

N: population size - total number of items in population $\in \{1, 2, \dots\}$

m: number of items classified as success in population $\in \{0, 1, 2, \dots, N\}$

n: sample size - total number of items drawn from population $\in \{0, 1, 2, \dots, N\}$

k: number of success in n draws $\in \{0, 1, 2, \dots, n\}$

Table III explains hypergeometric experiment with classic ball model.

With the parameters defined, the variable X of the number of successful draws which follows hypergeometric distribution can be given by Equation 2.1:

$$P(X = k) = \frac{\binom{m}{k} \binom{N-m}{n-k}}{\binom{N}{n}} \quad (2.1)$$

TABLE III
HYPERGEOMETRIC EXPERIMENT WITH BALL MODEL

	Drawn	Not Drawn	Total
Black (success)	k	$m-k$	m
White (failure)	$n-k$	$N-n-m+k$	$N-m$
Total	n	$N-n$	N

The numerator is the number of ways to get k success out of n draws from the total population of N with m items as success in it. The denominator is the number of ways to randomly draw n items from total population of N . The outcome is the probability of getting k success out of n draws from population N which has m items classified as success. If the probability is very small, which means a very unusual event occurs.

Hypergeometric distribution is widely used in sampling without replacement. In gene functional enrichment analysis, the whole genome acts like item population of N and a study group acts like a sample of n items. The genes related with the subject of interest can be classified as items of success. So, gene functional enrichment analysis has the properties of hypergeometric experiment and it is a good example to apply hypergeometric distribution in bioinformatics area.

2.3 Web based information visualization

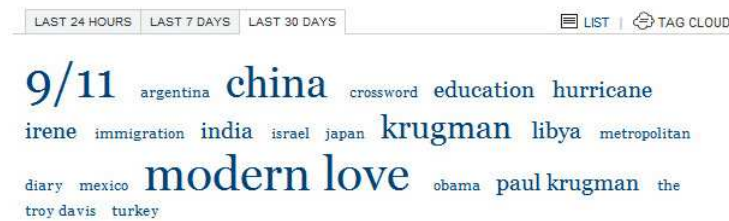
With the massive growth of information on the internet, how should we present the information in the limited space like a web page with as much information as possible and without getting user lost? There are lots of creative ideas and innovations for solving this problem in other areas like media, entertaining etc on the web. And those ideas can also be solutions to some problems in bioinformatics.

2.3.1 Word cloud

Word cloud or tag cloud is a visual representation of a collection of text (31). Figure 7 shows the most searched tags in NT times website in tag cloud. Usually, it is applied to key words or tags to label or categorize data from literatures or user inputs on websites. The tags have different weights indicating the popularity or importance of the data it applies to. And the weight is usually represented by the font size or bold colors. Words in word clouds can be arranged alphabetically or grouped by their meanings or centered by the most important tags with less important words surrounded. In most of the time, word cloud is organized in a rectangular area for neatness of a web page and better readability. But sometimes, to utilize space on web page as much as possible, word clouds also can display as a cluster of words and terms without any fixed shape of boundaries.

The first word cloud application online is Flickr (32), which is an online photograph sharing website. Tag cloud is used to sort out the most popular labels that users created for their albums.

Figure 7. Most searched topics in NY Times displayed in tag cloud



There are mainly two types of word clouds (33). Although they have similar visual representation, there is slight difference of their meanings and what those tags represent. The first type of word cloud is used for indicating the popularity of each key word or tag. The weight of a tag represents the number of items that tag has been applied to. This type of word cloud can be used to free text. Each key word in the tag cloud with different font size indicates the frequency of occurrences. The other type of word clouds serves as an index of categories. The size of tag reflects the number of items under that category. For example, an online library website can use tag cloud to label the categories of each book and generate the tag clouds with the tag size representing how large the book collection is for that category.

Because on website, usually word clouds are served for labeling and organizing data, they must present a way to explore the actual data under that tag or category. Most word clouds on the websites are represented with hyperlinks to all the text or data related to the tag. Word

clouds are used as index to make it easier for user to find what they are interested in from a large amount of information on a website (34).

Word clouds are usually implemented as inline html on web clients. The size of tag is determined by the frequencies of occurrences. If the frequencies are relatively small, they can be directly mapped to the font size of each tag. Otherwise, they need to be scaling to a range that the maximum and minimum font sizes are both readable and well proportioned to differentiate their importance.

With the growth of information on the web and high demand of innovative ways for information visualization, more and more revolutionary word cloud web sites have emerged. They have changed the original ways that word clouds used to be, adding more colors, fonts, dynamic visual effects to provide a better experience of knowledge exploring. One of the most famous and beautiful application is Wordle (35). It is an online word cloud gallery where you can create and share your own word cloud generated by the text you upload. It is not limited to the rectangular shape any more, word cloud can be tweaked in different fonts, shapes and color schemes. Figure 8 is the Wordle graph for gene "KLF4" using its GeneRIF sentences.

Generally, word cloud is a useful as well as beautiful way for information visualization, providing helpful aid to online exploring (36).

2.3.2 ASP.NET and PHP

ASP.NET and PHP are two popular technologies for web page development. They are both used to create dynamic web pages which allow user interaction and backend database connection. Both have advantages and disadvantages and are used by many popular websites.

Figure 8. Word cloud for gene "KLF4" using Wordle



ASP.NET is a successor of ASP which stands for Active Server Pages. It is developed by Microsoft for building dynamic web application. ASP.NET separates the code for actual web event from the static HTML web forms, where the web designer can focus on the web page creation without any interrupt of server side code. It achieves the isolation of presentation with content (37). Also, ASP.NET provides a large selection of user controls including custom controls which makes it even easier for building web pages. Also ASP.NET embraces the extension of AJAX which handles asynchronous request for web server using javascript without refreshing the entire web page. ASP.NET also support multiple programming languages like C++, C#, VB.net, Python, Perl, Java and Delphi. But ASP.NET also has its disadvantages. ASP.NET is a product of Microsoft and it is run using Internet Information Services (IIS) and it is mostly hosted on Windows platform, which means the deployment of ASP.NET website can cost a lot. Also, ASP.NET is designed for Windows platform and has limited portability to other operating systems.

PHP is an open source language for creating dynamic web pages. It stands for Hypertext preprocessor. Unlike the ASP.NET, PHP code usually is embedded into html pages to response to the server side events. PHP can run on most operating systems, like Windows, Linux, Mac OS. So PHP has very good crossing platform ability and absolutely free of development and deployment. PHP also supports most databases like SQL server, MySQL etc. like ASP.NET does and it can be edited in most text editor like Vi, or notepad. However, because PHP is an interpreted language, the speed could be slower compared to ASP.NET which is a compiled language. As a result, PHP is more suitable for developing small to mid size website instead

TABLE IV
COMPARISION OF ASP.NET AND PHP

	ASP.NET	PHP
Released by	Microsoft	PHP Community
Source code mode	Closed	Open
Cross platform	Strongly tied to Windows	Any
Price	.NET framwork is free, but Windows and IIS are not	Free
Type	Compiled	Interpreted
Languages	C#, VBScript, C++, etc.	PHP
Database	Most Databases	Most Databases
Extensibility	.Net based extensibility	PECL:PHP Extension Community Library
Recommended for large systme	Yes	No
Performace	fast	fast for small system

of very large and complex website. We compared ASP.NED and PHP in different aspects in Table IV.

CHAPTER 3

GENEGIF

3.1 Introduction

GeneRIF (Gene Reference Into Function) provides a simple way to gene functional annotation: each GeneRIF is a textual statement up to 255 characters to document the function of a gene. One can scan the functions of a gene through each GeneRIF quickly. However, when a gene has dozens or even hundreds of GeneRIFs, it will be time-consuming to go through all of them. The situation is getting worse when users need to get some ideas of a long list of genes, such as over-expressed genes from microarray experiments. The users will get lost in the information and miss the message of interests. To get a general idea of what functions a gene has, a faster and more intuitive way is in demand.

