An Intelligent Biological Information Management System

Mathew Palakal
Computer Science Department
Indiana University Purdue
University Indianapolis
Indianapolis, IN 46202
Tel: (01) 317-274-9735

mpalakal@cs.iupui.edu

Snehasis Mukhopadhyay
Computer Science Department
Indiana University Purdue
University Indianapolis
Indianapolis, IN 46202
Tel: (01) 317-274-9732

smukhopa@cs.iupui.edu

Javed Mostafa
School of Library and
Information Science
Indiana University
Bloomington, IN 47405
Tel: (01) 812-855-6268

jm@indiana.edu

ABSTRACT

As biomedical researchers are amassing a plethora of information in a variety of forms resulting from the advancements in biomedical research, there is a critical need for innovative information management and knowledge discovery tools to sift through these vast volumes of heterogeneous data and analysis tools. In this paper we present a general model for an information management system that is adaptable and scalable, followed by a detailed design and implementation of one component of the model. The prototype, called BioSifter, was applied to problems in the bioinformatics area. The results indicate that BioSifter is a powerful tool for biological researchers to automatically retrieve relevant text documents from biological literature based on their interest profile. The paper also presents experimental studies with real users to illustrate the efficacy of the approach.

Keywords

Bionformatics, Information filtering, Machine learning, Document representation, Document clustering

1. INTRODUCTION

The application of information science and technology to computational biology has focused mainly on three primary areas: the creation of databases which hold diverse information related to biology, the development of computational algorithms for biological data analysis, and software tools that allow researchers to access the data over the Internet and analyze them. With the advent of the World Wide Web and the development of new computer languages, such as Java, numerous software tools with excellent user interfaces are available that enable database searching and analysis over the Internet.

SAC '02, Madrid, Spain

The National Center for Biotechnology Information (NCBI) (http://www.ncbi.nlm.nih.gov/Genbank/) has created several tools that can be exploited by intelligent software systems (e.g., agents). For example, the Entrez system [1] provides an integrated view of DNA and protein sequence data, 3D structure data, and associated PubMed entries. A query can be formulated as a compact URL which when submitted to the Entrez server, can generate HTML output or route the results to a pre-designated e-mail address. Tools to extract scientific terms from domain specific articles are also available [2].

There are also systems that can provide customized information delivery services primarily based on information filtering techniques. For example, PubCrawler (http://www.pubcrawler.ie/) is a free "alerting" service that scans daily updates to the NCBI Medline (PubMed) and GenBank databases. PubCrawler can keep scientists informed of the current contents of Medline and GenBank, by listing new database entries that match their research interests. Syskill & Webert is a web based information agent capable of recommending relevant new content based on Baeysian analysis of past selection and retrieval of pages [3]. Another tool called CIAgent NewsFilter can connect to specific usenet newsgroups and download relevant news articles based on profiles created by the user [4].

The rapidly growing number and size of the databases makes it very difficult for researchers to obtain the customized information in an effective manner. Most databases are designed to support queries that require a direct human involvement. The user is expected to locate the correct resource, master the interaction conventions, and then formulate and execute the searches. Further, such mechanisms force the scientists to receive a view of the data as envisioned by their creators. Moreover, the information that the researcher may obtain from these databases is only a subset of the relevant data that is available if the search is directed to a specific data source. Therefore, many of the conclusions drawn today may use only a small percentage of the relevant information.

We propose a unified solution model for information management — from data retrieval to knowledge discovery, adaptable to user's interest. In this paper, we specifically address the first stage of this model, i.e., customized information retrieval using

information filtering (IF) approach as a possible solution. This stage of the model will enable researchers to stay abreast with the ever evolving biological information repositories without drowning in an ocean of irrelevant or unwanted data. The retrieved documents are rank ordered and presented based on a customized profile. The user profile is automatically learned on the basis of a simple user relevance feedback.

2. INFORMATION MANAGEMENT

Biological researchers repeatedly formulate queries, analyze search results, and refine search queries to encode their interests that are relatively stable over long-term. Any intelligent system that would attempt to automate this process must adhere to three key requirements; the system must be *active*, *personalized*, and *adaptive*. An active system means the ability to gather information with minimal user intervention. Personalization means it is cognizant of the interests and requirements of an individual researcher or a group of researchers sharing a common objective. This involves employing profiles that are suitable representations of user interests or requirements, and developing methods to construct such profiles. The adaptation feature requires that the system should have the ability to reconfigure in response to dynamic changes in the information domain (e.g., new resources or content changes) as well as user interests (a change in the research project and/or interests).

The overall problem of information management in this context can be modeled as a set of mapping functions as,

$$\underset{R^n}{D} \xrightarrow{f_D: D \to \Re} \underset{R^m}{I} \xrightarrow{f_I: I \to K_s} \underset{R^p}{K}$$

where, D is the original data; $\Re$ is a relevance value; f_D is a personalized profile; I is the information; K_s is the knowledge structure; f_I is the personalized profile for knowledge; K is knowledge; p, m, and n are the data, information and knowledge spaces respectively, where $p<<m<<n$. Here the data D is first mapped to information, I and information I gets mapped to knowledge, K. The mapping is based on some profile f at each level that allows both scalability and adaptability. The difficulty here is to obtain a metric for relevance at each stage of the mapping process.

In this paper, we specifically address the first stage of the information management process; i.e., mapping of biological text *data* into useful relevant *information*. We use information filtering techniques to aid in this mapping process. A prototype version of this phase, called BioSifter, has been developed and several experiments were conducted to validate its adaptivity and usability. BioSifter is an active, personalized and adaptive biological information discovery and delivery system. It actively searches, retrieves, correlates, and presents relevant information to the researcher in a seamless manner.

