Genes, Themes and Microarrays

Using Information Retrieval for Large-Scale Gene Analysis

Hagit Shatkay **Stephen Edwards** **W. John Wilbur** **Mark Boguski**

National Center for Biotechnology Information
NLM, NIH
Bethesda, Maryland 20984
{shatkay,edwards}@ncbi.nlm.nih.gov

Abstract

The immense volume of data resulting from DNA microarray experiments, accompanied by an increase in the number of publications discussing gene-related discoveries, presents a major data analysis challenge. Current methods for genome-wide analysis of expression data typically rely on cluster analysis of gene expression patterns. Clustering indeed reveals potentially meaningful relationships among genes, but can not explain the underlying biological mechanisms. In an attempt to address this problem, we have developed a new approach for utilizing the literature in order to establish functional relationships among genes on a *genome-wide scale.* Our method is based on revealing coherent themes within the literature, using a similarity-based search in document space. Content-based relationships among abstracts are then translated into functional connections among genes. We describe preliminary experiments applying our algorithm to a database of documents discussing yeast genes. A comparison of the produced results with well-established yeast gene functions demonstrates the effectiveness of our approach.

Keywords: *genomics, microarray, machine learning, information retrieval, document databases*

Introduction

The development of DNA microarrays during the last few years (Schena *et al.* 1995; DeRisi, Iyer, & Brown 1997), allows researchers to simultaneously measure the expression levels of thousands of different genes. Experiments involving such arrays produce overwhelming amounts of data. In response, much recent work has been concerned with automating the analysis of microarray data. Currently pursued techniques (e.g. Eisen *et. al.* (1998), Tamayo *et. al.* (1999), Ben-Dor *et. al.* (1999)) concentrate mostly on applying clustering methods directly to the expression data, in order to find clusters of genes demonstrating similar expression patterns. The assumption motivating such search for co-expressed genes is that simultaneously expressed genes often share a common function. However, there are several reasons that cluster analysis alone cannot fully address this core issue:

1. Genes that are functionally related may demonstrate strong anti-correlation in their expression levels, (a gene may be strongly suppressed to allow another to be expressed), thus clustered into separate groups, blurring the relationship between them.
2. As shown later, simultaneously expressed genes do not always share a function. Moreover, genes that are expressed at different times may serve complementing roles of one unifying function.
3. Even when similar expression levels correspond to similar functions, the function and the relationships between genes in the same cluster can not be determined from the cluster data alone. Testing, justifying, and explaining the formed clusters requires a lot of additional research effort.
4. Due to the interrelated nature of biological processes, genes may have more than a single function. The strict assignment of genes to clusters, resulting from most clustering methods currently used, may prove overly stringent, potentially preventing the exposure of complex interrelationships between genes.

The work described in this paper aims to complement the existing methods by providing a much-needed biological context, based on a survey of the existing literature. The assumption underlying our approach is that the function of many genes is described in the literature, and by relating documents talking about well understood genes to documents discussing other genes, we can predict, detect and explain the functional relationships between the many genes that are involved in large-scale experiments. We do not attempt here to draw any functional or relational information from the expression array itself. Instead, we use a large database of documents as our information search space. Each gene is represented by a document, roughly discussing the gene's biological function. The literature database is then searched for documents similar to the gene's document. Thus, for each gene we produce a set of documents that are related to its functional role. We then look for similarities between the resulting sets of documents. Since each set corresponds to a gene, we can map the similar document sets back to their corresponding genes, and establish functional relationships among these genes.

To accomplish this goal, we use a new statistical information-retrieval method (Shatkay, Wilbur 2000) to conduct the similarity search based on the gene's document. As an integral part of our algorithm, we produce an "executive summary", consisting of a few characteristic content bearing terms in the set of documents assigned to each gene. Thus we simultaneously achieve three goals:

- Finding functional relationships between genes.
- Obtaining the literature specifically relevant to the function of these genes.
- Producing a short summary justifying why the genes were considered relevant to each other, and what their function is.

This functional information can then be correlated with the expression array cluster analysis to refine the resulting hypotheses and, by extension, future experiments.

The rest of this paper is organized as follows: The next section surveys related work on gene analysis, both based directly on expression array data and on literature mining. We then describe our approach of using the literature to find function and relationships between genes. Next we discuss our preliminary experiments and results over the set of well-studied yeast genes discussed by Spellman *et. al.* (1998). Our results demonstrate that the automated usage of literature is an extremely powerful tool for determining relationships between genes, for explaining expression-based clusters obtained from array-based experiments, and for assisting in the design of further experiments.

Related Work

The first part of this section provides further background on the analysis of data obtained from gene expression arrays and the challenges it poses; the second part discusses current methods for using the literature for gene analysis.

Analyzing Gene Expression Arrays

DNA microarrays represent the latest in a series of powerful tools based on hybridizing a soluble DNA/RNA molecule to its complementary strand immobilized on a solid support (Southern 1975; Wahl, Meinkoth, & Kimmel 1987; Schena *et al.* 1995). With DNA microarrays, cDNA corresponding to known genes is spotted onto the solid support (usually a glass slide). The mRNA from cells or tissues is then converted into fluorescently labeled cDNA and applied to the unlabeled cDNA matrix (Schena 1999). Since each spot on the matrix corresponds to a known gene or EST, the expression level of thousands of genes can be measured in a single experiment. DNA microarrays consisting of the entire known genome from *Escherichia coli, Mycobacterium tuberculosis*, and *Saccharomyces cerevisiae* already exist (Brown & Botstein 1999), and those representing *Caenorhabditus elegans* and *Drosophilia melanogaster* genome sequences should be available soon. In addition, commercially available DNA microarrays and oligonucleotide arrays exist for most of the human genes characterized to date and can be expected for the whole human genome once it is completely sequenced and annotated within the next three years.

This new technology allows gene expression experiments to be performed on a genome-wide scale. Experiments with *S. cerevisiae* have studied changes in gene expression patterns for over 95% of the protein coding genes simultaneously under a variety of conditions (Cho *et al.* 1998; Spellman *et al.* 1998; DeRisi, Iyer, & Brown 1997; Chu *et al.* 1998). This increase in percentage of genome measured, has an immediate impact on the number of genes awaiting analysis. For example, the number of genes collectively identified as being induced during sporulation dramatically increased from a total of 50 to approximately 500 from a single set of genome wide microarray experiments (Chu *et al.* 1998).