The first observation is that for genes with many GeneRIFs, there are quite some overlaps among each of them. Thus if we emphasis the keywords and present them with the importance associated, it will be easier for users to get the idea of functionalities.

One solution to this problem is the so called in-line html tag cloud which appeared firstly in Douglas Coupland's book(38) in 1995. Tag cloud drawing has become a popular way to display data with its frequency used in many website such as New York Times (39) and Flickr. The advantage of this method is that it delivers the important information by a bunch of key words according to its popularity and avoid going through several paragraphs of sentences. Usually,

in this method, to check what a key word is referring to, one has to navigate to a new page. However, one needs to go back to the tag cloud homepage to check the details of another word. A recent solution is to use a nice and convenient word clouds graph called Wordle which is developed by Jonathan Feinberg. The graph is also generated base on the word frequency. The font size is proportional to its word counts. Compared with in-line html tag, Wordle-like graphs utilize space more effectively by meshing up the words. To enhance readability, colors and directions are added to each word.

To compare different representations of gene annotation, a survey⁽⁴⁰⁾ was conducted among experts who work on genomic data analysis using microarrays. In this survey, a graph (similar to Figure 8) is generated by Wordle based on GeneRIF of gene KLF4 followed by 8 questions related with usage of GeneGIF and participant characteristics (such as gender, age, education level, study field and native language). 53 valid responses were collected in the end. The result showed that in terms of usage, 64% of the users were either positive or neutral toward using GeneGIF in their daily work; in terms of preference, 51% of the users preferred visual (GeneGIF) information than textual (GeneRIF) information.

Some of the participants gave very useful comments on both the advantages and drawbacks of these two types of gene annotation methods: GeneRIF vs GeneGIF. Most participants think GeneGIF is more convenient when one needs to check functions of many genes at a time or get an outline of the functions. It gives a quick idea of the functional annotation for a gene. But traditional GeneRIF provides a more precise description of gene function. It is necessary when

one needs to study the gene function in detail. Many of the participants suggested to make GeneGIF clickable so that the GeneRIF could be displayed when needed.

In this work, we are presenting an visual annotation tool: GeneGIF which combines Tag cloud, Wordle-like graph as a pluggable web 2.0 component to biologists.

3.2 System and methods

3.2.1 Data set

The genes used in this project cover the entire human genome whose Taxonomy ID is 9606 in the NCBI database. By September 2011, there are 40765 human genes and 413937 GeneRIFs . 15926 of human genes have at least one GeneRIF related to it. So on average, each gene has about 26 GeneRIFs. Figure 9 shows the statistics of how many genes for each range of number of GeneRIF sentences.

The first thing is to get the data file from NCBI gene site. We examined each file and extracted the necessary information into our database. Thus, information of genes, GeneRIFs and Gene Ontologiy are well organized in data tables to optimize query and update.

3.2.2 Text mining

Before generating the graphs, first we needed to pre-process the sentences in GeneRIFs and derive a word frequency list from the raw GeneRIFs. In this stage, some meaningless words (stop words) will be removed while terms related to genomic data will be identified.

3.2.2.1 Stopwords removing

After carefully checking the graph in Figure 8, we found some words in the graph with high frequency but useless in terms of gene function. We divided them into three domains: common

Figure 9. Number of genes vs. number of GeneRIFs

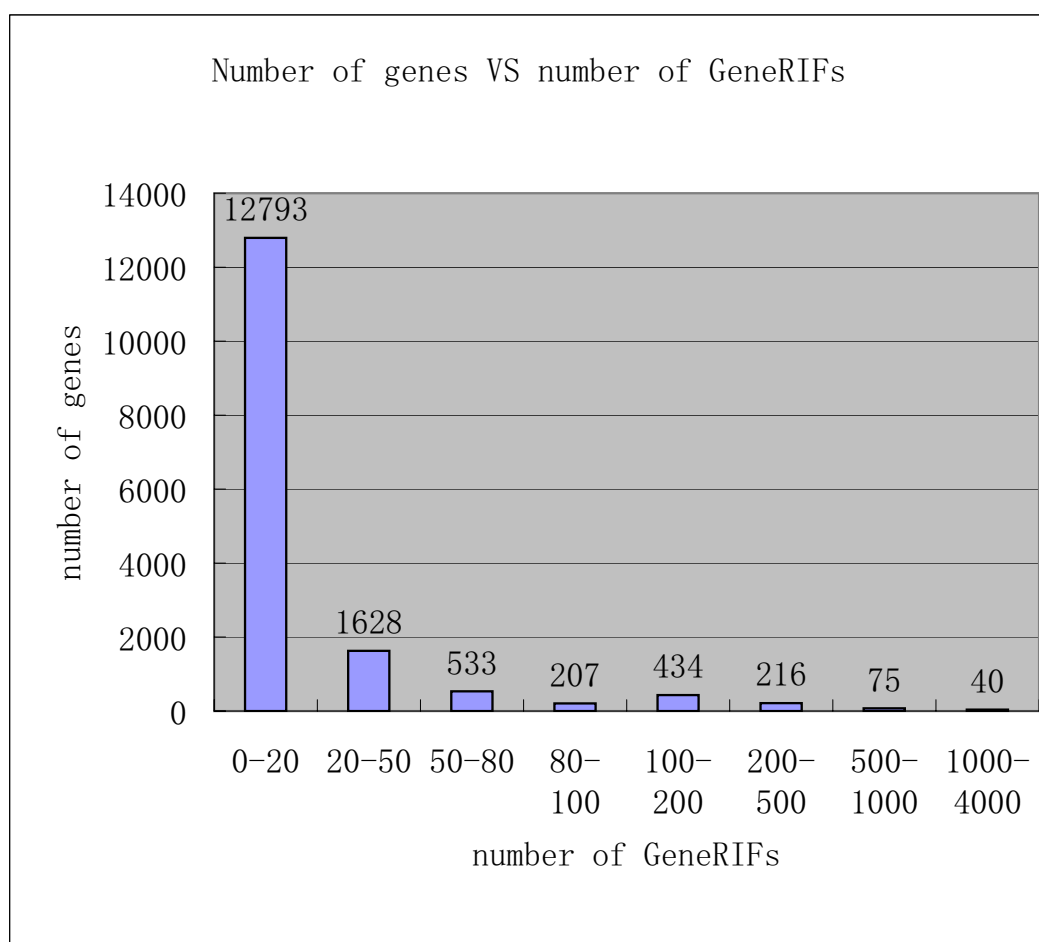


TABLE V
STOPWORD LIST IN GENEGIF

Common English stopwords		Biology domain-specific stopwords		Expert suggested stopwords		Self-referring stopwords
word	frequency	word	frequency	word	frequency	word
of	676768	cell	53344	show	21508	KLF
the	518474	gene	44659	play	12778	MSH6
and	467563	protein	44383	signal	4987	
in	448203	receptor	27148	result	2517	
a	204164	dna	11377	review	1700	
to	172195	active	4724	transfer	1180	
is	142363	bind	2281	finding	612	
that	135835			paper	346	
with	124964			prove	160	
by	101519					
for	97942					
be	36879					
at	26704					
which	24524					

English stopwords (eg. "the", "of"), biology domain-specific stopwords (eg. "active", "protein") and experts suggested stopwords (eg. "paper", "review"). These words should be filtered out before we generate our graphs. Also the self-referring words, such as the gene symbol ("KLF4" in Figure 8) and the gene name are removed from the graph. Table V shows most common stopwords with their frequencies in GeneRIF in each category. By September 2011, total number of words in GeneRIF sentences is 12305989.

