

Comparison of Two Schemes for Automatic Keyword Extraction from MEDLINE for Functional Gene Clustering

Ying Liu

College of Computing
Georgia Institute of Technology
Atlanta, Ga 30332-0280
yingliu@cc.gatech.edu

Brian J. Ciliax

Department of Neurology
Emory University Sch Medicine
Atlanta, Ga 30322
bciliax@emory.edu

Karin Borges

Department of Pharmacology
Emory University Sch Med
Atlanta, Ga 30322
kborges@pharm.emory.edu

Venu Dasigi

Southern Polytechnic State
University
Marietta, Ga 30060
vdasigi@spsu.edu

Ashwin Ram

College of Computing
Georgia Institute of Technology
Atlanta, Ga 30332-0280
ashwin@cc.gatech.edu

Shamkant B. Navathe

College of Computing
Georgia Institute of Technology
Atlanta, Ga 30332-0280
sham@cc.gatech.edu

Ray Dingledine*

Department of Pharmacology
Emory University Sch Med
Atlanta, Ga 30322
rdingledine@pharm.emory.edu

* Correspondence Author.

Abstract

One of the key challenges of microarray studies is to derive biological insights from the unprecedented quantities of data on gene-expression patterns. Clustering genes by functional keyword association can provide direct information about the nature of the functional links among genes within the derived clusters. However, the quality of the keyword lists extracted from biomedical literature for each gene significantly affects the clustering results. We extracted keywords from MEDLINE that describe the most prominent functions of the genes, and used the resulting weights of the keywords as feature vectors for gene clustering. By analyzing the resulting cluster quality, we compared two keyword weighting schemes: normalized z-score and term frequency-inverse document frequency (TFIDF). The best combination of background comparison set, stop list and stemming algorithm was selected based on precision and recall metrics. In a test set of four known gene groups, a hierarchical algorithm correctly assigned 25 of 26 genes to the appropriate clusters based on keywords extracted by the TDFIDF

weighting scheme, but only 23 of 26 with the z-score method. To evaluate the effectiveness of the weighting schemes for keyword extraction for gene clusters from microarray profiles, 44 yeast genes that are differentially expressed during the cell cycle were used as a second test set. Using established measures of cluster quality, the results produced from TFIDF-weighted keywords had higher purity, lower entropy, and higher mutual information than those produced from normalized z-score weighted keywords. The optimized algorithms should be useful for sorting genes from microarray lists into functionally discrete clusters.

1. Introduction

DNA microarray technology provides biologists with the ability to measure the expression levels of thousands of genes in a single experiment. As data from such experiments accumulate, it is essential to derive efficient methods to catalogue these genes into useful and functionally meaningful groups [1,2].

Many algorithms are available to cluster genes based on similarities in their expression profiles [2-4]. Although these approaches add interpretive value, the task of finding functional relationships between specific genes is left to the investigator. If, instead of organizing by expression pattern similarity, genes were grouped according to shared function, investigators might more quickly discover patterns or themes of biological processes that were revealed by their microarray experiments and focus on a select group of functionally related genes. A number of strategies based on shared functions rather than similar expression patterns have been devised to link information from medical literature with gene names [5-14]. Liu et al. [9] reported that keyword associations derived from MEDLINE abstracts could be used to cluster genes effectively. We believe that clustering genes by functional keyword associations should be useful for discovering novel relationships among sets of genes because it links them by shared functional keywords rather than just reporting known interactions based on published reports. Thus, genes that never co-occur in the same publication could still be linked by their shared keywords.

The value of the clustering result depends on the quality of the keyword lists extracted from the abstracts. Ideally, high quality keyword lists for gene identification should be able to distinguish certain individual genes from others. Various weighting schemes have been developed to determine the importance of a word to a document. The “z-score” method, a statistical profiling approach that accepts user-supplied abstracts related to a protein of interest and returns an ordered set of keywords that occur in those abstracts more often than would be expected by chance [15], had been used by us [9] and others [5]. Term frequency-inverse document frequency (TFIDF) [16], one of the most widely used weighting schemes in the information retrieval research area, has also been applied to analyze biomedical literature to identify functionally coherent gene groups [12].

In this paper, we first expand, extend, and optimize the z-score method by testing new background sets, a new stemming algorithm, and a new, extensive stop list customized for use with the biological literature. We then compare the performance of the zscore method with TFIDF for the purpose of extracting the functional keywords for each tested gene set by evaluating the quality of the gene clusters generated from gene-associated keywords using a hierarchical clustering algorithm.

2. Methods

2.1. Keyword extraction from biomedical literature

We used the z-score method and TFIDF to extract keywords from MEDLINE. These methods estimate the significance of words by comparing the frequency of words in a test (gene-related) set of abstracts with their frequency in a background set of abstracts.