With this increased volume of data manual gene analysis becomes impractical, and there is an immediate need for more powerful methods of data analysis (Ermolaeva *et al.* 1998; Bassett, Eisen, & Boguski 1999). Most efforts to date have involved clustering genes based on their expression patterns and using these clusters to infer functional correlation. Methods involving hierarchical clustering, commonly applied in sequence and phylogenetic analysis, have been used with the yeast data sets described previously (Eisen *et al.* 1998). As expected, in many cases this clustering revealed that genes with a common function were indeed coexpressed (Spellman *et al.* 1998; Eisen *et al.* 1998). Self- organizing maps (Tamayo *et al.* 1999) and other clustering methods (Wen *et al.* 1998; Ben-Dor & Yakhini 1999) have also been shown to effectively group genes by the observed expression patterns.

While clusters of simultaneously expressed genes can correlate with shared function, this is not always the case. The complex and parallel nature of the system causes some genes to share similar expression profiles despite the distinct biological processes in which they are involved. In fact, careful analysis of the CLB2 cluster described by Spellman *et. al.* (1998) reveals genes involved in several different cellular functions. For example, CHS2, BUD8, and IQG1 are all involved in maintenance of the cell wall while ACE2, ALK1, and HST3 are involved in nuclear events. This example demonstrates the wealth of biological information that is not represented by temporal gene clusters.

In addition, some members of a common signaling pathway may play antagonistic roles and actually show an anti-correlation with regards to gene expression. As a result, the clusters obtained from shared gene expression profiles must still be analyzed with respect to known biological roles, before reliable conclusions about their biological functions can be drawn from the data.

A more recent approach to array analysis uses Bayesian networks to describe relationships between genes (Friedman *et al.* 2000). Rather than simply group genes according to their related expression patterns, this approach allows the identification of *causal relationships* among genes. Indeed, based on the analysis of 800 genes shown to have regulated gene expression during the yeast cell cycle (Spellman *et al.* 1998), only a few of these genes appeared to dominate the order of expression (Friedman *et al.* 2000), and the results could highlight the critical genes for establishing the yeast cell cycle. While this analysis can suggest causal relationships between genes, it does not provide the biological explanation for these relations. In some cases, only further experimentation can determine the involved mechanism. However, it is highly likely that in many of these cases, this information currently exists in the published literature.

The current method for explaining the discovered clusters and relationships, has been for individuals to search through the literature, gene by gene, or rely on their own knowledge of the biological processes involved. While such a method can be effective on a small scale, it produces a major bottleneck when performing experiments on a genome-wide scale.

It is for this reason that we propose the development of an automated method for relating genes according to their biological function based on the current literature. Our method complements the approaches describe above, by providing literature-based explanations to the clusters and the relationships that are discovered through the expression arrays. The next section surveys current research aimed at automating literature mining in the area of gene analysis.

Text Usage in Biological Analysis

With the advancement of genome sequencing techniques comes an overwhelming increase in the amount of literature discussing the discovered genes. As an illustrative example, the number of *PubMed* documents containing the word *gene* published between the years $1970 - 1980$ is a little over $35,000$, while the number of such documents published between the years $1990 - 2000$ is $402,700$ – over a ten fold increase. Thus, surveying the literature for information about genes requires a great deal of time and effort. It can not be effectively and efficiently done using the currently available search techniques, given the large number of genes involved in current expression array experiments. The problem is further aggravated by the non-uniform nomenclature used in the literature as illustrated below.

The most widely used on-line source for gene-related abstracts is the *PubMed* database. An initial step in the search for relevant literature in *PubMed* is the specification of a *boolean* query. The user provides either a single term (e.g. OLE1), or a boolean combination of terms (e.g. OLE1 *AND* sterol). The result is the set of *all* documents found in the database which satisfy the constraints specified in the query. This form of query suffers from several well-known deficiencies:

- A *prohibitively large* number of documents are typically retrieved.
- A substantial part of the retrieved documents are *irrelevant* to the user's information needs.
- Many relevant documents *may not be retrieved*, despite their relevance. For instance, documents that talk about OLE1 using one of its aliases such as *DNA repair protein fatty-acid desaturase 1* or *ACYL-COA desaturase 1* will not be retrieved.

A lot of recent work on mining the literature for genes and proteins aims at supporting the boolean paradigm, improving it to produce more accurate results (thus mostly addressing the first two problems). Such work concentrates on automated natural language processing for finding relevant phrases and useful facts in text. It is intended to assist in finding documents about a given gene, or about the relationships between specific genes. Leek (1997) suggests a way of using hidden Markov models (HMM)s for extracting sentences discussing gene positions on chromosomes from text. Craven and Kumlien (1999) introduce a method for transforming flat text documents into databases of facts about relationships between genes/proteins, performing a task similar to the one Leek addresses, without the need to obtain an HMM for discovering these relationships. Rindflesch *et. al.* (2000) present a method based on parsing and using thesauri to automatically extract facts about genes and proteins from documents. Blaschke *et. al.* (1999) also use a similar method for extracting information about protein interaction from scientific text. Most of the above methods have only been applied to small and limited sample sets of documents/terms. They all stem from the boolean query paradigm, and require the user to specify a very accurate query in order to provide high-quality results.

Another recent system aiming at improving the quality of the results returned from boolean search over genes is *MedMiner* by Tanabe *et. al.* (1999). It provides a good interface to two databases, *Genecards* and *PubMed*. In order to retrieve documents that are likely to be of interest to the user, it relies on a human-generated list of keywords, whose presence in a document discussing genes typically indicates that the document is of high quality and relevance. Still, *MedMiner* provides abundant information about a single gene or about the relationship between two specified genes. Such quantities of information generated per gene when hundreds of genes are involved can not be effectively handled by a user.

The above methods all rely on strong assumptions regarding the use of natural language, such as the terms typically used to indicate relationships and the way sentences are structured. With the shift towards the analysis of mammalian systems the problem of non-uniform nomenclature and language usage is likely to worsen. Gene symbols are rarely used in the mammalian system literature. Instead, the discussion involves a large variety of terms describing the genes. This additional complication will make it difficult for the user to form accurate boolean queries. It is also likely to reduce the effectiveness of literature mining strategies that are based on gene symbol identifiers (such as the one suggested by Leek) and on strong assumptions about the way genes names are used in sentences. Moreover, these systems can indeed be helpful when searching for information about a few genes at a time, but do not address the need for finding links and functional relationships among thousands of genes.