TABLE VI
MORPHOLOGICAL RULES USED TO DERIVE BASE FORM OF A WORD

Nouns	Verbs	Adjectives
"s" → ""	"s" → ""	"er" → ""
"ses" → "s"	"ies" → "y"	"est" → ""
"xes" → "x"	"es" → "e"	"er" → "e"
"zes" → "z"	"es" → ""	"est" → "e"
"ches" → "ch"	"ed" → "e"	
"shes" → "sh"	"ed" → ""	
"men" → "man"	"ing" → "e"	
"ies" → "y"	"ing" → ""	

3.2.2.2 Morphological unification

Next, we unify the different forms of nouns and verbs into their prototype to make the frequency of each word more accurate. For nouns, plural nouns will be restored to the single form. For example, the word "cells" is considered the same as its singular form "cell". For verbs, all different tenses will be restored to present tense and first person. For example, "regulates", "regulated", "regulating" and "regulation" will be counted into the frequency of "regulate". Porter Stemming Algorithm (41) is used to implement this function. Table VI shows rules applied for morphological unification.

We used free library "Morphy" from WordNet (42) for morphological unification. It applies different rules for detaching the suffixes for nouns, verbs and adjectives. Also there is an exception list containing all the words that don't meet any of the rules of detachment. It works

as follows: when a word is passed into "Morphy", the program first looks up the exception list. If it is not found in the list, morphological rules will be applied according to the syntactic category. Because some of the words (eg. "axes" with base form as "axe" or "axis") may have several base forms, the program will return every possible base form for the input and stop until NULL is returned. The process also illustrated in Figure 10.

3.2.2.3 Phrase recognition

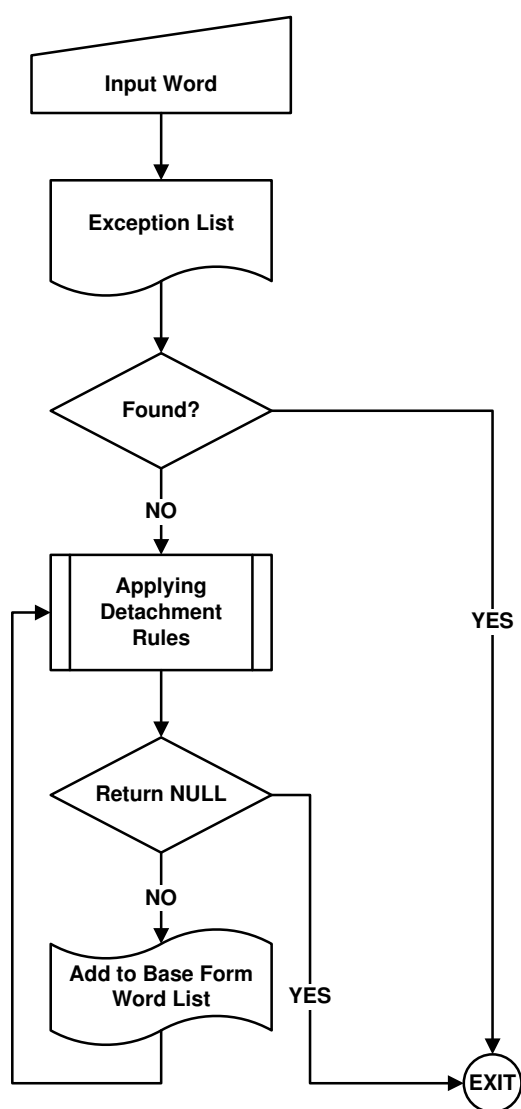
In gene annotation, sometimes, phrases contain much more information than single words and should be parsed as a whole term. For example, "cell cycle" is recognized as one term, because neither the individual word "cell" nor "cycle" can express its meaning. And for the phrase recognition part, we use the Gene Ontology terms as our glossary to define the meaningful phrases. Currently, we only identify phrases with two words.

3.2.3 Generation of graph

In our GeneGIF project, one of our ideas is to use an efficient way to present the result of our analysis, which makes it easier for users to pick up the important information from the massive amount of outputs. As in other annotation tools, usually result is gathered into tables or list or tree structures. And the amount of information can be overwhelming and users are required to click one after one link to see the information they need.

Given the processed word frequency list, we can generate a colorful annotation graph using Thomas Boutell's open source GD-library (43). With GD-library, it is very easy to generate the bounding box (the smallest rectangle containing the word) of the word with arbitrary font, color, size and rotation.

Figure 10. Data flow for mophological unification



The font size is based on our text mining result, proportional to its frequency and scaled for best viewing. If ft_{max} is the maximum font size, ft_{min} is the minimum font size, Ct_{max} is the frequency of the word occurs most, Ct_{min} is the frequency of the word occurs least within the GeneRIF sentences of one gene, Ct is the frequency of the word, then font size of this word Ft can be calculated as in Equation 3.1:

$$Ft = ft_{max} \cdot \frac{(Ct - Ct_{min})}{(ft_{max} - ft_{min})} \quad (3.1)$$

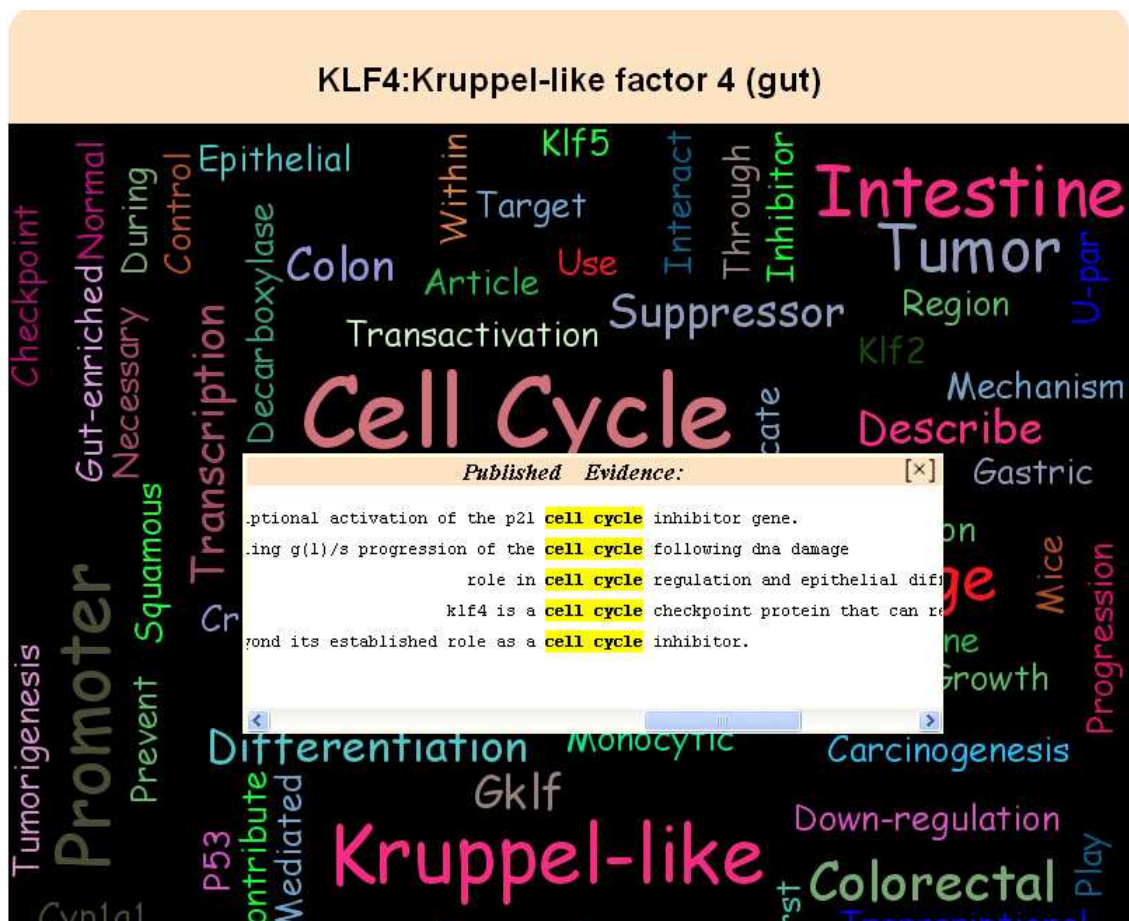
Therefore, each tag in the tag cloud can be assigned with appropriate size with proportional to the tag that has largest frequency. The position and rotation of each word is currently set to random which means we randomly throw the word on to the screen until it fits. See Figure 11 for an example.