Background Sets: The background sets of abstracts were used to build a hash table of words and their respective statistics for comparison with the corresponding words in the test sets. The background set used by Andrade and Valencia [15] consisted of abstracts associated with 71 protein families in the 1993 release of the PDBSELECT database. By the year 2000 this database had grown to 1155 protein families, 760 of which have >4 members. We used abstracts associated with the PDB-1155 and PDB-760 protein families, which have an average of 41 and 57 abstracts per family, respectively. A third background set was created consisting of 50,000 randomly selected MEDLINE abstracts sorted into 1000 pseudo-families of 50 abstracts each. Finally, we built a large random background set (approximately 112,000 pseudo-families of 50 abstracts each), which incorporated all the MEDLINE abstracts up to year 2000.

Test Sets: For each gene analyzed, word frequencies were calculated from a group of MEDLINE abstracts retrieved by an SQL search, in the TITLE field, for the specific gene name or any known aliases. The resulting set of abstracts was processed to generate a specific keyword list.

We used three test sets in our comparisons. The first group of genes was used to evaluate the keyword identification algorithms by precision-recall and error-minimization tests as described below. We evaluated the accuracy of the keyword-selection algorithms by comparing their output with the set of keywords selected by three knowledgeable investigators from the same set of abstracts. For each of 10 genes with diverse biological functions (adenylate cyclase, androgen receptor, calmodulin, caspase-3, dopamine D2 receptor, GluR2 AMPA receptor subunit, glutamic acid decarboxylase-65, histone H4, L-type calcium channel, and tyrosine hydroxylase), we retrieved a set of abstracts by a simple search for the gene name in the citation TITLE field (limited to the 10 most recent citations for each set). These 10 sets of 10 abstracts each were processed for keyword selection by the two

weighting schemes. These abstracts were also hand-processed by three authors (KB, BJC, and RD), who selected non-methodological keywords that are

reflective of the biological functions described in each abstract.

The second group of genes was clustered by keyword associations. We selected 26 genes in four

Table 1. Gene sets manually clustered based on functional similarity

Group	Genes	Functions
1	<i>GluR1, GluR2, GluR3, GluR4, GluR6, KAI, KA2, NMDA-R1, NMDA-R2A, NMDA-R2B</i>	Glutamate receptor channels
2	<i>Tyrosine hydroxylase, DOPA decarboxylase, Dopamine beta-hydroxylase, Phenethanolamine N-methyltransferase, Monoamine oxidase A, Monoamine oxidase B, Catechol-O-methyltransferase</i>	Catecholamine synthetic enzymes
3	<i>Actin, Alpha-tubulin, Beta-tubulin, Alpha-spectrin, Dynein</i>	Cytoskeletal proteins
4	<i>Chorismate mutase, Prephenate dehydratase, Prephenate dehydrogenase, Tyrosine transaminase</i>	Enzymes in tyrosine and phenylalanine synthesis

Table 2. 44 Yeast Genes grouped by transcriptional activators and cell cycle functions [1]

Group	Activators	Genes	Functions
1	Swi4, Swi6	<i>Cln1, Cln2, Gic1, Gic2, Msb2, Rsr1, Bud9, Mnn1, Och1, Exg1, Kre6, Cwp1</i>	Budding
2	Swi6, Mbp1	<i>Clb5, Clb6, Rnr1, Rad27, Cdc21, Dun1, Rad51, Cdc45, Mcm2</i>	DNA replication and repair
3	Swi4, Swi6	<i>Htb1, Htb2, Hta1, Hta2, Hta3, Hho1</i>	Chromatin
4	Fkh1	<i>Hhf1, Hht1, Tel2, Apr7</i>	Chromatin
5	Fkh1	<i>Tem1</i>	Mitosis control
6	Ndd1, Fkh2, Mcm1	<i>Clb2, Ace2, Swi5, Cdc20</i>	Mitosis control
7	Ace2, Swi5	<i>Cts1, Egt2</i>	Cytokinesis
8	Mcm1	<i>Mcm3, Mcm6, Cdc6, Cdc46</i>	Prereplication complex formation
9	Mcm1	<i>Ste2, Far1</i>	Mating

well-defined functional groups consisting of ten glutamate receptor subunits, seven enzymes in catecholamine metabolism, five cytoskeletal proteins and four enzymes in tyrosine and phenylalanine synthesis (Table 1). This experiment was performed to determine the quality of the keywords derived from the two weighting schemes. We tested whether the clustering algorithm can group genes appropriately into the four gene families or clusters that were known

a priori simply based on associations among the keywords derived in the previous step.

The third group was used for determining the keyword quality by testing whether the keywords could be used to group genes identified in microarray experiments. We selected 44 yeast genes involved in the cell cycle of budding yeast (*Saccharomyces cerevisiae*) that had altered expression patterns on spotted DNA microarrays [2]. These genes have been analyzed by Cherepinsky et al. [1] and a master

list of member genes for each cluster was assembled according to the common cell-cycle functions and regulatory systems inferred from the roles of various transcriptional activators [1] (Table 2).

2.1.1 Stemming. Word stemming is used to truncate suffixes and trailing numerals so that words having the same root (e.g., activate, activates, activation, and active) are collapsed to the same word for frequency counting. Two stemming algorithms were compared, one used by Andrade and Valencia [15], and one devised by Porter [17]. A third condition, in which the words were not stemmed, was used as a control.