An alternative to the boolean query paradigm is the use of *similarity queries*; the user provides a sample document that is relevant to the subject of interest, and gets back other documents discussing the same subject matter. Such a query mechanism does not depend on the user choice of query terms, but rather on the contents and quality of the example document. The ability to retrieve quality documents that are indeed similar in contents to the example document strongly depends on defining a similarity measure and a search procedure that ranks the relevant documents high and the irrelevant ones low. We have recently developed a probabilistic algorithm that, given an example document, finds a set of documents that are most relevant to it (*a theme*) and provides a set of terms summarizing the contents of this set of documents (Shatkay, Wilbur 2000). The use of similarity queries in general and this algorithm in particular, forms the basis to our approach as described in the next section.

The ultimate challenge in the use of literature for analyzing expression arrays is the ability to obtain an overview of the whole landscape of genes and their related literature. A good literature analysis tool should provide information such as which genes are functionally related to each other, what their shared functionality is and which documents discuss this functionality. It should also provide summaries that allow easy and quick browsing through the literature, and an easy access to the most relevant documents. The next section describes the new approach we have developed in order to meet such challenges.

Discovering Gene Functions and Relations through the Literature

The hypothesis underlying our approach is that the function of many individual genes is discussed in the literature and that a good analysis of the literature is a primary step both for experimental design and for result analysis following such experiments.

Acting under this hypothesis, we shift our attention from the gene-expression space to document space. Thus we start with a large database of documents containing all the relevant literature discussing the domain of interest (for instance – all the documents in *PubMed* that discuss yeast genes). Each gene is mapped to a single document discussing it; each such document is treated as a representative of the gene. We call each document thus associated with a gene the *kernel document* for that gene.

Using our algorithm for finding similar documents, we obtain for each gene a body of related literature (20-50 documents sharing a common *theme*) based on the document representing the gene, along with an "executive summary" containing the terms that characterize the relevant literature. It is important to note that the abstracts retrieved by our algorithm are considered relevant *not* because they contain the *same gene name* as the one associated with the kernel abstract, but rather because they discuss the *same issues* (which typically corresponds to functionality) as those discussed in the kernel document.

There are several ways to use the set of documents retrieved for each gene in order to derive relationships among genes:

- One can simply mine this set for the names of other genes as done by any of the algorithms described in the previous section. The main limitation of doing so is the dependency on explicit rules for detecting gene names, with the risk of overlooking important information while detecting unimportant relationships.
- A more effective way is to *automatically* compare the *sets of documents* retrieved for each gene, and determine that genes share similar functionality if the literature associated with each of them is similar.
- A third possible way is to use the *terms* characterizing the retrieved literature, as they occur in the summary, and consider genes as related if their summaries consist of the same (or almost the same) set of terms.

We currently use the second of these methods to determine relationships among genes, as described later in this section.

The first step in our approach requires mapping the set of genes $\langle G_1, \ldots, G_N \rangle$ to a set of kernel documents $\langle K_1, \ldots, K_N \rangle$ (see top of Figure 2). Kernel documents are currently obtained from the available curated literature about yeast genes (as explained in the experiments part of this paper). Our method strongly depends on the quality of the kernel documents. Abstracts discussing experimental methods rather than gene function tend to draw other documents describing the same experimental methods. The result is a document set not representative of the gene's function. On the other

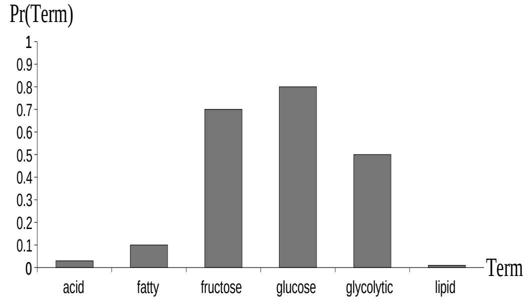

Figure 1: Typical term distribution for the *Nutrition* theme.

hand, kernels discussing gene biology typically lead to high quality information about the functionality of related genes. We are currently considering ways to automate the kernel selection process, so that each kernel faithfully represents the biology of its associated gene.

The rest of this section provides the details of our approach. We first outline the similarity query algorithm used for finding related abstracts starting from a kernel document. (A complete discussion of the models and the algorithms can be found in (Shatkay, Wilbur 2000)). We then describe how similarities between the obtained document collections are detected.

Similarity Queries over Documents

Our algorithm is based on the idea that documents which share a common theme can be modeled as though they were generated through sampling from a common set of independent *Bernoulli* distributions representing the theme. For example, a set of documents discussing genes responsible for *nutrition* during the cell-cycle, are likely to contain terms such as *fructose* or *glucose* and quite unlikely to contain the term *lipid*, as illustrated in Figure 1.

Each document in our document database, *DB*, is modeled as an M-dimensional binary vector, where M is the number of distinct terms[1] $\{t_1, \ldots, t_M\}$ in the database. Formally, a document d is a vector $\langle d_1, d_2, \ldots, d_M \rangle$, where:

$$d_i = \delta_{di} \overset{\text{def}}{=} \begin{cases} 1 & \text{if } t_i \in d \ , \\ 0 & \text{otherwise} \ . \end{cases} \quad (1)$$

Given a theme T, we view the presence/absence of terms in document d in the database DB, as a result of M independent Bernoulli events, each of which stems from one of three families of Bernoulli distributions:

- p_i^T — the probability that the term t_i occurs in a document d, given that d is a *theme* document: $p_i^T \overset{\text{def}}{=} \Pr(t_i \in d | d \in T)$.
- q_i^T — the probability that the term t_i occurs in a document d, given that d is an *off-theme* document: $q_i^T \overset{\text{def}}{=} \Pr(t_i \in d | d \notin T)$.
- DB_i — the probability that the term t_i occurs in a document d, given that d is a document in the database, regardless of its being an on-theme or an off-theme document: $DB_i \overset{\text{def}}{=} \Pr(t_i \in d | d \in DB)$.

The distribution DB_i models the possible arbitrary usage of terms in the language, without being strongly indicative of the main topic discussed. (e.g. the sentence *"He entered the building"* is not particularly relevant to the topic *construction*, despite the occurrence of the term *building* in it).