To reduce the chances of collision when throwing the next word onto GeneGIF graph when generating the graph, we started with the word with largest size. So that when available space is decreasing, words with smaller font size are added to the graph and there are better chances for small font sized word to fit the space.

3.2.4 Advanced web features

Based on our graph, we implement advanced features to enhance the usability. We provide different search configurations. User could search genes with ID, symbols and with options to limit the search key word only in gene names or GeneRIF sentences Figure 12(a). After user searching for the specific gene according to its ID or symbol, in the list of search result,

Figure 11. The GeneGIF of gene "KLF4" and the related GeneRIFs when clicking a specific word. The words related to the important gene functions like "Cell Cycle", "Cancer" etc. stand out in the graph.



a thumbnail of the GeneGIF Figure 12(b) will display when mouse moves over the gene name. Users can look through each image to get a quick idea of genes' function. If one is interested in a specific gene, the full size annotation graph will be presented after clicking the gene name. On this full size graph, when user clicks on each word, a panel of all the GeneRIFs containing this word will be displayed with key words highlighted. The panel can be dragged anywhere on the page. Also, each GeneRIF is a clickable link to the original Pubed article where this GeneRIF comes from. With this implementation, the detailed information of GeneRIF is included and user can easily trace back to the published evidence of the functionality.

Another feature GeneGIF provide is embeddability. For each GeneGIF graph, we provide a piece of simple HTML codes to let user conveniently embed GeneGIF graph to one's own work. For example, GeneGIF can be seamlessly embedded to any biomedical articles or database, which GeneGIF thumbnail graph can appear when mouse over a gene name.

For each GeneGIF, we provide a discussion board (Figure 13). The discussion board is pre-populated with the RefSeq records. Everyone can contribute their ideas of the genes, GeneRIFs, GeneGIF graphs, agreement or disagreement of previous comments. No login required for making comments. GeneGIF provide a friendly environment of communication between anyone interested in genomic research and knowledge discovery.

GeneGIF is implemented with ASP.NET. Reusable controls in ASP.NET make implementation convenient and more standard. Also, precompiled codes in ASP.net allow faster user interaction with GeneGIF. To further prompt the performance of GeneGIF, we also move all work of text preprocessing from online to offline. Each graph is generated in advance and stored


Gene ID	Gene Name	Symbol
482	ATPase, Na+/K+ transporting, beta 2 polypeptide	ATP1B2
490	ATPase, Ca++ transporting, plasma membrane 1	ATP2B1
491	ATPase, Ca++ transporting, plasma membrane 2	ATP2B2



Gene ID	Gene Name	Symbol
19	ATP-binding cassette, sub-family A (ABC1), member 1	ABCA1
20	ATP-binding cassette, sub-family A (ABC1), member 2	ABCA2
21	ATP-binding cassette, sub-family A (ABC1), member 3	ABCA3
22	ATP-binding cassette, sub-family B (MDR/TAP), member 7	ABCB7
23	ATP-binding cassette, sub-family F (GCN20), member 1	ABCF1
24	ATP-binding cassette, sub-family A (ABC1), member 4	ABCA4
47	ATP-citrate lyase	ACLY
172	Acetyl-coa	AFG3L1
215	Require	ABCD1
225	Diabetic	ABCD2
368	Histone acetylation	ABCC6
439	Factor-induce	ASNA1

(b) Search result page with thumbnail of GeneGIF when mouse over a gene name

Figure 13. Discussion board for GeneGIF graph with prepopulated RefSeq records

Patient
Squamous Require Carcinoma Survival

 Community Discussion Board (for this gene only)

2009-1-01 11:00:00  0  0

This gene encodes tumor protein p53, which responds to diverse cellular stresses to regulate target genes that induce cell cycle arrest, apoptosis, senescence, DNA repair, or changes in metabolism. p53 protein is expressed at low level in normal cells and at a high level in a variety of transformed cell lines, where it's believed to contribute to transformation and malignancy. p53 is a DNA-binding protein containing transcription activation, DNA-binding, and oligomerization domains. It is postulated to bind to a p53-binding site and activate expression of downstream genes that inhibit growth and/or invasion, and thus function as a tumor suppressor. Mutants of p53 that frequently occur in a number of different human cancers fail to bind the consensus DNA binding site, and hence cause the loss of tumor suppressor activity. Alterations of this gene occur not only as somatic mutations in human malignancies, but also as germline mutations in some cancer-prone families with Li-Fraumeni syndrome. Multiple p53 variants due to alternative promoters and multiple alternative splicing have been found. These variants encode distinct isoforms, which can regulate p53 transcriptional activity.
[provided by RefSeq]

Add Comments

on web server cutting off all unnecessary waiting time of response when users visit GeneGIF.

Figure 14 illustrates the GeneGIF system architecture.

3.3 Result and conclusion

GeneGIF provide a convenient and innovative way for biomedical information discovery. All raw GeneRIF sentences are preprocessed to refine the text that will be displayed in the GeneGIF graph. So that, in the limited spaces, users can view meaningful information as much as possible. The graphic layout makes knowledge discovery much easier and more intuitive. Combining thumbnails, graphs, GeneRIF sentences and links to original PubMed articles to four tiers of information (Figure 15) meet the various demands of how detail the information should be for different users. User can choose the detail level they want to go into. If one only want to get a general idea of gene, a thumbnail or GeneGIF graph may be sufficient. If one is interested or specialized in some genes, the detailed GeneRIF or even the original PubMed article are also necessary.

One of our future enhancements for GeneGIF is to extend phrase recognition to phrases with more than 3 words. Currently, only single words or phrases with two words are parsered. Also, we have observed that some synonymous words describe similar functionality in GeneRIF such as "promote" and "boost". We can experiment with grouping those synonyms and test out if the weight of each key word in graph will be more accurate. WordNet is one of the free software that includes most of English words and groups them into 117 000 synsets (sub sets of synonyms) and can be used for identifying synonymous in GeneGIF.