2.1.2 Stop-word Lists. Stop-word lists are typically used to filter out non-scientific English words that carry low domain-specific information content. We tested two stop-word lists initially: a simple list of 319 common English words (http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/stop_words), and an online dictionary of 22,205 words (<http://ftp.std.com/obi/Dictionary/dict>). Our initial tests led us to add methodological words that are unrelated to gene or protein function to the online dictionary, and to remove selected words. This resulted in a stop-word list customized for biological applications. This stop-list, abbreviated PD+ (pocket dictionary plus), is evolving as we delete more biological or functional words and add methodology words. We also analyzed keywords without applying a stop-word list, which served as the control condition.

2.2 Keyword Assessment

2.2.1. Z-score method. Statistical formulae from Andrade and Valencia [15] for word frequencies and z-scores were used without modification. The weight of word a for gene g is represented by the zscore, and is defined as:

$$Z_g^a = \frac{F_g^a - \bar{F}^a}{S^a} \quad (1)$$

where F_g^a equals the document frequency of word a in test set (gene) g and, as defined by [15], \bar{F}^a and S^a are the average frequency and standard deviation, respectively, of word a in the background set. For the random background set, the document frequencies of word a across pseudo-families of 50 randomly-selected abstracts each were used to calculate these latter metrics instead of the

proportions of proteins in individual families for which word a appears in at least one representative abstract [15].

2.2.2. TFIDF method. The standard TFIDF function was used [16]. TFIDF combines term frequency (TF), which measures the number of times a word occurs in the gene's set of abstracts (reflecting the importance of the word to the gene), and inverse document frequency (IDF), which measures the information content of a word – its rarity across all the families in the background set. The inverse document frequency (IDF) is calculated as:

$$idf^a = \log \frac{N}{df^a} \quad (2)$$

where idf^a denotes the inverse document frequency of word a in the background set; df^a denotes the number of families (or pseudo-families) in which word a occurs; and N is the total number of families or pseudo-families in the background set.

TFIDF is defined as:

$$tfidf_g^a = tf_g^a \times idf^a \quad (3)$$

$tfidf_g^a$ denotes the weight of the word a to the gene g ; tf_g^a the number of times word a occurs in the set of abstracts for gene g .

To distribute the word weights over the [0, 1] interval, the weights resulting from TFIDF were often normalized by *cosine normalization*, given by

$$Weight_g^a = \frac{tfidf_g^a}{\sqrt{\sum_{s=1}^{|W|} (tfidf_g^s)^2}} \quad (4)$$

where $|W|$ denotes the number of words in the abstracts of gene g .

2.2.3. Normalized z-score method. In order to compare with TFIDF, the z-scores of the words were also normalized (normalized z-score method) as:

$$Weight_g^a = \frac{Z_g^a}{\sqrt{\sum_{s=1}^{|W|} (Z_g^s)^2}} \quad (5)$$

The weight of a word is assigned the value “New”,

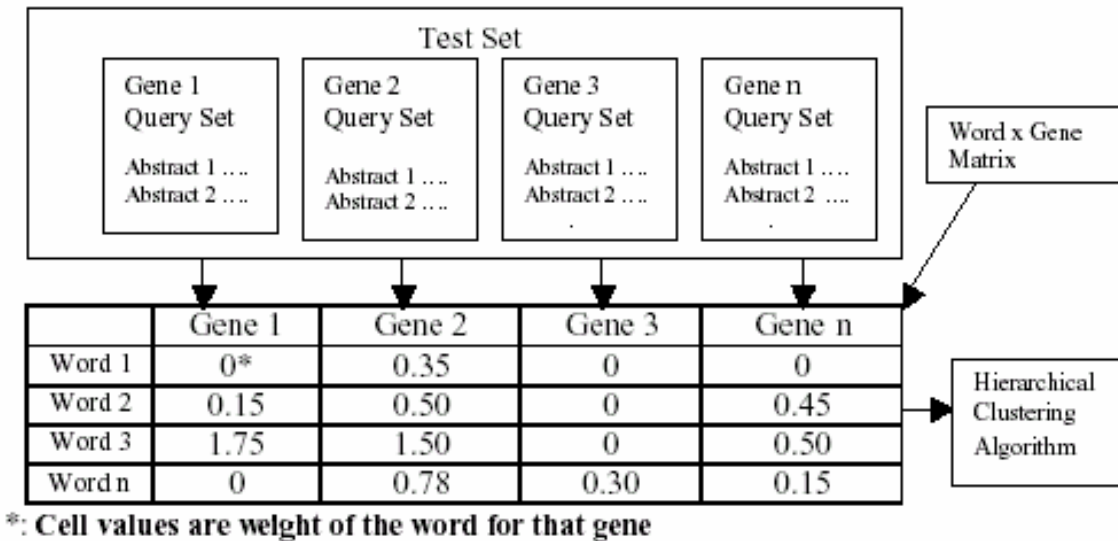


Figure 1. Procedure for clustering genes by the strength of their associated keywords.

if the word occurs in the test set but not in the background set, since no background statistics are available from which to calculate the zscore or tfidf values.