The a priori probability of any document $d \in DB$, regardless of its contents, to be a theme document is denoted as P_d: $P_d \overset{\text{def}}{=} Pr(d \in T)$.
Throughout this paper, we assume this parameter to be known and fixed for all documents, and we do not attempt to estimate it here. (In the experiments described later, $P_d = 0.01$ for all $d \in DB$.)

The last component of our model is the Bernoulli event representing the choice made for each term t_i, in each document d, whether it is to be generated according to the database probability, DB_i or according to the specific on/off-theme distribution. We denote this probability, for each term t_i, as λ_i.

The process by which each document $d \in DB$ is generated, given a specific theme, T, can be modeled as follows: First it is decided if the document d is inside the theme T or not. The probability for $d \in T$ is P_d. Then *for each term*, t_i, it is decided if t_i is generated according to the general database distribution, DB_i, or according to its specific theme/off-theme distribution. The probability of a term t_i to be generated according to the general database distribution DB_i is λ_i.
Finally, the decision whether to include the term in the document d is based on one of three possibilities:

- If t_i is to be generated according to the general DB distribution, it is included in d with probability DB_i. *Otherwise:*
- If d is a theme document, t_i is included in d with probability p_i^T.
- If d is an off-theme document, t_i is included in d with probability q_i^T.

Note that for each document $d \in DB$, we *know* the terms it contains. The *missing information* is which documents are *theme* documents and which terms are generated from the general distribution, DB_i, as opposed to the theme-specific ones, p_i^T and q_i^T.

Given a single document representing the gene, our task is to find the characteristic set of Bernoulli distributions, $(p^T, q^T \text{ and } \lambda)^2$, for all terms i, and use it to

[1] Terms consist of one or two words, excluding stop words. They are extracted from the raw text in a standard preprocessing stage.

[2] Note that estimating DB_i is straightforward since all

find the documents that are highly likely to have been generated by sampling from these distributions. The latter documents are the ones focused on the theme represented by these distributions. In addition, we produce a set of terms characterizing this theme. These are the terms that have a high probability to occur in theme documents (high p_i^T) *and* a much lower probability to occur in documents outside the theme (high ratio p_i^T/q_i^T).

To estimate the Bernoulli parameters under missing information as described above, we use an Expectation Maximization algorithm(EM) (Dempster, Laird, & Rubin 1977); it aims to maximize the likelihood of the database partition into theme/off-theme documents, given the Bernoulli parameters, based on the kernel document. The complete algorithm is described elsewhere (Shatkay, Wilbur 2000), and we provide only its outline here. An EM algorithm starts by initializing the model parameters, (p^T, q^T, λ^T), based on some prior knowledge; in our case the initial assignment is a rough approximation of the Bernoulli parameters based on the kernel document and its comparison to the rest of the database. It then alternates between:

- the *E-step* of computing the *expected values*, for the likelihood of the documents to be in the same theme as the kernel document, under the current parameter estimates, and
- the *M-step* of finding new model parameters that maximize the likelihood of the database partition into theme/off-theme documents given the parameters.

This iterative process is guaranteed, under mild conditions, to provide monotonically increasing convergence of the likelihood function, and we have proven that our algorithm indeed converges to such a local maximum.

We execute this algorithm for each of the kernel documents, $\langle K_1, \ldots, K_N \rangle$, representing each of the genes, $\langle G_1, \ldots, G_N \rangle$, as illustrated in the top part of Figure 2. The result from the run for each gene consists of:

- a list of the top 50 documents discussing the same theme as the kernel document, ordered by their degree of relevance to the theme, and
- a list of terms (keywords) characterizing the theme, ordered by their degree of relevance to the theme.

Note that the keywords provided in the list are not merely the terms most probable to occur in the set of documents discussing the theme, but rather those that are much more probable to occur in this set than in the rest of the database (p_i^T/q_i^T is high). Simply using the most frequent terms, (as done, for example, by Tanabe *et. al.* (1999)), typically results in terms that are common throughout the database and therefore non-informative. In contrast our method provides keywords that are informative and descriptive of the specific subject matter.

This output, as shown in the results section of this paper, in and of itself, provides valuable support for gene analysis. Still, we further extend it in the next phase, to assist in finding relations among the genes.

Finding Functional Relations among Genes

Obviously, establishing firm functional relationships between genes requires performing carefully designed experiments. However, the literature can be used to suggest possible relations and to provide coherent justification for these suggestions. In the following we describe our approach for utilizing the literature in this manner.

Our primary assumption, which is justified by our results, is that common relevant literature is a strong indicator of common functionality. That is, genes which have similar lists of top ranking documents associated with them, share some common function that is described in the common literature.

Our task is thus reduced to finding similarities between the lists of documents retrieved in the previous phase of the algorithm, and to associating with each gene all the other genes that have similar document lists. To do this we use the *PubMed identifiers* associated with the documents, without examining the documents' contents. Using the identifiers alone, we construct for each kernel a vector characterizing it based on the documents deemed relevant to it by the first phase of the algorithm. Using this vector representation, we can rank, for each kernel K_i, all the other kernels according to their proximity to K_i in the kernel-vector space. Since each kernel corresponds to a gene, we can map the inter-related kernels back to their respective genes, and obtain a set of genes that are closely related. The method is illustrated at the bottom part of Figure 2 and is further described in the following paragraphs.

First, we construct the set of *PubMed* Identifiers of relevant documents, S_r, as follows:

Let N be the number of kernel documents used for representing genes[3]. We denote each kernel document by K_i where $1 \leq i \leq N$.

For each kernel, K_i, let L_i be the set of *PubMed* identifiers for the 50 top ranking documents associated with kernel K_i, formally: $L_i \overset{\text{def}}{=} \{ID_1^i \ldots ID_{50}^i\}$,
where ID_j^i is the *PubMed* identifier of the j^{th} document ranked as relevant for kernel K_i.

Intuitively speaking, if two distinct genes, G_i and G_j, represented by kernels K_i and K_j, have similar sets of relevant *PubMed* identifiers, L_i and L_j, then the literature relevant to these two genes has a lot in com-

the required information is present in the database.

[3] The number of genes we are analyzing may *exceed* N since the same kernel document might discuss and represent more than a single gene.