Figure 14. GeneGIF system architecture

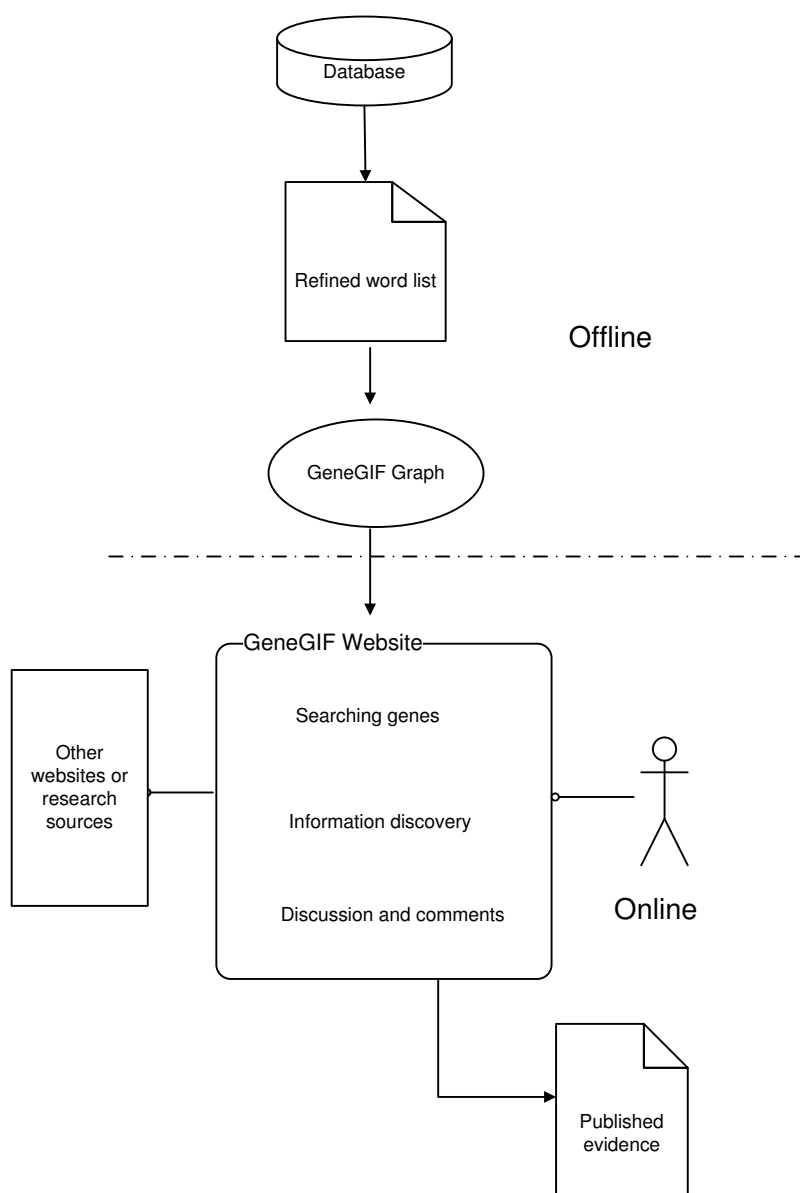
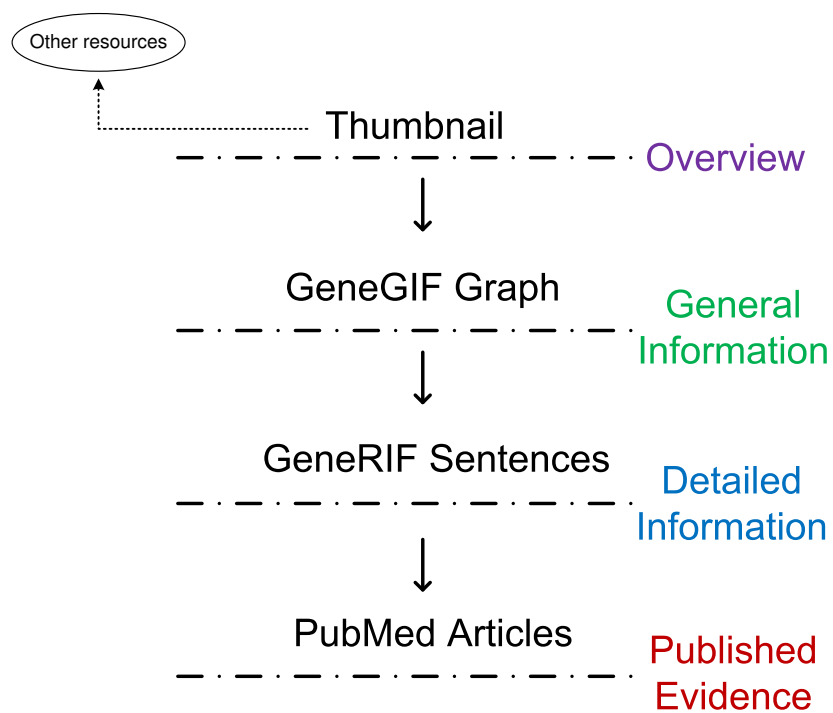


Figure 15. Four tiers of information levels in GeneGIF



In this work we have implemented GeneGIF for gene annotation which used of tag cloud and Wordle-like graphic techniques. To our knowledge, this is the first report of annotating the functions of each gene in the human genome visually. In the mean time, we employ new web tools to make the website more user-friendly. The project can be accessed here <http://proteomics.bioengr.uic.edu/genegif/>. This graphic annotation can be easily embedded in other database and resources.

CHAPTER 4

LISTGIF

4.1 Introduction

Post genomic era explores a new field for gene functional study: from one single gene to a group of genes. Studies on a single gene could reveal its functionality which may have huge impact on the entire organism (44). Meanwhile, with DNA microarray experiments, a list of genes is usually obtained, which leads to the study of functional annotation for a group of genes. One of the most important studies of a list of genes is functional enrichment analysis which is to find the overrepresented concepts among these genes.

One source for functional enrichment analysis is using Gene Ontology, such as DAVID. In GO system, every gene or gene product can be mapped onto some GO terms. Those GO terms are divided into three categories: biological processes, cellular components and molecular functions. Statistical test could be performed on those GO terms and identify the significance of each term associated with the list. GO terms are well organized and standardized to describe the basic characteristics of a gene or gene product, but GO terms limit the analysis to the existing phrases which do not cover some functionality researchers are interested in. Also, associating one gene to a GO term requires lots of effort from a biologist who may needs to go through all the GO terms and pick up those related with the gene or gene product.

The other source is using functional annotation which lies in literature database such as GeneRIF or PubMed for gene functional enrichment analysis. GeneRIF is a database which allows gene researchers and scientists to write functional description of a gene based on the related PubMed articles. It gives more detailed description of genes' function compared to GO terms. However, reading GeneRIFs is time consuming since each gene may have several even hundreds of GeneRIF sentences. Especially, when researchers want to compare the common features of a list of genes, it is easy to get distracted if one need to read all those GeneRIFs.

To overcome this limitation, some applications such as MILANO asks users to limits their query to specific words and provide the analysis which is only related to the given words. However, this requires users to have pre-understanding about the gene list. Also, key word only searching may hide some important information from users.

Based on those considerations above, we have developed a web server-ListGIF, which provides a comprehensive analysis and interactive interface to identify the overrepresented concepts from a list of genes.

The basic idea is that the co-occurring terms in functional annotation of many genes from a list must reveal some common features of the list. We apply text mining techniques and hypergeometric test on Gene Ontology terms and GeneRIF text to rate the significance of each term occurred in those two kinds of annotation based on the p-value from the hypergeometric test. Then a text based graph is drawn with each preprocessed terms of different sizes according to its significance. The graph therefore represents the significantly overrepresented concepts of the gene list.

To demonstrate the superiority of ListGIF, we tested it on several gene lists which had been analyzed in previous publications. The graphic results shown in ListGIF indicate the most significant functions and over-represented concepts as suggested and discovered in previous researches.

4.2 Methods

4.2.1 Hypergeometric distribution for enrichment analysis

We start with the assumption that each informative word from GeneRIFs or GO terms is related to the function of genes. A glossary is built to count the occurrence of each word for every individual gene in human genome. However, the simple frequency can't describe the significance of each word, since some terms occur frequently and evenly in the GeneRIF sentences for the entire genome. High occurrence could not guarantee an over-representation of a gene function. On the other hands, some words are exclusively related to the given gene list with small number of occurrence. In this case, only counting the number of frequency will underestimate the significance. To avoid this inaccuracy, we calculate the probability of getting such a gene list same as the input list by chance by hypergeometric test. Our goal is to discover the most important words by counting the occurrence and comparing it with the expected counts from chance for the user-defined gene list.