2.3 Precision-Recall and Error-Minimization

Using the keyword lists generated from the first test set, investigator-derived lists were used as the standard against which the algorithm-derived lists were evaluated by Precision and Recall measurements. Precision (P) and Recall (R) are the standard metrics for retrieval effectiveness in information retrieval. They were calculated as follows:

$$P = \frac{tp}{(tp + fp)} \quad R = \frac{tp}{(tp + fn)}$$

Where: **tp** = words in the algorithm-derived list also found in the investigator-derived list; **fp** = words in the algorithm-derived list not found in the investigator-derived list; **fn** = words in the investigator-derived list not found in the algorithm-derived list. They stand for true positive, false positive, and false negative, respectively.

The optimum combination of the parameters (different background sets, stemming algorithms, and stop lists) plus the zscore threshold for accepting a word was determined by minimizing the function: $E = V * (1 - P) + (1 - R)$ [18]. If $V > 1$, the cost of false positives is weighted more heavily than the cost of false negatives. We selected $V = 4$ empirically to limit the number of irrelevant words when classifying gene

function.

2.4 Keyword selection for gene clustering

We used word weight thresholds to select the keywords used for gene clustering. Those keywords with weights less than the thresholds were discarded. The outputs of the keyword selection for all genes in the second and the third test set are represented as sparse keyword (rows) x gene (columns) matrices with cells containing word weights.

2.4.1. Gene Clustering by Associated Keywords. The hierarchical clustering analysis were performed using Cluster/Treeview programs available online (<http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm>) [2]. Genes were grouped using the average linkage hierarchical clustering algorithm.

2.4.2. Evaluating the clustering results. To evaluate the quality of our resultant clusters, we used the established metrics of Purity, Entropy and Mutual Information [19]. Higher purity (best value is 1), lower entropy (best value is 0), and higher mutual information indicate better quality of the cluster result [19]. Mutual information is a measure of the concordance between the algorithm-derived clusters