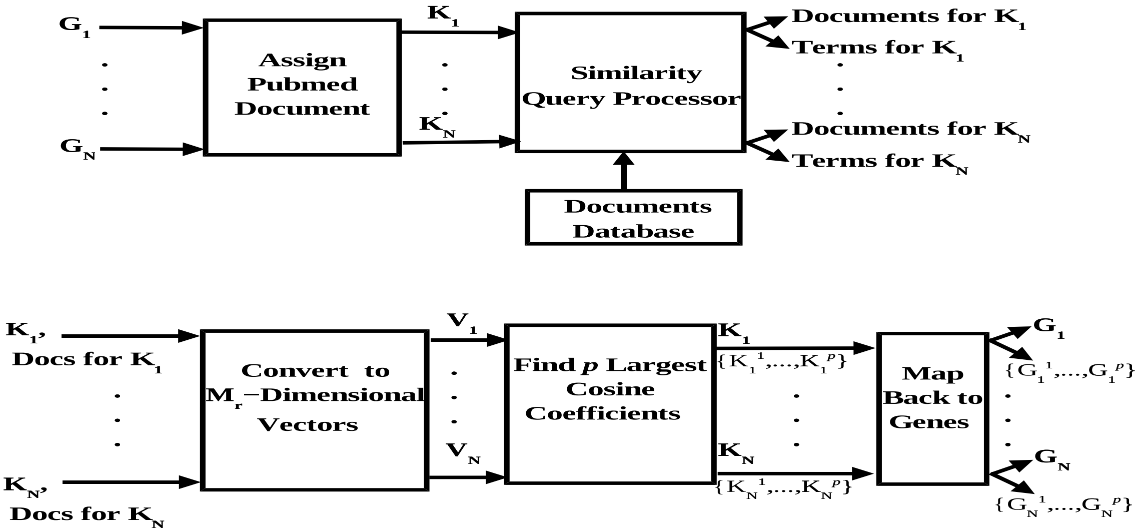

Figure 2: Finding Documents and Terms related to Genes (top), and Sets of Related Genes (bottom).

mon. This in turn suggests that some roles and functions (which are typically described in the literature) are shared by these two genes.

Note that when looking for similarities between lists of *PubMed* identifiers, identifiers that occur only within a single list L_i, and do not occur in any other list, L_j, do not contribute to the evaluation of L_j as similar to L_i. Using this observation, we can reduce the number of *PubMed* identifiers used for comparing document lists. Formally, let *ID* denote a *PubMed* identifier and $|ID|$ denote the total number of identifier lists, L_i, in which *ID* occurs. Our calculations need only take into account those identifiers for which $|ID| > 1$.

Thus, S_r is defined to be the set of *PubMed* identifiers of all documents that are in the relevance list of at least two kernels. Formally:

$$S_r \overset{\text{def}}{=} \bigcup_{i=1}^{N} L_i - \{ID \mid |ID| \leq 1\} \quad . \tag{2}$$

We denote the number of *PubMed* identifiers in S_r, $|S_r|$, by M_r, and denote each *PubMed* identifier in S_r as ID^j where $1 \leq j \leq M_r$.

We can now represent each kernel document K_i, as an M_r-dimensional vector, $V_i \overset{\text{def}}{=} \langle v_i^1 \dots v_i^{M_r} \rangle$ over S_r where v_i^j are defined as follows:

$$v_i^j = \delta_{ij} \overset{\text{def}}{=} \begin{cases} 1 & \text{if } ID^j \in L_i \\ 0 & \text{otherwise} \ . \end{cases} \tag{3}$$

We then divide each such kernel vector by its length, (the length in this case is simply the square root of the number of non-zero entries), obtaining a normalized representation of the kernels as vectors of length 1.

To gauge the similarity between each pair of kernels, we calculate the *cosine coefficient* between their respective vectors. The cosine coefficient is a well understood measure often used in information retrieval to roughly assess similarity between documents, when documents are represented as vectors of terms (see, for instance, Salton (1989)). We use it here in a non-traditional context, where our vector represents the kernels based on other *documents* rather than *terms*. Formally, the cosine coefficient between two vectors, V_i, V_k, whose respective lengths are $||V_i||, ||V_k||$ is the cosine of the angles between the vectors and is defined as:

$$\cos(V_i, V_k) \overset{\text{def}}{=} \frac{\sum_{j=1}^{M_r} v_i^j \cdot v_k^j}{||V_i|| \cdot ||V_k||} \quad .$$

Since the vectors representing the kernels are normalized, their length is 1 and only the numerator needs to be calculated.

We note that the cosine coefficient is 0 whenever the vectors V_i and V_j are *orthogonal* (*independent* of each other), and 1 when $V_i = V_j$. Thus, the closer V_i and V_j are, the closer the coefficient is to 1. Hence, by calculating for each kernel vector, V_i, the cosine coefficients with respect to all other kernel vectors, V_j, we obtain for each kernel a ranking of how related it is to each of the other kernels, K_j.

By recalling that each kernel K_i corresponds in turn to a gene G_i we obtain a relationship between the respective genes. The reasoning for the assumed relationship is given by the lists of terms associated with the themes generated from the kernel documents, and thus the reasoning behind the suggested relationships can be easily checked.

It is left to be shown that the documents retrieved as relevant to the genes, the summaries obtained and the relationships implied by using our algorithms are indeed useful. The experiments and the results reported in the next section demonstrate that our methods are indeed capable of meeting these criteria.

Experiments and Results

The main goal of the methods presented in this work is to provide researchers with quality literature and concise contents summaries regarding genes. A secondary goal is to present and reveal (possibly yet-unknown) relationships among genes.

To check the performance of our algorithms we apply them to yeast genes, and show how our methods indeed find relevant documents and provide accurate summary terms. Moreover, we also discover meaningful relationships among the genes. We have chosen the yeast DNA microarray testbed since the validity of our methods can only be assessed by comparison of the results with existing summaries of biological information. The Saccharomyces Genome Database[4] (Cherry *et al.* 1998; Ball *et al.* 2000) and the Yeast Proteome Database (Costanzo *et al.* 2000), as well as the functional analysis given by Spellman *et. al.* (1998), are critical for rapid, objective evaluation of our results.

We realize, of course, that the fact that the yeast genes are well studied biases the literature in *PubMed* to include many abstracts discussing these genes. However, given that *PubMed* consists of abstracts only, which typically contain little explicit information about the connections among genes, it is obvious that our algorithms contribute a great deal, finding information that can not be easily and effectively obtained by any currently available means.