There are two levels of probability from hypergeometric distribution being investigated in ListGIF: Gene Level and GeneRIF sentence level. In gene level test, the gene would be counted only once for a term if this term occurs at least once in GeneRIF text. Let N be the number of genes for the entire human genome and m be the number of genes which have at least one

count for word 'A'. The size of the gene list to be analyzed is n and k of them have at least one count for word 'A'. Using X as the random variable of number of genes whose GeneRIF contain word 'A' for the given list, we apply Equation 4.1 to obtain probability of getting such a list for word 'A' of gene level:

$$P(X = k) = \frac{\binom{m}{k} \binom{N-m}{n-k}}{\binom{N}{n}} \quad (4.1)$$

Probability of gene level serves a good measurement of significance analysis in most gene list studies. However, for some words, if there is an extremely unbalanced distribution of key words among GeneRIFs, it would introduce some bias when using gene level probability. For example, if we have a study feature 'A' and two gene lists with the same number of genes. If in the first gene list, there are 10 genes with total 100 GeneRIF sentences containing word 'A'. While in the second gene list, there are 10 genes with only total 20 GeneRIF sentences containing word 'A'. We believe that word 'A' is significantly overrepresented in the first gene list, since there is more published evidence of feature 'A' for the first gene list's genes. But with gene level analysis, there is no difference between probabilities calculated for word 'A' for the two gene lists.

To overcome this issue, we propose another probability of GeneRIF sentence level. In GeneRIF sentence level test, the method to obtain probability for word 'A' is similar to that of gene level except the calculation of term related count is done at GeneRIF level. The count for a term will increase by one every time one more GeneRIF sentence has this term. The probability of getting such a list for one single term 'A' in GeneRIF level can be described

as follows: let N_r be the number of GenRIFs for the entire human genome and m_r be the number of GeneRIF sentences which have at least one count for word 'A'. The total number of GenRIF sentences for the gene list to be analyzed is n_r and k of them have at least one count for word 'A'. Using X_r as the random variable of counts for word 'A' in the given list, we use Equation 4.2 to obtain probability of getting such a list for word 'A' of GeneRIF level is:

$$P(X_r = k) = \frac{\binom{m_r}{k} \binom{N_r - m_r}{n_r - k}}{\binom{N_r}{n_r}} \quad (4.2)$$

In both gene level and GeneRIF level test, the smaller probability represents more rarely this event will occur by chance. That means getting such a list with k number of genes or GeneRIF sentences containing this word has an extremely small possibility to happen. But in our input list, the probability is 100% for the given list as we have observed this event. So we use 1 divided by the probability by chance and get the significance level of association of the word with the gene list. For each word occurring in GeneRIF sentences of the given gene list, we execute calculation independently and get the probability.

Also, we apply statistical model to Gene Ontology similarly as to GeneRIF for gene level. If a gene is annotated with a GO term, it is considered as a success of a draw. Thus, with total N genes in human genome and m of them are related with the target GO term, for a gene list with n genes, if there are k genes related with a specific GO term, we consider k success draws out of n draws. And the same equation of Equation 4.1 can be applied to calculate probabilities getting a list as the input list of each GO term.

4.2.2 Graphic visualization and web access

ListGIF provides a graphic representation for the word extracted from GeneRIF sentences and terms in Gene Ontology using open source library GDLib, which is inspired by Wordle. In this implementation, words can be drawn with arbitrary color, size, rotation and font in a rectangular hidden bounding box. The font size is based on the p-value calculated by hypergeometric test and indicates the significance level of each word.

Before carrying out hypergeometric test and drawing the graph, we first get the data from NCBI gene database including data of genes, GeneRIFs and GO terms. Then, we pre-process the raw GeneRIF text to get the informative words by removing other English common words. For each gene list, we search the related GeneRIF text of the given gene list and maintain a three column table in our database including the word, the gene which contains this word and the occurring frequency in its GeneRIF text. Similarly, a word frequency table is created for each GO term occurred for genes in the list. With this data, the number of genes related to this word of GeneRIF or GO term is calculated and p-value is obtained. After all the p-values are derived, we calculate maximum and minimum font size and number of font size level based on the number of words to be displayed and variance of p values to ensure optimal readability and space utilization.

ListGIF is easily accessed through a web form. User defines a list of gene with official gene symbol and gets the graphic result by one click. The detailed GeneRIF sentences in which the word is contained are displayed when user mouses over the word. Also, the genes that contain this key word are displayed to show the specific sub gene group that is functional enriched of

Gene Visual Annotation Website

Provide Mutation Analysis Finding Potential

4.3 Results

To evaluate our program, we test it on several gene lists which were used in previous research and publication. Following are results from two sample lists.

To evaluate our program, we test it on several gene lists which were used in previous research and publication. Following are results from two sample lists.

Gene List 1: A small 50-gene list (45) categorized in three domains: (a) development; (b) Alzheimer’s disease (AD); (c) cancer biology. Results in ListGIF verifies the characteristics of this list with significant p values of term ”Alzheimer” and ”Cancer” ranking second and seventh respectively of all informative terms. Also word ”development” has appeared in many GO terms like ”forebrain development”, ”heart development” and ”embryo development”. (Table VII).

Gene List 2: Breast cancer gene list (46). This is a list of breast cancer genes which contains 70 genes. With this gene list, it discovered that ”matrix metalloproteinase 9” and ”VEGF” are associated with breast cancer prognosis (47). From the result of gene level, one could easily get an idea of the significant features of this gene list are ”breast”, ”cancer”, ”differential”, ”tumor” etc. Coupled with GeneRIF level result, some detailed terms such as: ”matrix”, ”Metalloproteinase 9”, ”Mmp”, ”vegfr” and ”vegfr” would easily help researchers uncover the strong relationship that ”Matrix metalloproteinase 9” and ”VEGF” are associated with breast cancer (Table VIII).

ListGIF also reports the most overrepresented concepts from GO terms for the given list of genes. Comparing with result from DAVID, ListGIF not only ranks each GO term with its calculated p-value and identifies the genes related to this GO term, but also provides the PubMed evidence to the relationship of gene and GO term. Also, the graphic representation dramatically reduces time and space of identifying concentrated information from a list of genes.

4.4 Discussion and conclusion

In this paper, we present a revolutionary way for gene list annotation. It utilizes the space to display large amounts of information on a single page and improve efficiency of important

TABLE VII
PROBABILITY FROM HYPERGEOMETRIC DISTRIBUTION FOR OVER
REPRESNETED TERMS IN 50-GENE LIST

Top over represented concepts of GO terms	
GO0005515: protein binding	1.9579E-13
GO0016563: ranscription activator activity	5.57117E-10
GO0007219: Notch signaling pathway	3.06203E-09
GO0030900: forebrain development	3.06203E-09
GO0043025: neuronal cell body	3.46861E-09
GO0001764: neuron migration	7.65973E-09
GO0042802: identical protein binding	1.35672E-08
GO0008219: cell death	3.85541E-08
GO0030424: axon	3.85541E-08
GO0009790: embryo development	3.97237E-08
GO0005886: plasma membrane	1.88849E-07
GO0007275: multicellular organismal development	2.85014E-07
GO0007507: heart development	4.92492E-07
GO0046982: protein heterodimerization activity	1.38453E-06
GO0007399: nervous system development	4.97692E-06
GO0006897: endocytosis	7.07226E-06

