To the Graduate Council:

I am submitting herewith a thesis written by Kevin Erich Heinrich entitled “Finding Functional Gene Relationships Using the Semantic Gene Organizer (SGO).” I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Science.

Michael W. Berry
Major Professor

We have read this thesis
and recommend its acceptance:

Ramin Homayouni

Jens Gregor

Accepted for the Council:

Anne Mayhew
Vice Chancellor and
Dean of Graduate Studies

(Original signatures are on file with official student records.)

Finding Functional Gene Relationships

Using the Semantic Gene Organizer

(SGO)

A Thesis

Presented for the

Master of Science Degree

The University of Tennessee, Knoxville

Kevin Erich Heinrich

August 2004

Acknowledgements

I would like to thank Dr. Michael Berry for his unwavering support over the years. If not for him, this thesis and the work behind it would have never happened. I am also indebted to Dr. Ramin Homayouni at the University of Tennessee Health Science Center for his enthusiasm and patience with the biological aspects of this work. Also at the Health Science Center, I would like to express my gratitude to Lai Wei and Lijing Xu, for their invaluable contributions and explanations. In addition, I would like to thank Dr. Jens Gregor for his support of my graduate career by serving on my thesis committee and Murray Browne for his help putting together this thesis. I would like to acknowledge Justin Giles, for explaining the LSI software to me. Last but not least, I would like to recognize and thank my family and my fiancé, Erica Chisholm, for giving me something to work for and for putting up with me while I was getting my work done.

Abstract

Understanding functional gene relationships is a major challenge in bioninformatics and computational biology. Currently, many approaches extract gene relationships via term co-occurrence models from the biomedical literature. Unfortunately, however, many genes that are experimentally identified to be related have not been previously studied together. As a result, many automated models fail to help researchers understand the nature of the relationships. In this work, the particular schema used to mine genomic data is called Latent Semantic Indexing (LSI). LSI performs a singular-value decomposition (SVD) to produce a low-rank approximation of the data set. Effectively, it allows queries to be interpreted in a more concept-based space and can allow for gene relationships to be discovered that would ordinarily be overlooked by other models.

Contents

1 Introduction **1**

2 Previous Work **3**

3 Information Retrieval **6**

3.1 Vector Space Model 6

3.1.1 Data Representation 7

3.1.2 Term Weighting 7

3.1.3 Query and Similarity 9

3.2 Latent Semantic Indexing 10

3.3 Evaluation Measures 12

4 Gene Document Construction **14**

4.1 Literature Sources 14

4.1.1 MEDLINE 15

4.1.2 LocusLink 16

4.2 Method of Document Construction 17

4.3 Test Data Set 18

5 Use and Evaluation of SGO **20**

5.1 User Interface 20

5.2 Query Types 22

5.3 Trees 23

5.4 Coding Issues 28

5.5 Results 29

6 Conclusion **36**

Bibliography **39**

Appendix **45**

A Genes Used in Test Data Set **46**

Vita **50**

List of Tables

5.1 Ranks for genes directly and indirectly associated with the *Reelin* signaling pathway. 30

5.2 SGO's AP performance for different keyword queries. 30

A.1 Genes in the test data set. 47

A.2 Classifications associated with each gene. 48

A.3 Number of citations associated with LocusLink entries for each gene sorted by total number of citations. 49

List of Figures

4.1 Schematic of the *Reelin* signaling pathway. 19

5.1 Screenshot of the SGO interface. 21

5.2 Screenshot of a gene document and return list with latent matches (*Lrp8*) enabled. 24

5.3 Screenshot of a hierarchical tree produced by SGO. 26

5.4 Screenshot of a nodal tree produced by SGO. 27

5.5 Interpolated precision values for identifying primary genes at the decile recall ranges. 31

5.6 Interpolated precision values for identifying primary and secondary genes at the decile recall ranges. 32

5.7 Effect on AP of decreasing abstract representation of the five primary *Reelin*-related gene documents. 34

Chapter 1

Introduction

Recent technological advances in genomics, proteomics, and related fields have enabled researchers to generate vast amounts of biological data. Techniques such as DNA microarray analysis are effective methods that help reveal functional gene relationships. Unfortunately, however, such techniques are often time-consuming and expensive. Uncovering new gene relationships is a combinatorially difficult problem that requires some direction to make it tractable.

One approach is simply to scan the literature for future direction. This approach, however, is infeasible. At the current rate of literature growth, a researcher would be required to scan over 130 journals and read 27 papers daily to stay current with information about breast cancer [HHW$^+$03]. The literature concerning other diseases exhibit similar unmanageable growth trends.

The Semantic Gene Organizer (SGO)[1] has been developed as a utility to help biological researchers more quickly identify and understand functional gene relationships. Currently, to understand functional gene relationships investigators must manually extract biological literature from several different databases. Most of these databases match queries in a simple term co-occurrence framework that does not retrieve potentially relevant documents in a satisfactory manner. SGO is based on Latent Semantic Indexing (LSI) [DDF+90], which can help identify latent structural similarities in the literature. As such, SGO can help uncover gene-gene and gene-keyword relationships with good accuracy.

Given the nature of text mining tools, SGO is not meant to replace biological techniques, but rather to enhance them. SGO is designed as an intelligent guide for future research as well as a verifier for new results. In no way does a result gleaned from SGO have more biological significance than one gained from the lab.

SGO has performed well on a few sample data sets. To interpret the results of SGO, however, the user must understand existing information retrieval (IR) techniques and what SGO is trying to uncover. Existing and related techniques are covered in Chapter 2. The information retrieval techniques used by SGO are introduced in Chapter 3, while the background biological information is covered in Chapter 4. SGO's results are examined in Chapter 5, and the possible future directions for SGO development are discussed in Chapter 6.

[1] SGO is available for use at http://shad.cs.utk.edu/sgo/.

Chapter 2

Previous Work

SGO and its interface are inspired by a previous and related work called the Semantic Conference Organizer (SCO) that was built to assist conference organizers with session building [HBDV03]. Seeing the effectiveness of SCO with clustering conference abstracts, the idea of clustering genes via similar methods arose.

Since biomedical literature has been growing at a high rate, data mining tools have been developed to help investigators extract meaningful information about genes of interest. [JLKH01] has developed PubGene, a literature network that assigns a functional association between two genes if there is a co-occurrence between gene symbols in MEDLINE (discussed in Chapter 4) abstracts. This network is an attempt to identify functionally related genes based on the published literature. Once this network has been created, graph theoretic methods to identify communities of related genes can be applied. The resulting partition will hopefully link abstracts on some common term co-occurrence that has spe-

cial meaning, like gene function [WH04]. Once groups of functionally related genes are identified, natural language processing techniques can be employed to further extract the nature of the relationships between genes [YM02].

These methods, however, have one underlying assumption—related genes will have term co-occurrence somewhere in the literature to produce an association between them. One of the problems of genomic data is the high occurrence of gene aliases. Depending on the time the literature was published, one of several aliases can be used to identify a gene. As such, many term co-occurrence models will fail to consider associations between genes that relate gene aliases rather than official gene names.

This scenario can be extended to where literature can identify relationships between genes without even mentioning a gene or any of its aliases. If a gene has certain functional attributes, the literature about it will generally exhibit the same fundamental structure. As a result, methods that base their similarity measures not on the number of simple term co-occurrences but on the underlying document word usage patterns will be more likely to find previously unknown relationships.

The information retrieval technique described is Latent Semantic Indexing (LSI) and is discussed in detail in Section 3.2. This technique is just one of many IR methods that falls within the broad category of vector space models (discussed in Chapter 3). Set theoretic and probabilistic models are the two other classifications that are typically assigned to IR techniques. Set theoretic models, in their simplicity, are already used in most biological contexts. That is, a document is retrieved if an index term occurs in it. Most of the term

co-occurrence models and their derivatives fall into this category. Probabilistic models are also being researched. For example, [MSY03] is refining a hidden Markov model that attempts to extract useful noun phrases from biomedical literature. This model will help identify related genes and refine existing ontologies used for classification.

Chapter 3

Information Retrieval

As the amount of data stored on the Web increases, efficient techniques to navigate and access that data must be explored. The field of information retrieval (IR) has been well-researched; however, its application in other disciplines is just recently being developed. This chapter contains an overview of the general vector space model. Additionally, a dimension-reduction technique, Latent Semantic Indexing (LSI), is explained, and the standard evaluation measures, precision and recall, are discussed.

3.1 Vector Space Model

The *vector space model* is one of many types of information retrieval techniques. Vector space models assume that the meaning of a document can be derived from the words that comprise it [Let96]. In fact, many popular vector space models only consider the distribution of some meaningful words while ignoring the order and proximity with which those

words occur.