The rest of this section describes the experimental setting and reports the results obtained by applying our algorithms to the data. The quality of the results was verified through comparison to the functional groups of genes according to Spellman *et. al.* (1998). The portion of Spellman's table relevant to the results discussed here is shown in Table 1. The table categorizes the yeast genes according to their functionality (rows) and the phase in the cell-cycle in which they are expressed (columns).

Experimental Setting

The experiments presented here consist of applying our algorithms to yeast genome data, in an attempt to find relevant literature and gene relations for the yeast genes analyzed by Spellman *et. al.* (1998). The names of all the genes used by Spellman[5] were compared against the Saccharomyces Genome Database (SGD). Out of about 800 genes found by Spellman *et. al.* to be cell-cycle regulated, only 408 genes had curated *PubMed* references in the SGD, and our experiments concentrate on these 408 genes.

For each of the genes, the oldest reference cited in SGD was chosen to be the kernel document corresponding to the gene. Since some of the closely related genes share the same reference, we obtain 344 distinct kernel documents on which we test our algorithm.

The database used in our experiments is a subset of *PubMed*, consisting of 33,700 documents discussing yeast genes. It was constructed by taking the 344 kernel documents, and applying the current *PubMed* neighboring algorithm (Wilbur & Coffee 1994) to each of the kernel documents. Neighboring was applied again to all the resulting documents and then applied a third time to all the documents in the resulting set. The resulting database contained 42,335 documents which included 2,250 documents deemed relevant for our 408 target genes by the SGD curators (86% of the total curated documents as of August, 1999). Many of the 42,335 had a title only and no abstract, and we eliminated them from the database, resulting in a set of 33,700 yeast-related documents. We eliminated from these documents the Mesh term taggings typically associated with *PubMed* entries, as well as all the terms that occur in over 10% of the documents in the database or in 2 or fewer documents. All these terms are typically useless and may have detrimental effect when looking for descriptive keywords. Eliminating such terms improves both the quality of the results and the running time of the program.

As a first phase in our experiments, we applied our similarity search program, described in the previous section, to the 344 kernels, searching over the database of 33,700 abstracts. For each kernel, the program outputs a list of the top 50 related documents and a list of keywords describing the contents of this relevant set.

The next phase consists of looking for *relationships* among genes. For each of the kernels, the previous phase produced a list of 50 relevant documents. The first step in the current phase is to construct the set of relevant documents retrieved for *all* the kernels, eliminating duplicates. That is, if a single document is relevant to more than one kernel, it is still included in the set of relevant documents only once. We then eliminate all documents that are relevant for a single kernel only, as explained in the previous section. We are left with a set of 3063 documents that are relevant to 2 or more kernel documents, (this is the set S_r, defined in Equation 2).

We then represent each kernel as a 3063-dimensional vector (as specified in Equation 3), and use the cosine coefficient to measure similarity between each kernel and all the other ones. Each kernel is then converted back to the gene(s) for which it was curated. The genes that are grouped as similar according to our method are compared with the ones grouped by functionality according to Spellman's table (parts of which are shown in Table 1).

[4] SGD, the Saccharomyces Genome Database can be accessed at *http://genome-www.stanford.edu/Saccharomyces* and YPD, the Yeast Proteome Database, at *http://www.proteome.com/databases/index.html.*

[5] Available through the genome web site at Stanford, http://genome-www.stanford.edu/cellcycle/ .

Biological Function	**G1**	**S**	**G2**	**M**	**M/G1**
Replication Initiation	CDC45		ORC1	CDC47 CDC54 MCM2 MCM6	CDC6 CDC46 MCM3
Fatty Acids/ Lipids/ Sterols/ Membranes	EPT1 LPP1 PSD1 SUR1 SUR2 SUR4		AUR1 ERG3 LCB3	ERG2 ERG5 PMA1 PMA2 PMP1	ELO1 FAA1 FAA3 FAA4 FAS1
Nutrition	BAT2 PHO8		AGP1 BAT1 GAP1	DIP5 FET3 FTR1 MEP3 PFK1 PHO3 PHO5 PHO11 PHO12 PHO84 RGT2 SUC2 SUT1 VAP1 VCX1 ZRT1	AUA1 GLK1 HXT1 HXT2 HXT4 HXT7

Table 1: Yeast Genes: expression during cell-cycle and functionality. (Adapted from Spellman *et. al.* (1998))

To check the validity of the keyword list assigned to each kernel, we compare each keyword to its associated functionality using a mini-thesaurus obtained from a panel of four independent yeast experts. Each functionality description listed in Spellman's table (such as *Secretion* or *Chromatin*) is associated with the terms judged most closely related to it according to the experts. Each expert received a list of the 22 function descriptions listed by Spellman *et al*, and a separate list of 330 alphabetically-sorted summary terms resulting from our program. The experts assigned to each term in the latter list, the functionality descriptors that they judged to be most related to it; non-specific terms were left unassigned. An example of two entries in the resulting thesaurus is shown in Table 2.

Function	**Associated Terms**
Chromatin	*chromatids, chromatin, chromosome, sister chromatids, telomere, telomeric*
Secretion	*acid phosphatase, coatomer, endoplasmic endoplasmic reticulum, er, golgi apparatus golgi complex, golgi transport, golgi, v snare*

Table 2: Example of thesaurus entries associating gene function with related terms.

For each gene, we compare its functionality according to Spellman with the functionality assigned by the panel to each of its key terms, counting how many of the key terms indeed correspond to the gene's functionality according to Spellman and how many do not. The results are described throughout the rest of this section.

Results

As stated before, for each gene represented by a kernel document we obtain through the similarity query mechanism applied to the whole database:

1. A set of related documents.
2. A set of summarizing keywords.

In addition, from the set of related documents we obtain, for each kernel, through the vector representation and the cosine coefficient calculation, a set of related kernels. The latter kernels are mapped back to form a set of related genes.