Top over represented concepts of GeneRIF level	
Tgf-beta	9.10E-281
Alzheimer	2.08E-242
Breast	3.00E-197
Amyloid	1.89E-142
Epsilon4	1.71E-141
Epidermal	3.82E-131
Cancer	1.43E-122
Presenilin	1.97E-94
Notch	8.82E-91
Apoe4	2.36E-85
Mutation	4.09E-83
Growth	3.19E-81
Apolipoprotein	2.23E-79
Brca2	3.92E-72
Observational	7.14E-72
Gene-disease	1.98E-68

TABLE VIII
PROBABILITY FROM HYPERGEOMETRIC DISTRIBUTION FOR OVER
REPRESNETED TERMS IN BREAST CANCER GENE LIST

Top over represented concepts of gene level	
Cancer	0.00566463
Differential	0.00717815
Tumor	0.0124176
Essential	0.013856
Enhance	0.0247058
Represent	0.026507
Proliferation	0.0396574
Epithelial	0.0411594
Growth	0.047448
Transcription	0.0477475
Breast	0.0543682
Phosphorylation	0.0640073
Carcinoma	0.065562

Top over represented concepts of GeneRIF level	
Matrix	1.45483E-91
Metalloproteinase-9	9.36207E-80
Vegfr-1	1.02323E-63
Igfbp-5	5.18242E-63
Mmp-2	3.24722E-60
Metalloproteinase	6.31738E-48
Flt-1	6.96683E-42
Sflt-1	7.20692E-41
Vegf	1.77548E-38
Timp-1	1.65555E-29
Mmp	1.27366E-28
Mutation	2.27833E-23
Preeclampsium	1.76974E-21

features discovery for a list of genes. There are four main advantages of ListGIF. First, we combine GO terms and GeneRIF text to enlarge the source of literature mining. The result is more informative and richer compared to that using GO term or GeneRIF only. Second, we provide two levels of hypergeometric test to reduce the bias that unbalanced GeneRIF distribution causes. Third, we introduce an innovative way to present the result with easily readable text graph, which ensures scientists to identify the important gene features more efficiently. Lastly, for better analyticity and traceability, we also provide details and links to published evidence of GO terms and GeneRIF sentences for those who want to make further investigation on the results.

The main issue of ListGIF is that there are some uninformative words occurring in the results. Although the original GeneRIF text is pre-processed and the terms displayed in the graph are selected based on a threshold of p-value, some field related words with little information still has high frequency to occur due to the universal description for particular biological process. For example, the top word with highest frequency in GeneRIF text is "associate", however, in most case, the two entities which are associated are much more meaningful to users. After carefully examine the graph, we found that most verbs don't bring much information but they usually have high frequencies. Identifying different part of a sentence such as nouns, verbs, subject and object will be the next step of ListGIF. Shallow parsing can be used to achieve this (48). Currently, with the implementation of showing original GeneRIF sentences by mouse over the term, users could see the detailed information about the term, which solve the issue on a certain level.

Result from ListGIF also reveals some unexpected term in the graph, which indicates ListGIF may help in finding new common features for a list of gene. We will validate this by testing on more datasets and verify the result with experiments. Third, this Wordle-like graph is not only limited to the use in genomic study. It is a convenient way to organize large amount of data by key word on limited space.

One of the enhancements of ListGIF is we can add query option for PubMed ID, which allows user to input a list of PubMed IDs and ListGIF will performed the enrichment analysis on all the PubMed abstracts that are retrieved by those IDs. Also, we could allow more than one gene lists as input, in which case, user may want to compare the similarity of over represented concepts between those two lists. Lastly, we can group queries into sub domains like drugs, chemicals, genes, diseases, symptoms and organism. So that, analysis will be performed on the concepts that are more interesting to user and unrelated information will be filtered out. There are some tools that work as biomedical dictionaries (49) and pre-tag key words from PubMed abstract into different sub domains.

In this paper, we present a novel representation for knowledge discovery: ListGIF. This tool can be access from <http://listgif.nubic.northwestern.edu/>. ListGIF emphasizes accuracy and interactiveness of gene functional enrichment analysis by combining hypergeometric test with the graphic display of result. It applies analysis to Gene Ontology and GeneRIF to derive concepts from both controlled literature and free text literatures. It shares a similar representation of GeneGIF which is a functional annotation tool for single gene. But in ListGIF for a group of gene, hypergeometric test is added to reduce the bias of analysis and test result

on several gene list shows that ListGIF can provide accurate functional enrichment analysis with more readable and feasible visual presentation.

CITED LITERATURE

1. J. McEntyre and D. Lipman. PubMed: bridging the information gap. *CMAJ*, 164:1317–1319, May 2001.
2. J. A. Mitchell, A. R. Aronson, J. G. Mork, L. C. Folk, S. M. Humphrey, and J. M. Ward. Gene indexing: characterization and analysis of NLM's GeneRIFs. *AMIA Annu Symp Proc*, pages 460–464, 2003.
3. M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, 25:25–29, May 2000.
4. GeneRIF. <http://www.ncbi.nlm.nih.gov/projects/GeneRIF/>.
5. M. A. Harris, J. Clark, A. Ireland, J. Lomax, M. Ashburner, R. Foulger, K. Eilbeck, S. Lewis, B. Marshall, C. Mungall, J. Richter, G. M. Rubin, J. A. Blake, C. Bult, M. Dolan, H. Drabkin, J. T. Eppig, D. P. Hill, L. Ni, M. Ringwald, R. Balakrishnan, J. M. Cherry, K. R. Christie, M. C. Costanzo, S. S. Dwight, S. Engel, D. G. Fisk, J. E. Hirschman, E. L. Hong, R. S. Nash, A. Sethuraman, C. L. Theesfeld, D. Botstein, K. Dolinski, B. Feierbach, T. Berardini, S. Mundodi, S. Y. Rhee, R. Apweiler, D. Barrell, E. Camon, E. Dimmer, V. Lee, R. Chisholm, P. Gaudet, W. Kibbe, R. Kishore, E. M. Schwarz, P. Sternberg, M. Gwinn, L. Hannick, J. Wortman, M. Berriman, V. Wood, N. de la Cruz, P. Tonellato, P. Jaiswal, T. Seigfried, and R. White. The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res.*, 32:D258–261, Jan 2004.
6. GONUTS. [http://gowiki.tamu.edu/wiki/index.php/Category:G0:0008217\(!_regulation_of_blood_pressure\)](http://gowiki.tamu.edu/wiki/index.php/Category:G0:0008217(!_regulation_of_blood_pressure)).
7. S. Carbon, A. Ireland, C. J. Mungall, S. Shu, B. Marshall, S. Lewis, A. Ireland, J. Lomax, S. Carbon, C. Mungall, B. Hitz, R. Balakrishnan, M. Dolan, V. Wood, E. Hong, and P. Gaudet. AmiGO: online access to ontology and annotation data. *Bioinformatics*, 25:288–289, Jan 2009.
8. M. Hearst. Untangling text data mining, 1999.