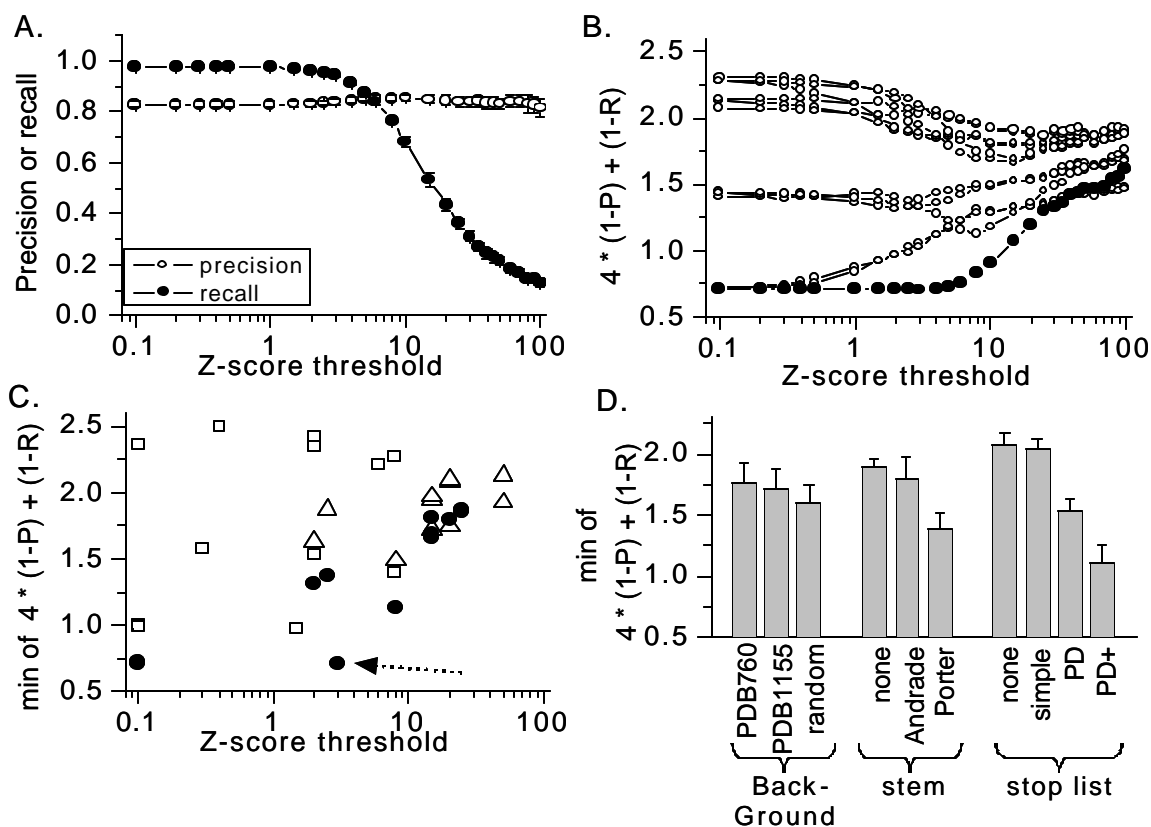


Figure 2. Evaluation and optimization of the keyword selection algorithm. A) Precision and recall as a function of the z-score threshold for accepting a word. B) The optimization function is plotted for each parameter set that includes the Porter strong stemmer. Solid circles represent data from the random background set and the PD+ stop list. Open circles show the other 11 conditions. C) The minima of the optimization function were determined from plots in B) and are plotted against the corresponding z-score for all parameter combinations. Solid circles = Porter stemmer, open boxes = the stemmer described in [15], and open triangles = no stemming. The arrow points to the optimum combination of parameters, which involve a zscore threshold > 3 and the combination of Porter stemmer, random background set and PD+ stop list. D) The sensitivity of the algorithm performance to changes in each parameter was systematically evaluated by calculating the mean ($\pm SEM$) of all optimization function minima in a data set, holding each parameter constant in turn. Performance was most affected by the stop list.

and the actual clusters. It is the measure of how much information the algorithm-derived clusters can tell us to infer the actual clusters. Random clustering has mutual information of 0 in the limit. Higher mutual information indicates higher similarity between the algorithm-derived clusters and the actual clusters. Entropy and mutual information had been used to build the relevance network in functional genomics [20].

2.4.3. Effect of breakpoint on cluster quality.

Hierarchical clustering organizes expression data into

a binary tree without providing clear boundaries between clusters. In practice, investigators define clusters by a manual scan of the genes in each node and rely on their biological expertise to notice shared functional properties of genes. To determine the effect of the breakpoint selected on the cluster quality, we used the Mutual Information metric to evaluate clusters created by different breakpoints.

3. Results

3.1. Optimization of the keyword selection algorithm.

We applied techniques prevalent in the field of information retrieval and text analysis to generate a matrix that contains feature vectors for genes of interest. The performance of this system depends on the weighting scheme, the background set of abstracts, the stemming algorithm, and the stop list selected. An overview of the keyword identification process is provided in Figure 1.

The performance of the keyword-selection weighting schemes was evaluated initially by comparing their output with the set of keywords selected by human investigators from an identical set of 100 abstracts. The statistical algorithms used 1008 combinations of three background sets: PDB-1155, PDB-760, and random families; three stemming rules: none, weak [15], and strong [17]; four stop lists: none, simple stop list of 319 words, a 22,205 word online pocket dictionary (PD), and the supplemented pocket dictionary named PD+; and 28 z-score thresholds for accepting a keyword as being associated with the gene. A word was deemed to be associated with a gene by the algorithm only if the weight was above a user-set threshold. The investigator-derived lists were then used as the standard for evaluation of the algorithm-derived lists. For each combination of parameters we used the typical metrics of Precision (P) and Recall (R) to evaluate algorithm performance.

For the case in which 1000 families in the random background list were stemmed by the Porter algorithm and filtered by the PD+ stop list, as word selection became more stringent (increasing z-scores), recall fell but precision was nearly unaffected (Figure 2A). Examination of all P-R plots indicated that the extensive stop list was primarily responsible for the relatively flat precision because less extensive stop lists caused low precision at low z-score values. Figure 2B plots the error minimization function with $V=4$ for all 12 parameter sets that included the Porter strong stemming algorithm, and Figure 2C plots the minimum of this function against the z-score threshold for each parameter combination. Overall the best performance was achieved with the random background set, Porter strong stemming, the PD+ stop list and a z-score acceptance threshold of 3-8 for V ranging between 2 and 4.

Examination of Figure 2C shows that the stronger stemming algorithm (Porter, solid circles) often

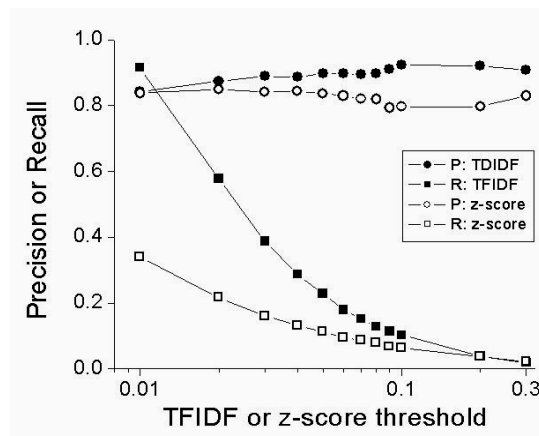


Figure 3. Keyword extraction by two weighting schemes (TFIDF and normalized z-score). Precision and recall is plotted as a function of the weight threshold for accepting a word.

performed better than the weaker stemming algorithm (open squares) or no stemming (open triangles). To determine which parameter (background set, stemming algorithm or stop list) exerts the most influence on the performance of the algorithm, we calculated the mean value of the optimization function with each parameter being fixed in turn. Figure 2D shows that a stringent stop list (PD+) is most important for optimizing the algorithms, followed by a strong stemming algorithm. Selection of the background set had relatively little effect on the performance of the keyword selection algorithm, which indicates that as long as the weight of a word is reduced if it occurs commonly in a fairly large set of MEDLINE abstracts, it may be less important how that set is chosen.