3.1.1 Data Representation

Treating each document as a *bag of words*, the text must first be parsed into keywords or *tokens*. Such parsing typically ignores capitalization and most non-alphanumeric characters. Also, articles and other non-distinguishing words are removed. The resulting view of each document is a list of "meaningful" words that represent it.

Given a dictionary of m tokens, a document is represented by a vector of length m. Thus, document j can be represented by $d_j = (w_{1j}, \ldots, w_{mj})$, where w_{kj} is the weight associated with term k in document j. The n documents in a collection comprise the columns of an $m \times n$ *term-by-document matrix* $A = [w_{ij}]$. Conversely, the rows of A represent m term vectors that show the correspondence between each term and the documents in which it occurs.

3.1.2 Term Weighting

Each matrix entry w_{ij} is a weighted value that represents the occurrence of token i in document j and can be computed as

$$w_{ij} = l_{ij} g_i d_j,$$

where l_{ij} denotes the local weight of term i in document j, g_i is the global weight of term i, and d_j is a document normalization factor that can normalize the columns of A [BB99].

Normalizing the columns ensures that each column has a norm of unit length and helps eliminate the influence that document size can have on some weighting schemes.

One of the simplest weighting schemes is *term frequency* or

$$l_{ij} = f_{ij},$$

where f_{ij} denotes the frequency with which term i occurs in document j. Since, under this scheme, rankings (discussed in Section 3.1.3) are biased toward larger documents and more common terms, an *inverse document frequency* global weighting factor such as

$$g_i = \frac{\sum_j f_{ij}}{\sum_j \chi(f_{ij})}$$

can be introduced, where $\chi(f_{ij})$ is a binary weight that equals one if f_{ij} is nonzero and zero otherwise [BYRN99]. This weighting scheme, called *term frequency, inverse document frequency* or *tf-idf* has an alternative scheme (referred to as *tf-idf2*) where the global weighting factor is changed to

$$g_i = \log_2 \frac{n}{\sum_j \chi(f_{ij})} + 1.$$

Another weighting scheme designed to give distinguishing tokens higher weight is *log-entropy,* given by

$$l_{ij} \quad = \quad \log\,(1 + f_{ij}),$$

$$g_i \;=\; 1 + \left(\frac{\sum\limits_j (p_{ij} \log p_{ij})}{\log n} \right),$$

where $p_{ij} = f_{ij} / \sum\limits_j f_{ij}$ represents the probability of term i occurring in document j [BB99].

3.1.3 Query and Similarity

Once term weights are computed and the term-by-document matrix is constructed, a query can be represented by a *pseudo*-document vector, $q = (g_1, g_2, \ldots, g_m)$. Since queries are commonly shorter than documents and rarely contain repetitive terms, the local weight components are usually ignored. However, query vectors can be modified to include local weights if queries are sufficiently large. Also, as with document vectors, query vectors can be normalized.

Once a query is constructed, computing the similarity of document j with respect to the query q is accomplished by computing the cosine of the angle between the two vectors. That is,

$$sim(q, d_j) = \cos\theta_j = \frac{\vec{q} \bullet \vec{d_j}}{|\vec{q}| \times |\vec{d_j}|} = \frac{\sum\limits_{k=1}^{m} g_k w_{kj}}{\sqrt{\sum\limits_{k=1}^{m} g_k^2} \sqrt{\sum\limits_{k=1}^{m} w_{kj}^2}}.$$

This computation is performed for each of the n documents and the result is sorted to produce a ranking of the documents with respect to the query q [BYRN99].[1] The actual value of the similarity score carries little value—the real relevance information is gained by

[1]Cosine is one of many similarity measures. For more similarity measures, the reader is directed to [SM83, ORRW81, Cho99].

the rank of a document with respect to another document. In practice, however, similarity scores are thresholded, and only documents with a similarity score above that threshold are presented back to the user [BB99].

3.2 Latent Semantic Indexing

Two major obstacles of almost all information retrieval models are *synonymy* and *polysemy*. Synonymy refers to different words having the same meaning, while polysemy refers to the same word having a different meaning depending on its context. In both cases, a simple vector space model does not attempt to handle these anomalies. A variant of the vector space model called Latent Semantic Indexing, however, does. LSI attempts to discern global usage patterns in vocabulary to determine the hidden or latent structure of the documents; in effect, LSI is an attempt to overcome the problems of synonymy and polysemy [Jia97]. [DDF$^+$90] argues that LSI overcomes the problem of synonymy well, while it has marginal success dealing with polysemy.

After the term-by-document matrix, A, has been computed, a truncated singular value decomposition is performed to generate three factor matrices

$$A = U\Sigma V^T,$$

where U is the $m \times r$ matrix of eigenvectors of AA^T, Σ is the $r \times r$ diagonal matrix of the r singular values of A, V^T is the $r \times n$ matrix of eigenvectors of A^TA, and r is the rank of

the A [GL96]. A rank-s approximation A_s of A can be computed by truncating each of the factor matrices to the first s columns. That is,

$$A_s = U_s \Sigma_s V_s^T.$$

Document-to-document similarity is then given by

$$A_s^T A_s = (V_s \Sigma_s)(V_s \Sigma_s)^T.$$

Queries in LSI must be projected into the appropriate low-rank approximation space. Given the initial *pseudo*-document q_0 of associated term weights, a projected query, q, is given by

$$q = q_0^T U_s \Sigma_s^{-1},$$

where U_s and Σ_s denote the first s columns of U and Σ, respectively [BB99]. If scaled document vectors $s_j = \Sigma_k V_k^T e_j$, where e_j denotes the jth column of the $n \times n$ identity matrix, are calculated, similarity between each s_j and q can be computed by

$$sim(s_j, q) = \cos\theta_j = \frac{s_j^T (U_k^T q)}{||s_j||_2 ||q||_2}, j = 1, 2, \ldots, n.$$

LSI's end effect is to project the term-by-document matrix into a lower-dimensional space, thereby forcing queries and documents to be interpreted in a more conceptual manner rather than a literal one by explicitly modeling the interrelationships among terms

[BDO95]. Choice of the number of factors or dimensions of the factorization determines the conceptual level at which documents are compared, with more factors tending to a more literal comparison. Because of this, LSI can find similarities between documents that have no term co-occurrence, and many of the negative effects of *noise* are reduced [LB97]. The optimal choice of factors is an open question, and often the choice of dimensions is an empirical tradeoff between accuracy and problem size or storage [BDJ99].

3.3 Evaluation Measures

Information retrieval systems are often evaluated by the standard measures of precision and recall. *Precision* is the ratio of relevant returned documents to the total number of returned documents. *Recall* refers to the ratio of relevant returned documents to the total number of relevant documents [BYRN99].

Often, graphs of a system's performance are given, measuring precision at varying levels of recall. To condense a system's performance into one value, the concept of *average precision (AP)* is introduced. The n-point (interpolated) average precision is given by

$$P_{av} = \frac{1}{n} \sum_{i=0}^{n-1} \tilde{P}(\frac{i}{n-1}),$$

where $\tilde{P}(x)$ is the maximum precision up to the ith document. That is,

$$\tilde{P}(x) = \max \frac{r_i}{i}, i = 1, 2, \ldots, n,$$

where r_i is the number of relevant documents up to and including the ith position of the returned list of documents [BB99]. In general, precision is observed at the endpoints of the ten decile ranges (i.e., at recall levels of 0%, 10%, 20%, etc.) to form a standard 11-point average precision value.

Chapter 4

Gene Document Construction

Information retrieval techniques assume that meaningful document collections exist and can be easily retrieved. IR methods do not handle the actual construction of the document collections. In the case of SGO, the construction of documents to represent genes is a nontrivial process. This chapter describes the method of gene document construction used to create a test data set, and gives a brief description of the genes in the test data set.

4.1 Literature Sources

Ultimately, genes affect many diseases. Currently, the entire human genome as well as several key model organisms have recently been sequenced and annotated. While gene research is a relatively new field, documentation for disease treatment is much more extensive. Furthermore, researchers are interested in linking known gene information with existing medical literature, which is a non-trivial task. There are many databases avail-

able online concerning both medical and gene information; however, SGO primarily uses MEDLINE and LocusLink.

4.1.1 MEDLINE

The United States National Library of Medicine[1] (NLM) has a bibliographic database called MEDLINE (Medical Literature, Analysis, and Retrieval System Online) that contains over fourteen million references to biological journal articles with a concentration in medicine. MEDLINE covers topics related to basic biomedical research, clinical sciences, and life sciences that concern biomedical practitioners. Citations span over 4,600 journals worldwide, while approximately half of the cited articles are published in the United States. PubMed and NLM Gateway are publicly available tools to search MEDLINE.