To assess the value of the results obtained in the first phase we examine the set of summarizing keywords. (Obviously, objectively assessing the quality of the retrieved documents themselves would also be desirable but there is no well-defined way to do it.) We also examine the lists of related genes obtained in the second phase. The quality of the results is checked through a comparison with the functionality assigned to genes by Spellman *et. al.*, shown in Table 1. Since many of the genes in the experiment are not assigned any functionality by Spellman (120 out of the 344 kernels used) , we can only verify in this manner results for the ones whose functionality was determined by Spellman *et. al.*

An example of a typical successful search is shown in Table 3. The left column of the table lists the *PubMed* identifiers for two kernel documents together with the genes they stand for and the functionality of these genes according to Spellman *et. al.* The second column lists, for each of the two kernels, the 10 top keywords associated with the retrieved set of documents, as determined by our algorithm. The third column lists the top 10 genes[6] associated with each of the two kernels, based on the cosine coefficient. The fourth column lists the function of each gene according to Spellman *et. al*, as a mean

[6] ELO1 has only 9 genes associated with it, since there were only 9 non-zero cosine coefficients associated with its kernel.

Kernel (PMID, Gene,Function)	**Keywords**	**Assoc. Genes**	**Function**
8702485 **ELO1** **Fatty Acid/ Lipids/ Sterols/ Membranes**	fatty acid,	OLE1	(Fatty Acid, Sterol. Met.)*
	fatty,	FAA4	Fatty Acid/Lipids/Sterols/Membranes
	lipids,	FAA3	Fatty Acid/Lipids/Sterols/Membranes
	acid,	SUR2	Fatty Acid/Lipids/Sterols/Membranes
	grown,	FAA1	Fatty Acid/Lipids/Sterols/Membranes
	medium,	ERG2	Fatty Acid/Lipids/Sterols/Membranes
	carbon,	PSD1	Fatty Acid/Lipids/Sterols/Membranes
	synthase,	CYB5	(Fatty Acid, Sterol. Met.)*
	strains,	PGM1	(Carbohydrates Met.)*
	deficient		
7651133 **HXT7** **Nutrition**	hexose,	HXT1	Nutrition
	glucose uptake,	RGT2	Nutrition
	glucose conc.,	HXT4	Nutrition
	fructose,	HXT2	Nutrition
	glycolytic,	GLK1	Nutrition
	glucose,	SEO1	(Small Molecules Transport)*
	sugars,	PRB1	(Protein Degradation)*
	uptake,	AGP1	Nutrition
	aerobic,	ZRT1	Nutrition
	utilization	MIG2	(Carbohydrates Met.)*

Table 3: Example of a result obtained from two different kernel/gene using our algorithm, compared with functionality according to Spellman or YPD (YPD functionality denoted by *).

for checking the validity of our results. Since our experiment included more genes than listed in Spellman's table, some of the genes in the third column are not assigned functionality by Spellman. For these genes, (denoted by an * in the table), we found the functionality in YPD.

The table shows that except for two genes (PGM1 and PRB1) all of the genes found for these two kernels have a strong functional relationship to the genes represented by the kernels, and the keywords provide a strong indication of this functionality. (Note that the keywords are associated as a *set* with the whole kernel entry and not separated as one keyword per associated gene.) We note that PGM1 is involved in carbohydrates metabolism which is still functionally related to fatty acids metabolism. PRB1 is responsible for protein degradation, which is not related to nutrition. It is included in this set, since the abstract chosen for its kernel document discusses regulation of the enzyme *prb1p* by glucose, rather than the function of *prb1p*.

The results for about 100 out of the 220 kernels for which we had the Spellman assigned functionality, closely resemble the ones demonstrated in Table 3 in the strong agreement with Spellman's cluster assignment and in the accurate description as given by the keywords learned by the similarity query algorithm. As a *quantitative* measure, we calculated the average number of *correct* and *incorrect* keywords among the 5 top-ranking keywords associated with each of these kernels. A keyword occurring in a list for a specific gene (kernel), is considered *correct* if it appears in our thesaurus entry labeled by the same function as the one assigned to the gene by Spellman. If its thesaurus entry is labeled by a *different* function, it is considered *wrong*. If it was assigned no function by our panel of experts it is considered *non-descriptive*. An average of *3.27* out of the 5 top ranking keywords, were associated with the *correct* function, while only *1.12* out of the 5 were associated with the wrong function, and *0.61* out of the 5 were non-descriptive. The difference between the high rate of correct keyword assignment relative to the wrong and the non-descriptive assignment is highly statistically significant ($p \ll 0.005$, according to the two-sample t-test).

For many other kernels the groups of related genes contain many genes not assigned functionality by Spellman, which makes the results harder to validate. Another set of cases, in which our results deviate from Spellman's functionality grouping of genes, are those for which the kernel document was not primarily focused on the function of the gene but contained a lot of detail discussing the experimental methods. In such cases, any document describing the same experimental method was considered similar and drawn into the set of relevant documents, resulting in a mixture of biologically-unrelated documents. The terms included in the keywords list indicate potential problems with this grouping and provide a warning that these results should not be taken at face value. An example of such a result is given in Table 4. In this case, the kernel document focuses on the technique used for studying the

Kernel (PMID, Gene,Function)	**Keywords**	**Assoc. Genes**	**Function**
6323245 **MCM2,MCM3,MCM6** **Replication Init.**	ars,	CDC10	Site Selection/Morphogenesis
	autonom. replicating,	PHO3	Nutrition
	replicating sequence,	EST1	DNA Syn
	autonomously,	MIF2	Chromatin
	minichromosomes,	PHO12	Nutrition
	replicating	POL2	DNA Syn.
	centromeric	DHS1	DNA repair
	leu2,	SNQ2	*
	plasmids,	SMC3	Chromat. Cohes.
	ura3,	EXG2	Cell Wall Synt.

Table 4: Example of a result obtained from an uninformative kernel using our algorithm, compared with functionality according to Spellman.

MCM genes, rather than the explicit function of these genes. Consequently, some of the kernels considered similar to it represent the use of similar techniques for studying different biological processes, rather than the biology of their associated genes. The result is a set of genes for which the commonality is that the documents curated for them all discuss manipulations within chromosomes rather than gene function. The keyword list (which highly ranks terms such as *autonomous replication* and contains *leu2* and *ura3* that are commonly used selectable markers for plasmids), indicates that the theme underlying this set of documents and genes is not relevant to functional genomics.

Obviously, obtaining good biological information (as shown in Table 3) is much preferable to an indication of poor quality, and for the most part this depends on starting from good quality kernel documents. The excellent experience with the 100 high-quality kernel documents demonstrates that once a single informative document is given for a gene, many other quality documents about the related genes are automatically found, accompanied by a succinct summary of the functional relationship between the genes.