Therefore, the best keyword selection performance for the z-score scheme utilizes a random background set, the PD+ stop list and Porter's stemming algorithm. To preclude the occurrence of the "New" words, which occur in the test but not background sets, we created a large random background set (about 112,000 pseudo-families of 50 abstracts each), which included all MEDLINE abstracts up to year 2000. For subsequent studies, we will be using the combination of this large random background set, PD+ stop list and Porter's stemming algorithm to extract keywords for each gene.

3.2. Comparison of TFIDF and Normalized

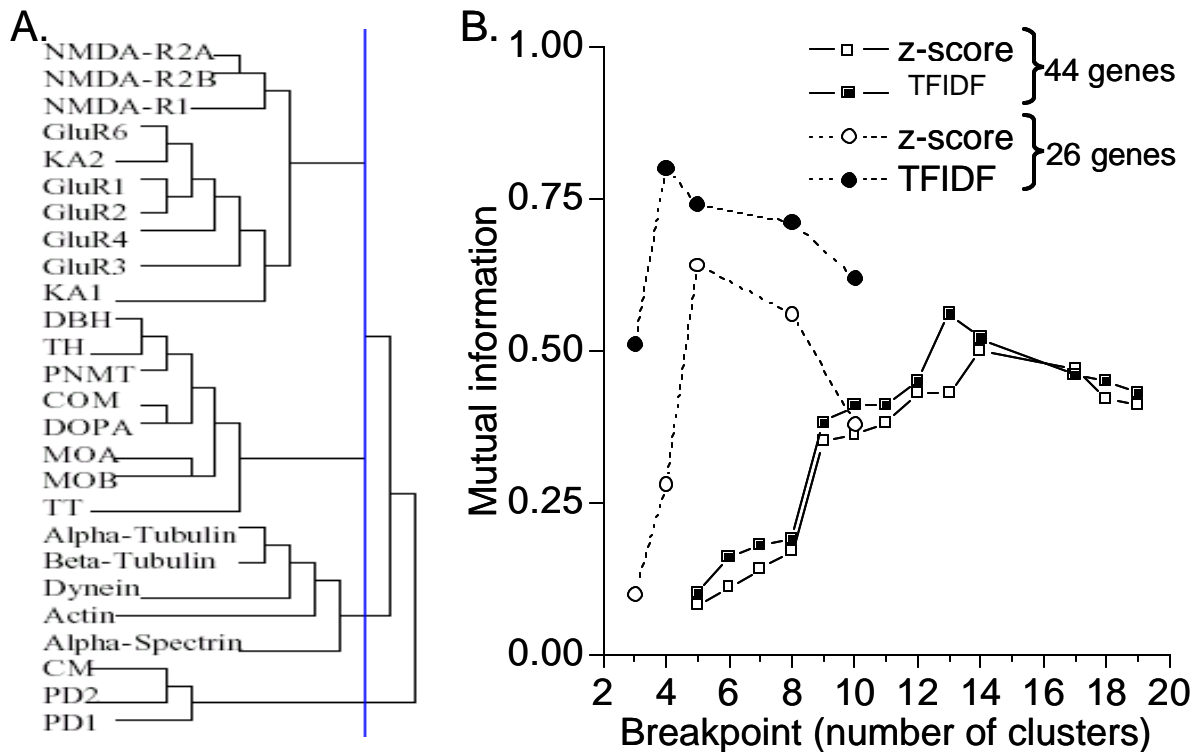


Figure 4. Gene clusters by keyword associations. Keywords with weight ≥ 0 were extracted from MEDLINE abstracts for 26 genes and 44 genes. The resulting word x gene sparse matrix was the input to the hierarchical clustering algorithm. A) The clustering result of 26 genes generated from TFIDF weighting scheme. The vertical line shows the breakpoint that creates four clusters. B) The effect of different breakpoints for cluster definition on the performance of TFIDF and normalized z-score weighting schemes. In paired t-tests of TFIDF vs z-score, TFIDF performed better regardless of cluster size for both 26-gene set ($P=.0216$) and the 44-gene set ($P=.0024$).

The top 10 genes in panel A are glutamate receptors. Other abbreviations. CM: *Chorismate mutase*; COM: *Catechol-O-methyltransferase*; DBH: *Dopamine beta-hydroxylase*; DOPA : *DOPA decarboxylase*; MOA: *Monoamine oxidase A*; MOB: *Monoamine oxidase B*; PD1: *Prephenate dehydratase*; PD2: *Prephenate dehydrogenase*; PNMT: *Phenethanolamine N-methyltransferase*; TH: *Tyrosine hydroxylase*; TT: *Tyrosine transaminase*.

Z-score Method.

The performance of keyword-selection by TFIDF and normalized z-score methods were also evaluated

with precision and recall metrics (Figure 3) by comparing the TFIDF and normalized z-score method outputs with the set of keywords selected by human investigators from an identical set of 100 abstracts (the first test set). Figure 3 shows that TFIDF outperforms the normalized z-score method with higher precision and recall values. Due to cosine normalization, the thresholds are much smaller in Figure 3 than those in Figure 2.