All citations in MEDLINE start with indexing year 1966. Approximately 109,000 citations from 1953-1965 are in OLDMEDLINE and can be searched and retrieved when using NLM Gateway. PubMed, however, does not have this feature and only includes citations from 1966 onward.

Almost a half million completed references are added to MEDLINE yearly, with about 2,000 citations added daily six days a week. Each citation is manually indexed with Medical Subject Headings (MeSH) terms, a controlled vocabulary provided by NLM. MeSH terms are organized in a hierarchical fashion.

PubMed retrieves articles from MEDLINE based on any combination of attributes rang-

[1]NLM, MEDLINE, PubMed, and MeSH are all registered trademarks.

ing from MeSH terms to simple keyword and keyword phrases. PubMed also provides access to other selected life science journals not in MEDLINE, although MEDLINE makes up the vast majority of PubMed's coverage [NLM].

Not all abstracts in MEDLINE relate to genes. Since MEDLINE contains literature that spans the last forty years, some of the genomic nomenclature has changed over time. As a result, simply querying PubMed for a gene name will not retrieve the expected amount of abstracts from MEDLINE. Querying with gene aliases, however, often gathers too many abstracts from MEDLINE that usually have very little to do with the gene in question. As a result, ad hoc methods that refine PubMed alias queries with certain keywords to produce an appropriate number and type of abstracts is being researched and is discussed in Section 5.2.

4.1.2 LocusLink

The National Center for Biotechnology Information (NCBI) is a division of NLM that focuses more on molecular biology. Specifically, NCBI provides access to a human-curated gene-centric database called LocusLink in addition to other databases and services. LocusLink is a single query interface to a comprehensive directory for genes and gene reference sequences for key genomes ranging from sea urchins to fruit flies to humans. LocusLink provides links to other public databases such as MapViewer,[2] OMIM (Online

[2]http://www.ncbi.nlm.nih.gov/mapview/

Mendelian Inheritance in Man),[3] UniGene,[4] Gene Ontology (GO) Annotation,[5] and the Genome Browser.[6] LocusLink provides links to related records in PubMed and other citations that are deemed relevant to a specific sequence area or gene. Also, whenever available, LocusLink displays other information such as official gene name and symbols as well as other known aliases [PKSM00]. In addition, annotators provide a RefSeq Summary of gene function and links to key MEDLINE citations relevant to each gene.

LocusLink, however, does not cover all relevant citations for each gene. In fact, LocusLink only links to a representative few. Currently, there are 22,661 abstracts associated with 38,504 Human LocusLink entries. The 70,413 and 27,393 Mouse and Rat LocusLink entries have 27,720 and 7,797 associated abstracts, respectively.

4.2 Method of Document Construction

Gene document construction is a nontrivial process that has not yet been perfected. One approach would be to use a small collection of highly relevant abstracts for given genes that have been assigned by professional curators at LocusLink. This approach would, in theory, accurately represent genes so known relationships could easily be identified, but the probability of finding hidden relationships is decreased.

A text document to represent a gene or *gene document* is created by concatenating all titles and abstracts of MEDLINE citations cross-referenced in the Mouse, Rat, and Human

[3] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM
[4] http://www.ncbi.nlm.nih.gov/UniGene/
[5] http://www.ebi.ac.uk/GOA/
[6] http://genome.ucsc.edu

LocusLink entries for that gene. It is important to note that sequencing abstracts were included in each gene document, and that the LocusLink references are not comprehensive. As such, noise is introduced and the recall of all abstracts associated with a gene is not guaranteed.

4.3 Test Data Set

A test data set consisting of 50 genes was created to test SGO. The genes occurred in at least one of three broad functional categories: development, Alzheimer's disease, and cancer biology. The genes in the test data set are listed in Table A.1 and their classifications are given in Table A.2. The number of LocusLink citations for each gene in the test data set is given in Table A.3.

The *Reelin* signaling pathway was used as a basis for evaluation and is depicted in Figure 4.1 [HHWB04]. *Reelin* binds directly to the lipoprotein receptors *Vldlr* and *Apoer2* and induces tyrosine phosphorylation of the cytoplasmic adapter protein *Dab1* by fyn tyrosine kinase. *Dab1* binds to amyloid precursor family proteins (APP) among other proteins and is phosphorylated on ser residues by *cyclin dependent protein kinase 5 (Cdk5)*. Disruption of the *Cdk5* gene or its activator *P35* causes brain structure abnormalities similar to those observed in *reeler* mice [KT98, DT01], and accumulating evidence suggests that some components of the *Reelin* signaling pathway are associated with Alzheimer's disease [HRSC99, HHWB04].

For evaluation purposes, the genes *Reln*, *Vldlr*, *Lrp8*, *Dab1*, and *Fyn* are assumed to

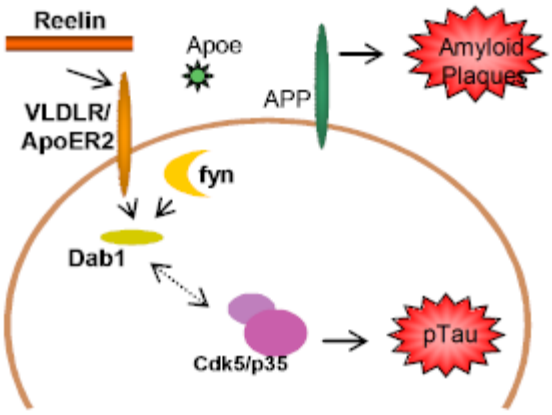

Figure 4.1: Schematic of the *Reelin* signaling pathway.

be directly related to *Reln*. Genes indirectly associated with *Reln* are assumed to be *Cdk5*, *Apoe*, *Src*, *Mapt*, *App*, *Aplp1*, and *Aplp2*. Biological justification for these assumptions are given in detail in [HHWB04].

Chapter 5

Use and Evaluation of SGO

SGO is built upon an existing information retrieval (IR) tool and must, in the end, present results that are meaningful to biologists. This chapter explains SGO in detail including its interface, features, and strengths and weaknesses.

Once a gene document collection has been created and extraneous tags filtered, only titles and abstracts should remain. The resulting documents are then parsed via General Text Parser (GTP), which also performs an SVD and stores the resulting matrices [GWB03]. When GTP finishes processing the document collection, users are able to query the collection via the web interface.

5.1 User Interface

A screenshot of the user interface is shown in Figure 5.1. During design, much thought was given to making the interface as intuitive for biologists as possible. From the start page,

Figure 5.1: Screenshot of the SGO interface.

the user is able to recall previous saved searches, browse genes in a collection, or query a collection. Demos are provided to illustrate different query types.

Given the research-oriented nature of the user base, the need to save and recall previous query sessions is evident. Storing query sessions gives SGO the flexibility to store session-specific information as well as the ability to incorporate future extensions more easily.

5.2 Query Types

A user can query either by gene or by keyword. To query by gene, a list of genes must be entered by their UniGene ID, LocusLink ID, OMIM ID, or GenBank Accession Number. Since biologists usually have lists of these identifiers readily available for the genes of their interest, this query method should be straightforward. When a gene query is performed, the entire gene document is used as the query vector for comparison. In essence, the return list will show gene-gene relationships as dictated by the literature.

On the other hand, keyword queries can be performed in a manner similar to most any web search engine. Each query is a few keywords that are used to create a *pseudo*-document that is then compared against all genes in the chosen database. In effect, the return list will show gene-keyword relationships. If keywords are chosen wisely, gene-disease, gene-function, and other novel gene relationships can be exposed.

In order to exploit the power of LSI, users are also able to choose the number of factors with which to query. As discussed in Section 3.2, the number of factors used dictates the semantic level at which queries are compared. As a result, users are able to compare queries

at the broadest conceptual level (2 factors) up to a significant level of detail (anywhere up to the number of documents in the collection). In practice, LSI is most effective at querying with approximately 300 dimensions for large collections [LLD04]. As a result, a maximum of 500 dimensions is enforced for any user query.

To demonstrate the effectiveness of SGO, the user can choose to show "latent" matches, or gene documents that contain none of the query words. This option is only available with keyword queries and helps users quickly identify possible previously-unknown relationships that exist in the semantic structure of the literature.

A sample query is shown in Figure 5.2. After a query has been performed, the user can view the return list by clicking on the query. Each gene will be listed in rank order along with its cosine similarity value. If applicable, latent matches will be highlighted in red. For quick assessment, links to the LocusLink or OMIM entries are provided when appropriate. Clicking on a gene returns the gene document in the upper left frame along with the document's top 100 terms and their corresponding global weights. This information can be used for possible PubMed query refinement and is discussed in Chapter 4.