Conclusions and Ongoing Work

Automatically finding connections among documents discussing genes has three clear advantages:

1. It is an efficient way for establishing putative relationships between genes as a preliminary step preceding direct experimental methods.
2. It provides the relevant literature needed by the researchers for performing the results analysis.
3. It generates a summary explaining the discovered relationships. This summary can help researchers explain and evaluate the relationships found through direct clustering of the expression levels.

Thus, this method can be used both for generating hypotheses prior to the experiments, as well as for post-experimental interpretation of the results.

The results presented in this paper demonstrate that given a functionally descriptive kernel document our program can provide insight into gene functional groupings, similar to that currently obtained through laborious, manual literature surveys relying on a lot of human expertise. Obviously our method can not ascribe function to genes which have not yet been studied. However, it can indicate functional relationships among known genes which heretofore have gone unnoticed.

The main limitation our technique currently faces is that of obtaining functionally descriptive kernel documents. We are considering several machine-learning techniques that can greatly assist in automating the kernel selection process. The expectation is that such kernel selection would consistently lead to good results.

Our method complements current techniques used for cluster analysis of the expression array data. We strongly believe that by combining this approach with techniques such as the one suggested by Friedman *et. al.* , as well as with expression array clustering approaches, we can achieve a great deal of automation and expedite the tedious task of analyzing the overwhelming amounts of data generated from experiments conducted over gene expression arrays.

Acknowledgments

We are grateful to Jan Fassler, Ken Katz, Steven Sullivan and Tyra Wolfsberg for the time and effort they have put into assigning functional tags to terms.

References

Ball, C. A. *et al.* 2000. Integrating functional genomic information into the Saccharomyces genome database. *Nucleic Acids Res.* 28:77-80.

Bassett, D. E.; Eisen, M. B.; and Boguski, M. S. 1999. Gene expression informatics – it's all in your mine. *Nature Genetics* 21:51–5.

Ben-Dor, A., and Yakhini, Z. 1999. Clustering gene expression patterns. In *Proceedings of the Third Annual Inter-*

national Conference on Computational Molecular Biology (RECOMB99).

Ben-Dor, A.; Shamir, R.; and Yakhini, Z. 1999. Clustering gene expression patterns. *Journal of Computational Biology* 6(3/4):281–297.

Blaschke, C.; Andrade, M. A.; Ouzounis, C.; and Valencia, A. 1999. Automatic extraction of biological information from scientific text: Protein-protein interactions. In *Proceedings of the AAAI Conference on Intelligent Systems in Molecular Biology*, 60–67.

Brown, P. O., and Botstein, D. 1999. Exploring the new world of the genome with DNA microarrays. *Nature Genetics* 21:33–7.

Cherry, J. M.*et. al.* 1998. SGD: saccharomyces genome database. *Nucleic Acids Res* 26:73–9.

Cho, R. J. *et. al.* 1998. A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol Cell* 2:65–73.

Chu, S. *et. al.* 1998. The transcriptional program of sporulation in budding yeast. *Science* 282:699–705.

Costanzo, M. C. *et. al* 2000. The yeast proteome database (YPD) and caenorhabditis elegans proteome database (WormPd): comprehensive resources for the organization and comparison of model organism protein information. *Nucleic Acids Res* 28:73–6.

Craven, M., and Kumlien, J. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the AAAI Conference on Intelligent Systems in Molecular Biology*, 77–86.

Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* 39(1):1–38.

DeRisi, J.; Iyer, V.; and Brown, P. 1997. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science* 278:680–686.

Eisen, M. B.; Spellman, P. T.; Brown, P. O.; and Botstein, D. 1998. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Science* 95:14863–14868.

Ermolaeva, O. *et. al.* 1998. Data management and analysis for gene expression arrays. *Nature Genetics* 20:19–23.

Ferea, T. L.; Botstein, D.; Brown, P. O.; and Rosenzweig, R. F. 1999. Systematic changes in gene expression patterns following adaptive evolution in yeast. *Proceedings of the National Academy of Science* 96:9721–6.

Friedman, N.; Linial, M.; Nachman, I.; and Pe'er, D. 2000. Using Bayesian networks to analyze expression data. *Life Sciences (to appear)*.

Gillespie, D., and Spiegelman, S. 1965. A quantitative assay for DNA-RNA hybrids with DNA immobilized on a membrane. *Journal of Molecular Biology* 12:829–42.

Leek, T. R. 1997. Information extraction using hidden Markov models. Master's thesis, Department of Computer Science, University of California, San Diego.

Rindflesch, T. C.; Tanabe, L.; Weinstein, J. N.; and Hunter, L. 2000. Edgar: Extraction of drugs, genes and relations from the biomedical literature. In *Proceedings of the Pacific Symposium on Biocomputing*.

Salton, G. 1989. *Automatic Text Processing*. Addison-Wesley.

Schena, M.; Shalon, D.; Davis, R. W.; and Brown, P. 1995. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science* 270:467–470.

Schena, M. 1999. *DNA Microarrays: A Practical Approach*. Oxford University Press.

Shatkay, H., and Wilbur, W. J. 2000. Finding Themes in MedLine Documents. In *Proceedings of the IEEE conference on Advances in Digital Libraries (ADL2000)*.

Southern, E. M. 1975. Detection of specific sequences among DNA fragments separated by gel electrophoresis. *Journal of Molecular Biology* 98:503–17.

Spellman, P. T. *et. al.* 1998. Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell* 9:3273–3297.

Tamayo, P. *et. al.* 1999. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Science* 96:2907–2912.

Tanabe, L. *et. al.* 1999. Medminer: An internet text-mining tool for biomedical information, with application to gene expression profiling. *BioTechniques* 27(6):1210–1217.

Wahl, G. M.; Meinkoth, J. L.; and Kimmel, A. R. 1987. Northern and southern blots. *Methods Enzymol* 152:572–81.

Wen, X. *et. al.* 1998. Large-scale temporal gene expression mapping of central nervous system development. *Proceedings of the National Academy of Science* 95:334–9.

Wilbur, W. J., and Coffee, L. 1994. The effectiveness of document neighboring in search enhancement. *Information Processing and Management* 30(2):253–266.