3.3. Gene Clustering by Keyword Associations.

3.3.1. 26-gene set. Keyword lists were generated for each gene. With a weight threshold 0, the resulting word x gene matrix generated by the normalized zscore method had 26 columns (genes) and approximately 5712 rows (words that appear in the test set), whereas that generated by TFIDF had 26 columns and 5741 rows. Taking the TFIDF generated word x gene matrix, the hierarchical algorithm correctly assigned 25 of 26 genes to the appropriate cluster based on the strength of keyword associations

(Figure 4A). TT (tyrosine transaminase) is the only gene that was incorrectly assigned. To determine the effect of the breakpoint for deciding the clusters on the performance of TFIDF and normalized z-score weighting schemes, we tested different breakpoints. The results are shown in Figure 4B. The clusters produced from TFIDF-generated keyword lists had higher mutual information than those produced from normalized z-score-generated keyword lists, regardless of the number of clusters created by using different breakpoints ($P < .02$ by paired t-test).

3.3.2. 44-gene set. Keyword lists were generated for each of the 44 yeast genes and a 1875 (words appearing in the test set) x 44 matrix was created by the normalized zscore method, whereas a 1874 x 44 matrix was created by TFIDF. The clusters produced from TFIDF-generated keyword lists, again, had higher mutual information than those produced from normalized z-score-generated keyword lists (Figure 4B), over the whole range of breakpoints ($P < .002$).

4. Discussion

We compared two weighting schemes, the normalized zscore method and TFIDF, for keyword extraction from MEDLINE citations. The results showed that TFIDF, as a weighting scheme, clearly outperforms an optimized and normalized z-score based approach in the quality of keywords extracted as judged by precision-recall analysis (Figure 3), and in the gene clustering task. For both of the two test sets, the cluster result produced by the hierarchical clustering algorithm from TFIDF-weighted keywords had higher mutual information than that produced from normalized zscore method-weighted keywords (Figure 4B). The single outlier gene, TT (tyrosine transaminase, Figure 4A) was assigned to the dopamine metabolism cluster rather than the tyrosine metabolism cluster, which reflects the strong functional similarity between these two gene groups.

4.1. Weighting schemes for keyword selection

Word weighting is an important step in information retrieval, text mining, and text categorization for indexing documents. The main function of a word-weighting scheme is to enhance retrieval effectiveness [16]. In gene clustering by functional keyword associations, the weighting

scheme is used to extract high quality keyword lists. Despite the variations in weighting schemes, the essential ideas on which they are based can be grouped into a few categories [25]: (1) A “word” which appears once in a document is likely to be a keyword for that document; (2) a “word” which appears frequently in a document is likely to be a keyword for that document; (3) a “word” which appears only in a limited number of documents is likely to be a keyword for any document in which it appears; (4) a “word” which appears relatively more frequently in a document than in the other documents is likely to be a keyword for that document; (5) a “word” which shows a specific distribution in a collection of documents is likely to be a keyword for that collection of documents.

Categories (1) and (2) emphasize the “representation” aspect of keywords, and categories (3) and (4) emphasize the “discrimination” aspect. While categories (1) to (4) focus on individual documents, category (5) takes into account the relationships among documents as seen from the overall distribution of words. Therefore, category (5) has the advantage of considering topics as represented by a group of documents, while categories (1) to (4) only treat each document as a basic topic unit. Accordingly, the weighting schemes based on category (5) vary considerably, both in theoretical viewpoints and in the resultant weights given to words [25]. TFIDF is based on categories (1) to (4) because it considers the representation and discrimination aspects of keywords by combining the term frequency and inverse-document frequency. On the other hand, the word distribution in the background set is also taken into account in the z score method because the word’s average frequency and standard deviation in the background set are used to calculate the z-scores. Andrade and Valencia [15] used a δ measure to present the distribution of the words in the background set. In their original z-score method, the abstracts in the background set were grouped by protein families, indicating that the abstracts inside a family were closely related. Therefore, it is reasonable to consider the relationship among families as seen from the overall word distribution. However, in the random background sets, the abstracts inside the pseudo-families were randomly chosen. Therefore, the word distribution among pseudo-families is meaningless. Our results show that TFIDF outperforms the normalized z-score method, indicating that the word distribution does not add any information to the metric.

4.2. Keyword selection algorithms

The use of keywords selected from gene-related literature to cluster functionally-related genes has two fundamental limitations. First, with the keyword selection algorithms described above, some words with high z -scores have low predictive potential for biological function or are erroneously associated with the gene in question [9]. Such results could occur more often when the gene name is referenced in the abstracts, but is not the actual topic of discussion, when the topic switches from the gene name to some other issue, or when the word “not” reverses the meaning of the sentence. Enhancements to the basic schemes could involve *i*) using natural language processing to exploit the added information in compound phrases, syntax, and grammatical structures such as negative sentences, and *ii*) improving our stop list. The sensitivity analysis (Fig 2D) indicates that the quality of the stop list is the most important element in algorithm performance. Second, inconsistency among human investigators in the task of agreeing upon keywords from a document places a fundamental limit on our ability to evaluate the performance of computer algorithms against human opinion. Keyword selection by an investigator is ultimately subjective and leads to ambiguities in document classification [21-24], with the consequence that performance better than ~75-80% precision may not be achievable.