5.3 Trees

In addition to simple ranked lists, SGO provides other visualization techniques. To do so, SGO must first modify its output to be used by other algorithms. A self-similarity matrix of the gene documents in a collection can be built by concatenating gene document queries

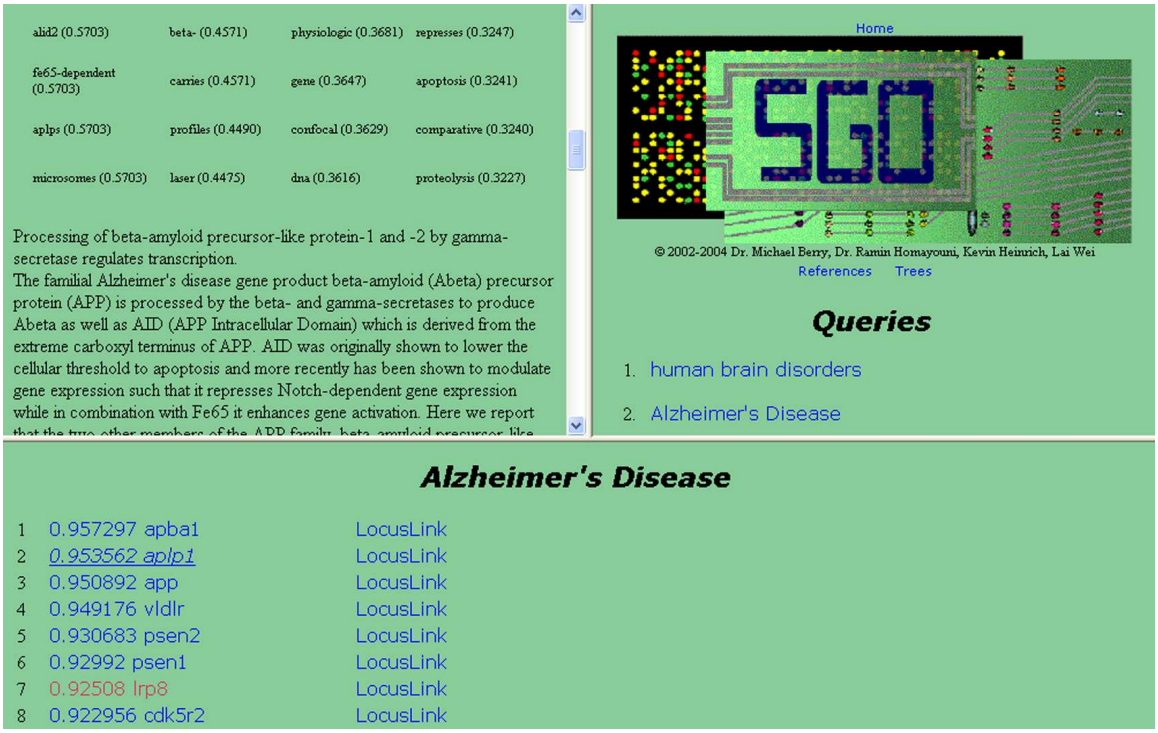

Figure 5.2: Screenshot of a gene document and return list with latent matches (*Lrp8*) enabled.

into a matrix. A *distance matrix*, $D = [d_{ij}]$, can be constructed such that

$$d_{ij} = 1 - \cos\theta_{ij},$$

where $\cos\theta_{ij}$ is the cosine similarity between documents i and j. This distance matrix can then be used by any number of tree-building algorithms to create hierarchical trees. Using the PHYLIP implementation[1] of the Fitch-Margoliash method [FM67], SGO produced the hierarchical tree shown in Figure 5.3 on the 50 gene test data set. The distance matrix for the tree shown was computed using 25 factors to compute similarity, and the tree was displayed with the ATV Java applet.[2]

By employing *thresholds* to the self-similarity matrix, a graph $G = (V, E)$ can be constructed where V is the set of vertices or genes and E is the set of edges. In the case of SGO, an edge $\{i, j\}$ is drawn between gene i and j if the similarity between them is higher than a predefined threshold. Figure 5.4 shows the output of a Java applet[3] that displays a graph or *Nodal tree* that was built with a threshold value of 0.7 on the 50 gene test data set. This applet is interactive—the user can drag a gene to see what genes cluster to and away from it. Such a graph structure, although simplistic, helps the user quickly identify the overall structure and relationships between all the genes in the collection.

[1] http://evolution.genetics.washington.edu/phylip.html
[2] http://www.genetics.wustl.edu/eddy/atv/
[3] modified from http://java.sun.com/applets/jdk/1.0/demo/GraphLayout/

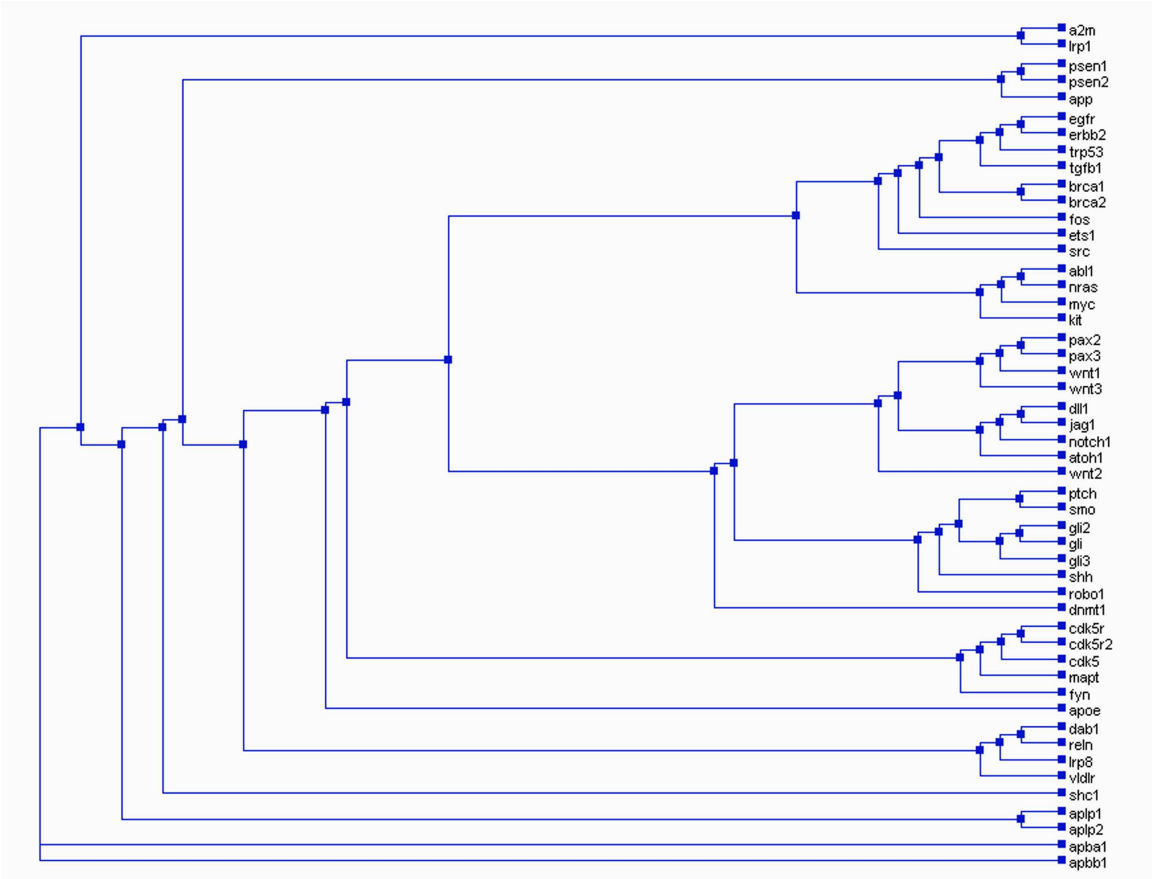

Figure 5.3: Screenshot of a hierarchical tree produced by SGO.