9. M. Chau, J. J. Xu, , and H. Chen. Extracting Meaningful Entities from Police Narrative Reports. *Proceedings of the National Conference for Digital Government Research, Los Angeles, California, USA*, pages 271–275, 2002.
10. K. B. Cohen and L. Hunter. Getting started in text mining. *PLoS Comput. Biol.*, 4:e20, Jan 2008.
11. C. Blaschke, M. A. Andrade, C. Ouzounis, and A. Valencia. Automatic extraction of biological information from scientific text: protein-protein interactions. *Proc Int Conf Intell Syst Mol Biol*, pages 60–67, 1999.
12. R. A. Fisher. On the interpretation of 2 from contingency tables, and the calculation of P. *Journal of the Royal Statistical Society 85 (1)*, pages 87–94, 1922.
13. I. Rivals, L. Personnaz, L. Taing, and M. C. Potier. Enrichment or depletion of a GO category within a class of genes: which test? *Bioinformatics*, 23:401–407, Feb 2007.
14. M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The KEGG resource for deciphering the genome. *Nucleic Acids Res.*, 32:D277–280, Jan 2004.
15. H. S. Leong and D. Kipling. Text-based over-representation analysis of microarray gene lists with annotation bias. *Nucleic Acids Res.*, 37:e79, Jun 2009.
16. R. Rubinstein and I. Simon. MILANO—custom annotation of microarray results using automatic literature searches. *BMC Bioinformatics*, 6:12, 2005.
17. N. H. Shah and N. V. Fedoroff. CLENCH: a program for calculating Cluster ENriCHment using the Gene Ontology. *Bioinformatics*, 20:1196–1197, May 2004.
18. G. Bindea, B. Mlecnik, H. Hackl, P. Charoentong, M. Tosolini, A. Kirilovsky, W. H. Fridman, F. Pages, Z. Trajanoski, and J. Galon. ClueGO: a Cytoscape plug-in to decipher functionally grouped gene ontology and pathway annotation networks. *Bioinformatics*, 25:1091–1093, Apr 2009.
19. d. a. W. Huang, B. T. Sherman, and R. A. Lempicki. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat Protoc*, 4:44–57, 2009.
20. V. Beisvag, F. K. Junge, H. Bergum, L. J?lsum, S. Lydersen, C. C. Gunther, H. Rammampiaro, M. Langaas, A. K. Sandvik, and A. Laegreid. GeneTools—application

- for functional annotation and statistical hypothesis testing. *BMC Bioinformatics*, 7:470, 2006.
21. G. F. Berriz, O. D. King, B. Bryant, C. Sander, and F. P. Roth. Characterizing gene sets with FuncAssociate. *Bioinformatics*, 19:2502–2504, Dec 2003.
 22. C. Henegar, J. Tordjman, V. Achard, D. Lacasa, I. Cremer, M. Guerre-Millo, C. Poitou, A. Basdevant, V. Stich, N. Viguerie, D. Langin, P. Bedossa, J. D. Zucker, and K. Clement. Adipose tissue transcriptomic signature highlights the pathological relevance of extracellular matrix in human obesity. *Genome Biol.*, 9:R14, 2008.
 23. Q. Zheng and X. J. Wang. GOEAST: a web-based software toolkit for Gene Ontology enrichment analysis. *Nucleic Acids Res.*, 36:W358–363, Jul 2008.
 24. S. Zhong, K. F. Storch, O. Lipan, M. C. Kao, C. J. Weitz, and W. H. Wong. GoSurfer: a graphical interactive tool for comparative analysis of large gene sets in Gene Ontology space. *Appl. Bioinformatics*, 3:261–264, 2004.
 25. E. Eden, R. Navon, I. Steinfeld, D. Lipson, and Z. Yakhini. GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists. *BMC Bioinformatics*, 10:48, 2009.
 26. T. G. Soldatos, S. I. O’Donoghue, V. P. Satagopam, L. J. Jensen, N. P. Brown, A. Barbosa-Silva, and R. Schneider. Martini: using literature keywords to compare gene sets. *Nucleic Acids Res.*, 38:26–38, Jan 2010.
 27. S. Grossmann, S. Bauer, P. N. Robinson, and M. Vingron. Improved detection of overrepresentation of Gene-Ontology annotations with parent child analysis. *Bioinformatics*, 23:3024–3031, Nov 2007.
 28. S. Bauer, S. Grossmann, M. Vingron, and P. N. Robinson. Ontologizer 2.0—a multifunctional tool for GO term enrichment analysis and data exploration. *Bioinformatics*, 24:1650–1651, Jul 2008.
 29. P. Khatri and S. Draghici. Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics*, 21:3587–3595, Sep 2005.
 30. Hypergeometric Distribution on Wikipedia. http://en.wikipedia.org/wiki/Hypergeometric_distribution.

31. Fernanda B. Viégas and Martin Wattenberg. Tag clouds and the case for vernacular visualization. *interactions*, 15(4):49–52, 2008.
32. Popular tags on Flickr.com. <http://http://www.flickr.com/photos/tags/>.
33. Kai Bielenberg and Marc Zacher. Groups in Social Software: Utilizing Tagging to Integrate Individual Contexts for Social Navigation. August 2005.
34. Byron Y. L. Kuo, Thomas Hentrich, Benjamin, and Mark D. Wilkinson. Tag clouds for summarizing web search results. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 1203–1204, New York, NY, USA, 2007. ACM Press.
35. Jonathan Feinberg. Wordle. <http://www.wordle.net/>.
36. James Sinclair and Michael Cardew-Hall. The folksonomy tag cloud: when is it useful? *J. Inf. Sci.*, 34(1):15–21, February 2008.
37. Architecture Journal Profile: Scott Guthrie. January 2007.
38. Douglas Coupland. *Microserfs*. Regan Books, HarperCollins, 1995.
39. Most searched topic in tag cloud on NYTimes.com. <http://www.nytimes.com/most-popular-searched?format=tagcloud&period=1>.
40. J. Desai, J. M. Flatow, J. Song, L. J. Zhu, P. Du, C. C. Huang, H. Lu, S. M. Lin, and W. A. Kibbe. Visual presentation as a welcome alternative to textual presentation of gene annotation information. *Adv. Exp. Med. Biol.*, 680:709–715, 2010.
41. C.J. van Rijsbergen, S.E. Robertson, and M.F. Porter. New models in probabilistic information retrieval. *British Library Research and Development Report*, no. 5587, 1980.
42. George A. Miller. WordNet: a lexical database for English. *Commun. ACM*, 38(11):39–41, November 1995.
43. Thomas Boutell. GD Graphics Library. www.libgd.org/.

44. Leland H. Hartwell, John J. Hopfield, Stanislas Leibler, and Andrew W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–C52, December 1999.
45. R. Homayouni, K. Heinrich, L. Wei, and M. W. Berry. Gene clustering by latent semantic indexing of MEDLINE abstracts. *Bioinformatics*, 21:104–115, Jan 2005.
46. Gennadi V. Glinsky, Anna B. Glinskii, Andrew J. Stephenson, Robert M. Hoffman, and William L. Gerald. Gene expression profiling predicts clinical outcome of prostate cancer. *Journal of Clinical Investigation*, 113(6):913–923, March 2004.
47. P. G. Febbo, M. G. Mulligan, D. A. Slonina, K. Stegmaier, D. Di Vizio, P. R. Martinez, M. Loda, and S. C. Taylor. Literature Lab: a method of automated literature interrogation to infer biology from microarray analysis. *BMC Genomics*, 8:461, 2007.
48. G. Leroy, H. Chen, and J. D. Martinez. A shallow parser based on closed-class words to capture relations in biomedical text. *J Biomed Inform*, 36:145–158, Jun 2003.
49. almaKnowledge Server 2. <http://www.bioalma.com/aks2/index.php>.
50. NCBI. <http://www.ncbi.nlm.nih.gov/>.
51. Tim O'Reilly. What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software. *Social Science Research Network Working Paper Series*, August 2007.

VITA

Jing Wen is a master student in bioinformatics program in University of Illinois of Chicago with advisor Dr. Hui Lu. She got her bachelor degree from Zhejiang University in China, 2006, majored in Software Engineering. Her main research is focusing on the application of novel gene annotation and visualization tools.

Now she is working as a QA analyst in Greenline Financial Technologies. Before coming to UIC, she was a QA engineer in Hewlett-Packard, Shanghai, China.

Publications:

- **Jing Wen**, Xishu Wang, Warren Kibbe, Simon Lin, Hui Lu "Visual Annotation of the Gene Database", IEEE EMBC 2009