For the reasons described above, the use of investigator-selected keywords as the “gold standard” for evaluating the performance of keyword-selection algorithms is imperfect. However, even in the face of these challenges, the keyword selection algorithms used here appear sufficiently robust to serve as the basis for functional gene clustering

4.3. The Effect of Cosine Normalization

A particular word is more likely to be repeated in a larger test set than in a shorter test set, and as a result, the term frequency of that word will be higher, which causes a higher TFIDF value since the IDF is the same. In our case, a larger test set means the gene has more abstracts and/or longer abstracts. Cosine normalization is applied in TFIDF so that the words in the longer documents are not unfairly given more weight. In order to compare with TFIDF, the z -score values were also normalized. In direct comparisons of cluster quality with keywords selected by the two

schemes, TFIDF outperformed the normalized z -score for both test sets of genes.

5. Acknowledgments

This work was supported by NINDS (RD) and the Emory-Georgia Tech Research Consortium. We would like to thank the Office of Genomics and Disease Prevention in Center of Disease Control and Prevention (CDC) for the financial support. We also would like to thank Brian Revennaugh for computer administration and technical support.

6. References

- [1] Cherepinsky, V. et al. (2003) Shrinkage-based similarity metric for cluster analysis of microarray data, Proc. Natl. Acad. Sci. USA, 100:9668-9673.
- [2] Eisen, M.B. et al. (1998) Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci USA, 95: 14863-14868.
- [3] Tamayo, P. et al. (1999) Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. Proc. Nat. Acad. Sci USA 96: 2907-2912.
- [4] Herrero, J. et al. (2001) A hierarchical unsupervised growing neural network for clustering gene expression patterns. Bioinformatics 17: 126-136.
- [5] Blaschke, C. et al. (2001) Mining functional information associated with expression arrays, Funct. Integr. Genomics, 1:256-268.
- [6] Chaussabel, D., and Sher, A. (2002) Mining microarray expression data by literature profiling, Genome Biology, 3:1-16.
- [7] Jenssen, T.K. et al. (2001) A literature network of human genes for high-throughput analysis of gene expression Nature Genetics. 28: 21-28.
- [8] Kankar, P. et al. (2002) MedMeSH summarizer: text mining for gene clusters. Proc SIAM International Conference in Data Mining, SDM 2002 (April 2002)
- [9] Liu, Y. et al. (2004) Text mining functional keywords associated with genes. MEDINFO 2004, 11th World Congress on Medical Informatics, in press.
- [10] Masys, D.R. et al. (2001). Use of keyword hierarchies to interpret gene expression patterns. Bioinformatics 17: 319-26.
- [11] Raychaudhuri, S. et al. (2003) The computational

- analysis of scientific literature to define and recognize gene expression clusters, *Nucleic Acids Research*, 15: 4553-4560.
- [12] Raychaudhuri, S. et al. (2002) Using text analysis to identify functionally coherent gene groups. *Genome Res.* 12: 1582-1590.
- [13] Swanson, D.R., and Smalheiser, N.R. (1997) An interactive system for finding complementary literatures: a stimulus to scientific discovery, *Artificial Intelligence* 91:183-203.
- [14] Tanabe, L. et al.(1999). MedMiner: an internet text-mining tool for biomedical information, with application to gene expression profiling. *Biotechniques* 27: 1210-1214.
- [15] Andrade, M., and Valencia, A. (1998) Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families, *Bioinformatics*, 14:600-607.
- [16] Salton, G., and Buckley, C. (1988) Text-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24:513-523.
- [17] Porter, M. (1980) An algorithm for suffix stripping, *Program*, 14:130-137.
- [18] Hvidsten et al. (2001) Predicting gene function from gene expression and ontologies. *Pacific Symposium on Biocomputing* 6:299-310.
- [19] Strehl, A. Ghosh, J. and Mooney, R. (2000) Impact of similarity measures on web-page clustering. *AAAI-2000, Workshop of Artificial Intelligence for Web Search*, 58-64.
- [20] Butte, A.J. and Kohane, I. S. (2000) Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. *Pacific Symposium on Biocomputing*, 5: 415-426.
- [21] Funk ME and CR Reid (1983) Indexing consistency in MEDLINE. *Bull. Med. Libr. Assoc.* 71: 176-183.
- [22] Blair, DC and ME Maron (1985) An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Communic for the Soc. for Computing Machinery*, 28: 289-299.
- [23] Swanson, D.R. (1960) Searching natural language text by computer. *Science* 132: 1099-1104.
- [24] Saracevic, T. (1991) Individual differences in organizing, searching and retrieving information. *Proc Amer. Soc. Information Science*, 28: 82-86.
- [25] Kageura, K., and Umino, B. (1996) Methods of automatic term recognition—a review. *Terminology*, 3: 259-289.