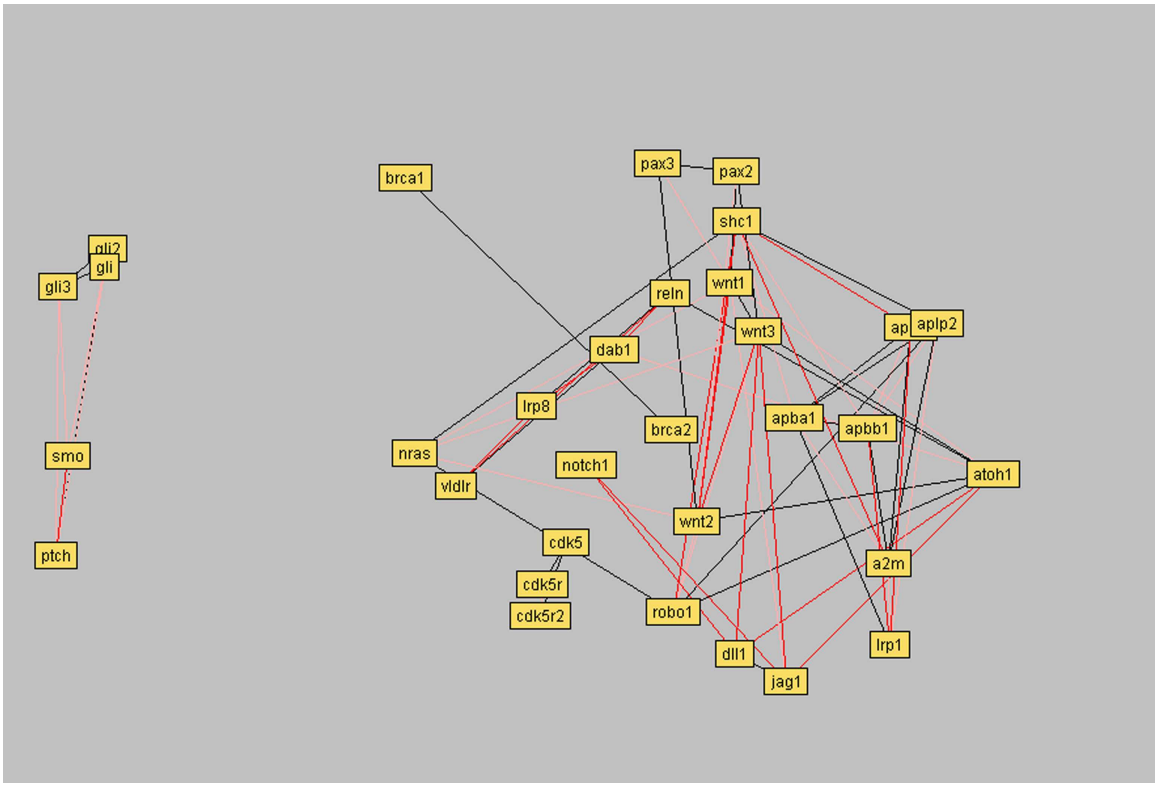

Figure 5.4: Screenshot of a nodal tree produced by SGO.

5.4 Coding Issues

Since SGO has an interactive web interface, server response time is a key issue. As a result, all actual querying of the database is performed when the user clicks on a query rather than when the query is submitted. Along the same lines, both the hierarchical and nodal trees are static for each collection. Since quality tree construction is a slow process, trees are built offline and presented after the collection is available. It is important to note, however, that SGO has a *scalable* interface—the user is able to query a 50 gene collection or a collection of thousands of genes with little noticeable difference in response time.

The 50 test gene collection contained 19,789 terms. If a global and document threshold of 1 is applied—that is, if terms that occur only once across the document collection or only once in a document are discarded—the number of terms is reduced to 8,754. With no threshold, the matrices produced by GTP are approximately 7 Mb in size; with thresholds, the matrices are approximately half that size. Simple vector space models produce matrices that are about 5.5 Mb in size. If the document collection is scaled up to include 20,856 Human LocusLink abstracts (85,999 terms), the storage space required is slightly less than 150 Mb. Likewise, a collection of 4,956 Rat LocusLink abstracts (28,905 terms) requires approximately 32 Mb.[4]

[4]Both LocusLink collections only have a 100-dimension factorization since they were created before the 500-factor query limit was decided. Expanding these collections to 500 dimensions would add storage requirements but would still easily be within a manageable size.

5.5 Results

SGO performance was evaluated against existing methods for finding gene relationships in literature, and LSI was compared against standard tf-idf vector space models. Table 5.1 shows the performance of SGO using various models to identify genes directly and indirectly associated with the *Reelin* signaling pathway. Although the LSI methods are slightly outperformed by the vector space methods for gene document queries, the LSI methods are more robust in that they are able to identify relationships for small queries. As collections scale up, this discrepancy will become more noticeable since simple vector space models require term co-occurrence to produce a non-zero similarity. Along the same lines, querying either PubMed or LocusLink for co-citations between genes produces results similar to the vector space models for keyword queries for the same reasons. Thus, LSI is able to rank all twelve genes related to *Reelin*, while other intuitive methods fail.

Genes from Gene Ontology (GO) classifications and genes known to be associated with certain human diseases were retrieved from the test data set using several models, and SGO's average precision for each case is presented in Table 5.2.

One advantage of SGO's interface is that it allows the user to specify the number of factors with which to query. This, in effect, determines the scope of the semantic space with which genes are compared. Figures 5.5 and 5.6 show the precision-recall graphs for SGO identifying the five direct and twelve indirect genes associated with *Reelin* signaling, respectively. Both graphs compare the effect of querying with 5, 25, and 50 factors for the keyword "Reelin," since querying by gene would skew the results in favor of *Reelin*.

Table 5.1: Ranks for genes directly and indirectly associated with the *Reelin* signaling pathway.

Gene	LSI 25 factors		LSI 50 factors		tf-idf		tf-idf2	
	Gene	Keyword	Gene	Keyword	Gene	Keyword	Gene	Keyword
RELN	1	2	1	3	1	1	1	1
DAB1	2	1	2	1	2	2	2	2
LRP8	3	3	3	2	3	3	3	3
VLDLR	4	4	4	4	4	4	4	4
FYN	41	34	24	47	14	-	29	-
CDK5	13	8	5	6	29	5	6	5
APOE	22	25	9	34	43	-	9	-
SRC	45	42	33	44	15	-	19	-
MAPT	18	41	7	48	47	-	26	-
APP	20	40	8	16	24	-	8	-
APLP1	10	14	45	30	10	-	18	-
APLP2	12	16	38	11	18	-	10	-
AP	0.634	0.593	0.728	0.617	0.604		0.757	

Table 5.2: SGO's AP performance for different keyword queries.

Query	Relevant Genes	LSI-25	LSI-50	tf-idf	tf-idf2
GO Classifications					
apoptosis	7	0.34	0.45	-	-
axon guidance	1	0.10	1.00	1.00	1.00
cell fate	2	0.59	0.64	0.22	0.62
kinase	8	0.72	0.80	0.93	0.97
neurogenesis	10	0.27	0.37	-	-
patterning	5	0.71	0.68	0.68	0.75
transcription	10	0.40	0.75	0.79	0.83
tyrosine kinase	3	0.19	0.30	0.27	0.38
Human Disease					
Alzheimer Disease	8	0.72	0.70	0.85	0.78
Breast Cancer	3	0.75	0.85	0.60	0.91
Lissencephaly	1	1.00	1.00	1.00	1.00

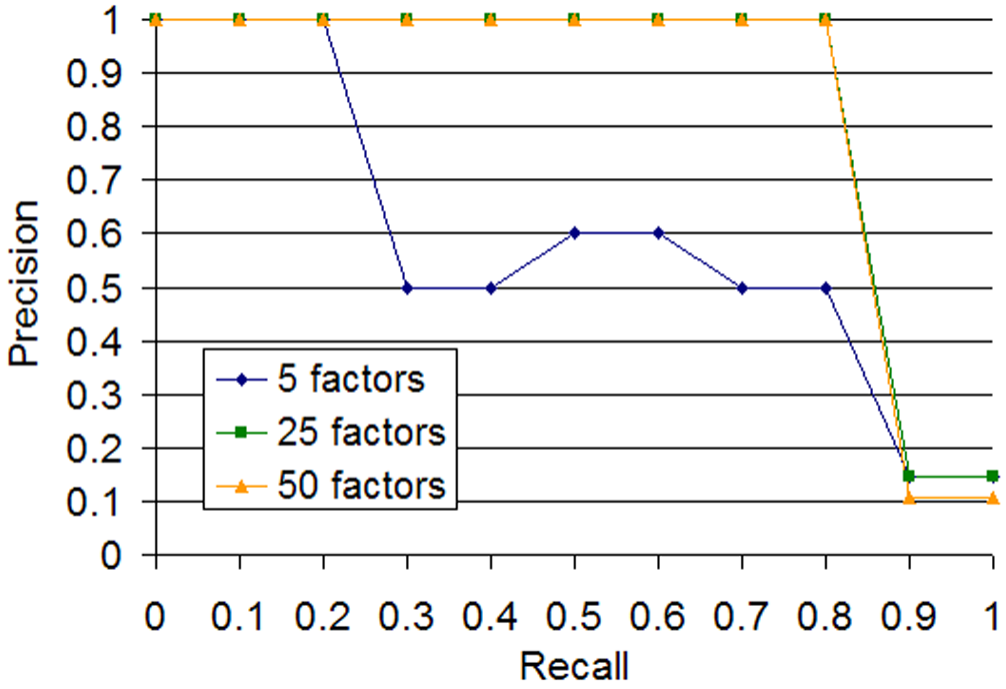

Figure 5.5: Interpolated precision values for identifying primary genes at the decile recall ranges.

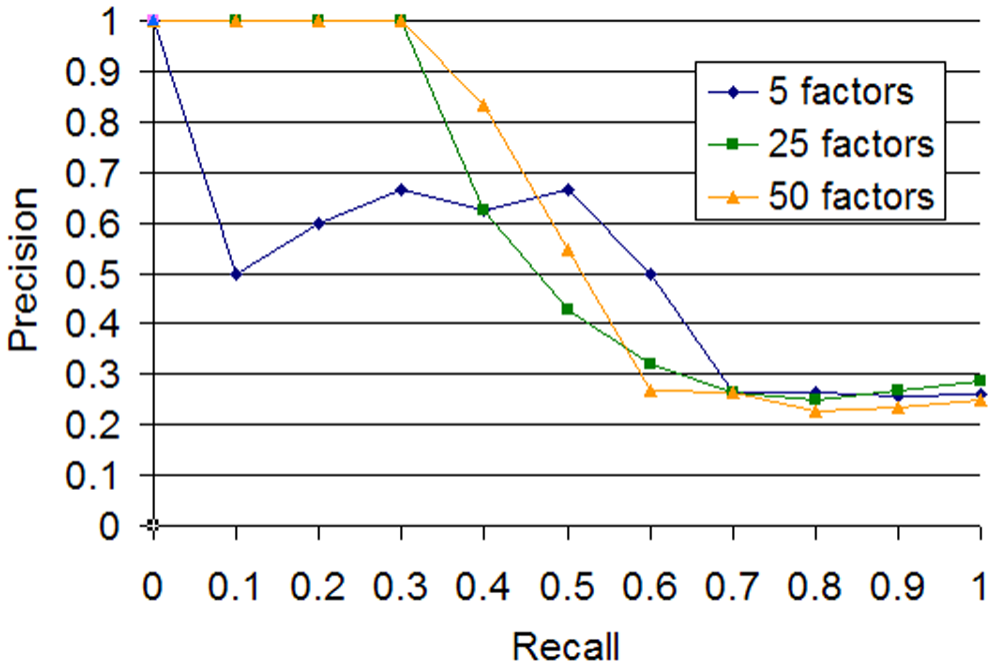

Figure 5.6: Interpolated precision values for identifying primary and secondary genes at the decile recall ranges.

Querying with 25 and 50 factors both produced an average precision of 84% for the five primary genes, while AP dropped to 61% when 5 factors were used. When identifying the twelve secondary genes, however, using 5, 25, and 50 factors produced AP values of 53%, 59%, and 61%, respectively.

The secondary genes, used to simulate latent relationships, demonstrate the power of SGO. SGO was able to correctly identify all twelve genes with acceptable AP. Other intuitive methods, however, were not so successful. For example, examining PubMed co-citations generates results comparable to SGO when identifying the five primary genes. However, only two of the remaining seven secondary genes were identified. Using abstract overlap of LocusLink citations fails to identify any of the indirectly associated genes.

To offset the bias of the 50 gene document collection towards the *Reelin* signaling pathway and to simulate a relatively larger collection with respect to the relevant genes, smaller representations of the five primary genes were constructed. 75%, 50%, 25%, and 5% of the original number of abstracts of the genes involved in *Reelin* signaling were chosen in three random samples. The average of the AP values are depicted in Figure 5.7, with the standard deviation shown at the top of each bar. For a collection of 20,856 Human LocusLink abstracts, SGO was able to identify the five primary genes associated with *Reelin* with an average precision of 47%. Further analysis of that and other large collections is underway.

Unfortunately, precision and recall mean little to biologists. If the hierarchical tree given in Figure 5.3 is cross-referenced with the classifications given in Table A.2, it is evi-

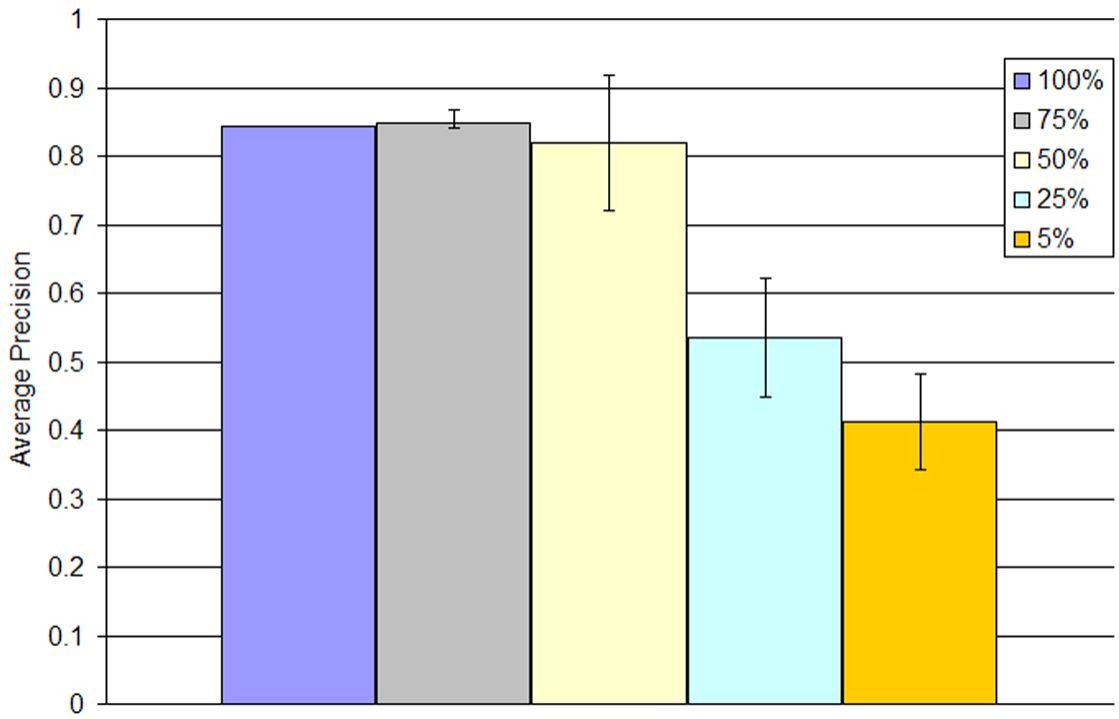

Figure 5.7: Effect on AP of decreasing abstract representation of the five primary *Reelin*-related gene documents.

dent that most functional clusters are preserved. The two most notable exceptions are that of *Fyn* and *Shc1*. *Fyn*, an oncogene, is clustered near the Alzheimer and *Reelin* genes although it did not rank highly with the "Reelin" queries. Similarly, *Shc1*, another oncogene, is clustered near the genes directly associated with *Reelin* signaling. At the time of the construction of the test data set,[5] there was very little evidence to support this association. Recently, however, it has been shown that *Shc1* directly interacts with *App* and plays a significant role with Alzheimer's disease [ZGB+04].

[5]July 7, 2003

Chapter 6

Conclusion

SGO was tested on the 50 gene test data set as a proof-of-principle experiment. With the encouraging results produced from that data set, SGO's data sets must be scaled up to include entire genomes. In fact, SGO's interface allows for other information to parsed and comparisons made between entities other than genes. For example, rather than create gene documents, "patient" documents can be made that describe a patient's symptoms to help physicians quickly identify possible disease threats.

In addition to increasing the size of document collections, the method with which gene documents are created must also be examined. Currently, only titles and abstracts from MEDLINE citations of LocusLink entries are used to form a gene document. LocusLink will soon be replaced by Entrez Gene, so another method must be devised to construct gene documents. Whether through LocusLink, Entrez Gene, or another database, steps must be made to ensure that the abstracts in the gene documents are as noise-free as possible. For

example, sequencing files that contribute little or no meaningful value to a gene document must be removed.

Along the same lines, methods must be devised to help overcome the recall problem. That is, given a gene, find all or a high percentage of the abstracts related to the gene in question rather than a representative few. This approach will help increase the chances of uncovering latent relationships. Another obvious approach that should expose latent relationships is to include entire text documents rather than just titles and abstracts. Of course, including entire documents may introduce more noise than meaningful information.

So far, SGO uses genomic literature to represent genes; however, other information can be used to represent genes. [GDAW03] has been developing a system to deal with the growth of structural protein information. [SB03] applies similar text mining techniques not to genomic literature, but to protein sequence data. In essence, this data can represent genes in a genotypic sense, while using genomic literature to represent genes covers a gene's phenotype. Methods can be derived that combine both genotype (structural and sequence data) and phenotype information about a gene to produce a multi-modal similarity. As hinted in [YM02], literature-based similarity methods often produce results correlated to sequence-based alignment methods. As such, positive results from a multi-modal method would have more inherent validation than one that only considers one aspect of genomic information.

SGO focuses on LSI as its primary retrieval model. It is, however, able to incorporate other models such as the simple vector space model. One such model that looks promis-

ing in the bioinformatics context is the Nonnegative Matrix Factorization (NMF). Rather than ensure orthogonality of factor matrices, NMF guarantees that all factor matrices will remain, as the name implies, nonnegative. As a result, specific features such as gene function can hopefully be identified. This method has been shown to have good clustering and classification results but has yet to be applied to genomic information [Sha04].

As well as increasing the scope of SGO, steps must be taken to ensure that information remains current. Weekly updates of gene document information would be one such step. However, as the amount of information increases, the need for more scalable storage media will become apparent. Using network storage to house larger matrices and other information is one viable option that could help increase the availability of SGO [Mir03].

To make SGO more meaningful for biologists who study large groups of genes, a dynamic tree-building option must be implemented where biologists can submit a list of several hundred genes with which to build a tree. Currently, the Fitch-Margoliash method is used to build trees and was originally chosen for the accurate trees it produced. However, speed may be traded for acceptable losses in accuracy if faster methods such as the neighbor-joining method can produce trees in an interactive manner [KF94].

In the end, SGO must continue to use information to produce useful results in an easily-interpretable format to the user. As with all applications, SGO development will continue to balance speed with solution quality. Ultimately, however, SGO remains and should always remain a tool to help validate current lab research and uncover directions for future exploration.

Bibliography

Bibliography

[BB99] M.W. Berry and M. Browne. *Understanding Search Engines: Mathematical Modeling and Text Retrieval*. SIAM, Philadelphia, PA, 1999.

[BDJ99] M.W. Berry, Z. Drmač, and E.R. Jessup. Matrices, vector spaces, and information retrieval. *SIAM Review*, 41(2):335–362, 1999.

[BDO95] M.W. Berry, S.T. Dumais, and G.W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):573–595, 1995.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, New York, 1999.

[Cho99] G.G. Chowdhury. *Introduction to Modern Information Retrieval*. Library Association, London, 1999.

[DDF+90] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

[DT01] R. Dhavan and L.H. Tsai. A decade of cdk5. *Molecular Cell Biology*, 2:749–759, 2001.

[FM67] W.M. Fitch and E. Margoliash. Construction of phylogenetic trees. *Science*, 155(3760):279–284, 1967.

[GDAW03] R. Gaizauskas, G. Demetriou, P.J. Artymiuk, and P. Willett. Protein structure and information extraction from biological texts: The PASTA system. *Bioinformatics*, 19(1):135–143, 2003.

[GL96] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 3rd edition, 1996.

[GWB03] J.T. Giles, L. Wo, and M.W. Berry. GTP (General Text Parser) software for text mining. In H. Bozdogan, editor, *Statistical Data Mining and Knowledge Discovery*, pages 455–471, Boca Raton, FL, 2003. CRC Press.

[HBDV03] K. Heinrich, M.W. Berry, J.J. Dongarra, and S. Vadhiyar. The semantic conference organizer. In H. Bozdogan, editor, *Statistical Data Mining and Knowledge Discovery*, pages 571–581, Boca Raton, FL, 2003. CRC Press.

[HHW+03] Y. Hu, L.M. Hines, H. Weng, D. Zuo, M. Rivera, A. Richardson, and J. LaBaer. Analysis of genomic and proteomic data using advanced literature mining. *Journal of Proteome Research*, 2(4):405–412, 2003.

[HHWB04] R. Homayouni, K. Heinrich, L. Wei, and M.W. Berry. Gene clustering by latent semantic indexing of MEDLINE abstracts. *Bioinformatics*, 2004. In Review.

[HRSC99] R. Homayouni, D.S. Rice, M. Sheldon, and T. Curran. Disabled-1 binds to the cytoplasmic domain of amyloid precursor-like protein 1. *Journal of Neuroscience*, 19:7507–7515, 1999.

[Jia97] Jingqian Jiang. Using latent semantic indexing for data mining. Master's Thesis, Department of Computer Science, University of Tennessee, 1997.

[JLKH01] T.K. Jenssen, A. Lægreid, J. Komorowski, and E. Hovig. A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics*, 28:21–28, 2001.

[KF94] M.K. Kuhner and J. Felsenstein. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology Evolution*, 11(3):459–468, 1994.

[KT98] Y.T. Kwon and L.H. Tsai. A novel disruption of cortical development in p35 (-/-) mice distinct from reeler. *Journal of Computational Neurology*, 155:510–522, 1998.

[LB97] T.A. Letsche and M.W. Berry. Large-scale information retrieval with latent semantic indexing. *Information Sciences*, 100:105–137, 1997.

[Let96] T.A. Letsche. Toward large-scale information retrieval using latent semantic indexing. Master's Thesis, Department of Computer Science, University of Tennessee, 1996.

[LLD04] T.K. Landauer, D. Laham, and M. Derr. From paragraph to graph: Latent semantic analysis for information visualization. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl. 1):5214–5219, 2004.

[Mir03] S. Mironova. Integrating network storage into information retrieval applications. Master's Thesis, Department of Computer Science, University of Tennessee, 2003.

[MSY03] W.H. Majoros, G.M. Subramanian, and M.D. Yandell. Identification of key concepts in biomedical literature using a modified markov heuristic. *Bioinformatics*, 19(3):402–407, 2003.

[NLM] NLM fact sheets. www.nlm.nih.gov/pubs/factsheets/factsheets.html, accessed July, 2004.

[ORRW81] R.N. Oddy, S.E. Robertson, C.J. van Rijsbergen, and P.W. Williams, editors. *Information Retrieval Research*. Butterworth, London, 1981.

[PKSM00] K.D. Pruitt, K.S. Katz, H. Sicotte, and D.R. Maglott. Introducing RefSeq and LocusLink: curated human genome resources at the NCBI. *Trends in Genetics*, 16(1):44–47, 2000.

[SB03] G.W. Stuart and M.W. Berry. Comprehensive whole genome bacterial phylogeny using correlated peptide motifs defined in a high dimensional vector space. *Journal of Bioinformatics and Computational Biology*, 1(3):475–493, 2003.

[Sha04] F. Shahnaz. A clustering method based on nonnegative matrix factorization for text mining. Master's Thesis, Department of Computer Science, University of Tennessee, 2004.

[SM83] G. Salton and M.J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.

[WH04] D.M. Wilkinson and B.A. Huberman. A method for finding communities of related genes. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl. 1):5241–5248, 2004.

[YM02] M.D. Yandell and W.H. Majoros. Genomics and natural language processing. *Nature Reviews Genetics*, 3:601–610, 2002.

[ZGB+04] N. Zambrano, D. Gianni, P. Bruni, F. Passaro, F. Telese, and T. Russo. Fe65 is not involved in the platelet-derived growth factor-induced processing of Alzheimer's amyloid precursor protein, which activates its caspase-directed cleavage. *Journal of Biological Chemistry*, 279:16161–16169, 2004.

Appendix

Appendix A

Genes Used in Test Data Set

The 50 gene test data set and all associated information is presented in this appendix [HHWB04].

1. Table A.1 gives a list of the genes along with their official gene names.

2. Table A.2 lists the genes along with their assumed primary and secondary classifications.

3. Table A.3 shows the number of Human (H), Rat (R), and Mouse (M) LocusLink citations for each gene, along with other identifying information such as GenBank Accession Number, Unigene ID, and LocusLink IDs.

Table A.1: Genes in the test data set.

Gene Symbol	Official Gene Name
A2M	alpha-2-macroglobulin
ABL1	v-abl Abelson murine leukemia oncogene 1
APBA1	amyloid beta (A4) precursor protein-binding, family A, member 1
APBB1	amyloid beta (A4) precursor protein-binding, family B, member 1
APLP1	amyloid beta (A4) precursor-like protein 1
APLP2	amyloid beta (A4) precursor-like protein 2
APOE	apolipoprotein E
APP	amyloid beta (A4) precursor protein
ATOH1	atonal homolog 1 (Drosophila)
BRCA1	breast cancer 1
BRCA2	breast cancer 2
CDK5	cyclin-dependent kinase 5
CDK5R	cyclin-dependent kinase 5, regulatory subunit (p35)
CDK5R2	cyclin-dependent kinase 5, regulatory subunit 2 (p39)
DAB1	disabled homolog 1 (Drosophila)
DLL1	delta-like 1 (Drosophila)
DNMT1	DNA methyltransferase (cytosine-5) 1
EGFR	epidermal growth factor receptor
ERBB2	v-erb-b2 erythroblastic leukemia viral oncogene homolog 2
ETS1	E26 avian leukemia oncogene 1, 5′ domain
FOS	FBJ osteosarcoma oncogene
FYN	Fyn proto-oncogene
GLI	GLI-Kruppel family member GLI
GLI2	GLI-Kruppel family member GLI2
GLI3	GLI-Kruppel family member GLI3
JAG1	jagged 1
KIT	kit oncogene
LRP1	low density lipoprotein receptor-related protein 1
LRP8	low density lipoprotein receptor-related protein 8, apolipoprotein
MAPT	microtubule-associated protein tau
MYC	myelocytomatosis oncogene
NOTCH1	Notch gene homolog 1 (Drosophila)
NRAS	neuroblastoma ras oncogene
PAX2	paired box gene 2
PAX3	paired box gene 3
PSEN1	presenilin 1
PSEN2	presenilin 2
PTCH	patched homolog
RELN	reelin
ROBO1	roundabout homolog 1 (Drosophila)
SHC1	src homology 2 domain-containing transforming protein C1
SHH	sonic hedgehog
SMO	smoothened homolog (Drosophila)
SRC	Rous sarcoma oncogene
TGFB1	transforming growth factor, beta 1
TRP53	transformation related protein 53
VLDLR	very low density lipoprotein receptor
WNT1	wingless-related MMTV integration site 1
WNT2	wingless-related MMTV integration site 2
WNT3	wingless-related MMTV integration site 3

Table A.2: Classifications associated with each gene.

Gene Symbol	Classification	
	Primary	Secondary
A2M	Alzheimer	
APBA1	Alzheimer	
APBB1	Alzheimer	
APLP1	Alzheimer	
APLP2	Alzheimer	
APOE	Alzheimer	
APP	Alzheimer	
LRP1	Alzheimer	
MAPT	Alzheimer	
PSEN1	Alzheimer	
PSEN2	Alzheimer	
ABL1	Cancer	
BRCA1	Cancer	
BRCA2	Cancer	
DNMT1	Cancer	
EGFR	Cancer	
ERBB2	Cancer	
ETS1	Cancer	
FOS	Cancer	
KIT	Cancer	
MYC	Cancer	
NRAS	Cancer	
TRP53	Cancer	
SHC1	Cancer	
SRC	Cancer	
FYN	Cancer	Reelin
ATOH1	Development	
CDK5	Development	Alzheimer
CDK5R	Development	Alzheimer
CDK5R2	Development	Alzheimer
DLL1	Development	Cancer
GLI	Development	Cancer
GLI2	Development	Cancer
GLI3	Development	Cancer
JAG1	Development	Cancer
NOTCH1	Development	Cancer
PAX2	Development	Cancer
PAX3	Development	Cancer
PTCH	Development	Cancer
ROBO1	Development	Cancer
SHH	Development	Cancer
SMO	Development	Cancer
TGFB1	Development	Cancer
WNT1	Development	Cancer
WNT2	Development	Cancer
WNT3	Development	Cancer
DAB1	Development	Reelin
LRP8	Development	Reelin
RELN	Development	Reelin
VLDLR	Development	Reelin

Table A.3: Number of citations associated with LocusLink entries for each gene sorted by total number of citations.

Gene Symbol	Accession Number	Unigene ID	LocusLink ID			Number of Citations			
			H	R	M	H	R	M	Total
APLP1	L04538	Mm.2381	333	11803	29572	4	3	1	8
CDK5R2	U90267	Mm.288703	8941	12570		3	5	0	8
WNT3	M32502	Mm.5188	7473	22415	24882	5	6	0	11
ROBO1	Y17793	Mm.310772	6091	19876	58946	3	8	1	12
DAB1	Y08380	Mm.289682	1600	13131	266729	3	11	0	14
LRP8	AJ312058	Mm.276656	7804	16975		6	9	0	15
WNT2	AI507247	Mm.33653	7472	22413	114487	5	8	2	15
ATOH1	D43693	Mm.57229	474	11921		3	13	0	16
DLL1	AV007019	Mm.4875	28514	13388	84010	9	7	0	16
APBA1	AF029106	Mm.22879	320	108119	83589	14	2	1	17
GLI	AB025922	Mm.336839	2735	14632	140589	11	6	0	17
SMO	AF089721	Mm.29279	6608	20596		9	9	0	18
VLDLR	L33417	Mm.4141	7436	22359	25696	6	12	0	18
GLI2	X99104	Mm.273292	2736	14633		8	11	0	19
SHC1	AI050321	Mm.86595	6464	20416	85385	6	13	1	20
A2M	AY185125	Mm.30151	2	232345	24153	14	2	5	21
CDK5R	U89527	Mm.142275	8851	12569	116671	8	10	3	21
PAX2	X55781		5076	18504		16	7	0	23
APLP2	AV313336	Mm.19133	334	11804	25382	6	14	4	24
WNT1	M11943	Mm.1123	7471	22408	24881	11	14	0	25
APBB1	AI839886	Mm.38469	322	11785	29722	15	8	3	26
GLI3	X95255	Mm.5098	2737	14634	140588	10	16	0	26
LRP1	X67469	Mm.271854	4035	16971		15	11	0	26
JAG1	AF171092	Mm.22398	182	16449	29146	19	7	1	27
PTCH	U46155	Mm.3057	5727	19206	89830	18	11	1	30
NRAS	X13664	Mm.256975	4893	18176	24605	18	12	1	31
PAX3	X59358	Mm.1371	5077	18505	114502	17	14	0	31
ETS1	AA929300	Rn.88756	2113	23871	24356	19	14	1	34
CDK5	D29678	Mm.298798	1020	12568	140908	10	17	8	35
DNMT1	AF036008	Mm.128580	1786	13433	84350	14	20	2	36
FYN	M27266	Mm.4848	2534	14360	25150	17	18	4	39
PSEN2	U57325	Mm.330850	5664	19165	81751	27	11	3	41
RELN	AV263736	Mm.3057	5649	19699	24718	17	22	4	43
SRC	M17031	Mm.22845	6714	20779	83805	27	14	5	46
ABL1	J02995	Mm.1318	25	11350	311860	44	12	0	56
BRCA2	U89652	Mm.236256	675	12190	25082	48	10	2	60
MAPT	M18775	Mm.1287	4137	17762	29477	43	14	4	61
FOS	AV252296	Mm.246513	2353	14281	24371	23	19	21	63
KIT	Y00864	Mm.247073	3815	16590	64030	32	30	1	63
NOTCH1	AV374287	Mm.290610	4851	18128	25494	20	44	5	69
SHH	X76290	Mm.57202	6469	20423	29499	18	46	7	71
PSEN1	L42177	Mm.998	5663	19164	29192	53	26	6	85
MYC	L00039	Mm.2444	4609	17869	24577	59	34	9	102
ERBB2	AW213701	Mm.290822	2064	13866	24337	95	23	9	127
APOE	AV092985	Mm.305152	348	11816	25728	93	31	4	128
APP	U82624	Mm.277585	351	11820	54226	85	34	12	131
BRCA1	U32446	Mm.244975	672	12189	24227	114	20	3	137
EGFR	L06864	Mm.8534	1956	13649	24329	89	40	11	140
TGFB1	AJ009862	Mm.248380	7040	21803	59086	111	49	22	182
TRP53	AB021961	Mm.222	7157	22059	24842	222	122	17	361

Vita

Kevin Erich Heinrich was born in Würzburg, Germany on October 24, 1979. He graduated Valedictorian from Maryville High School in Maryville, Tennessee in 1997. He received Top Graduate Honors and a Bachelor of Science degree in Computer Science and Honors Mathematics with a minor in Economics from the University of Tennessee in May 2001. Interested in his current projects, he remained at the University at Tennessee to receive a Master of Science degree in Computer Science in August 2004. He plans to continue his research en route to a Doctorate in Computer Science.