3. INFORMATION CUSTOMIZATION

Information customization in BioSifter is achieved through an information filtering process. The task of information filtering is to perform a mapping from a space of documents to a space of user relevance values, $\underset{R^n}{D} \xrightarrow{f_D: D \to \Re} \underset{R^m}{I}$. To reduce the overall complexity of filtering related to changing document contents and user-interests, we decompose this mapping into a multi-level process (see ref. [5] for more details). The intermediate levels of this process involve four basic tasks: thesaurus (ontology) discovery, information representation, document classification, and user profile management. In this process, we pose the overall filtering problem as learning a map f_D: $D \to \Re$ where D represents the document set, $\Re$ represents relevance assessment captured from users, and $f(d)$ corresponds to the relevance of a document d. Given that such a map is known for all points in D, a finite set of documents can always be rank-ordered and presented in a prioritized fashion to the user.

In this mapping process, f_D is not known *a priori* and has to be estimated on-line from the user feedback. Considering the high dimensionality of any reasonable representation of the documents, such a direct on-line learning of the map f_D is computationally intensive and requires a large amount of user feedback. To provide a practical, feasible solution, we decompose the problem into higher- and lower-levels. The higher-level decomposition represents a classification mapping f_1 from the document space to a finite number of classes (i.e., $f_1 : D \to \{C_1, \cdots, C_m\}$). This mapping is learned in an off-line setting without user involvement. It is based on a representative database of documents, either using *prior* information concerning the classes and examples, or by automatically discovering the abstractions using a clustering technique. Hence, this higher level partitions the document space into m equivalence classes over which the user relevance is estimated. The lower level subsequently estimates the mapping f_2 describing the user relevance for the different classes (i.e., $f_2 : \{C_1, \cdots, C_m\} \to \Re$). Since f_2, unlike f_D and f_1, deals with a finite input set of relatively few classes, the on-line learning of f_2 is not time-consuming and burdensome on the user. Thus, the map f_D is being learned as the composition of f_1 and f_2.

The general model of the first level mapping itself consists primarily of four modules: Thesaurus discovery, Document Representation, Document Classifier, and User Profile Manager. In the context of the multi-level decomposition of the map $f_D : D \to \Re$ (*i.e.*, $f_D = f_2 \circ f_1$) . The 1st and 2nd modules determine the input space for f_1, the 3rd module maps the resulting vector representation to the classification space (i.e., the output for f_1), and the 4th module implements the map f_2.

3.1 First Level Mapping

The term discovery module automatically builds a thesaurus (i.e., a set of key terms) from a collection of documents obtained through a key word search that is of specific interest to the researcher. In addition, the algorithm has parameters that can be adjusted by the researcher to control the granularity (specificity) of the terms and their associations (see [5] for details of this algorithm). The term discovery algorithm is motivated primarily by techniques developed in IR (especially, for automated thesaurus generation, see [6]). Research in IR actually produced a variety of token weighting and refinement techniques, ranging from those that are mainly statistically-oriented to those that rely heavily on analysis based on NLP. However, a general and surprising finding in IR is that the term frequency based approaches are at least as effective as the more sophisticated approaches [7,8].

The second component of the system converts documents into structures that can be efficiently parsed without the loss of vital content. At the core of this module is the thesaurus, an array T of atomic tokens (a single term) each identified by a unique numeric identifier culled from authoritative sources or automatically generated from a document collection. A thesaurus is an extremely valuable component in term-normalization tasks and for replacing an uncontrolled vocabulary set with a controlled set [10]. Beyond the use of the thesaurus, the *tf.idf* (the term frequency multiplied with inverse document frequency) algorithm [6] is applied as an additional measure for achieving more accurate and refined discrimination at the term representation level.

The classification module consists mainly of two processing stages: an unsupervised cluster learning stage and a vector classification stage. These are conducted in a batch mode to autonomously discover or learn classes. A heuristic unsupervised clustering algorithm, called the Maximin-Distance algorithm [9], is used to determine the centroids over the document vector space.

The function of the user profile learning module is to determine the user's preference for the different classes of information and prioritize the incoming documents based on their classes as well as the estimated user preferences for the classes. To accomplish this task, the learning process maintains and updates a simplified model of the user, based on the relevance feedback. The algorithm currently used to learn the user model is based on a reinforcement learning algorithm studied in the area of Learning Automata (Narendra and Thathachar, 1989) by the Artificial Intelligence and Mathematical Psychology communities. The details of this learning algorithm can be found in [11,12].

4. EXPERIMENTAL RESULTS

Several experiments were carried out to demonstrate the "data→information" mapping process (filtering) and to evaluate the performance of BioSifter. For brevity, in this paper we present the results of two such experiments; analysis of Genetic Polymorphism and Extracorporal shockwave lithotripsy (ESWL) problems.

In Genetic Polymorphism problem area, the researcher investigates genetic polymorphisms that influence the outcome of graft-versus-host disease (GVHD) - the major complication of bone marrow transplantation. More information on how to genetically screen for ideal donors would likely lead to a decrease in the incidence of life-threatening GVHD. In the second experiment, the researcher deals with Extracorporal shockwave lithotripsy (ESWL) treatment of kidney- and other stones by subjecting them to a focused shock wave originating outside the body. In both these problems the researchers are interested in obtaining relevant information specific to the areas of interest on a regular basis.

4. 1 Experimental Procedure

Users interact with BioSifter using a graphical user interface (GUI). This GUI allows the user to: (1) create a domain of interest, (2) start automatic creation of the thesaurus (profile), (3) provide data subscription sources as web links to databases, (4) set a time interval to view a list of recent documents, and (5) view the ordered documents and provide a feedback.

The first step toward using BioSifter is to adapt it for the specific domain of interest. This is an off-line process accomplished by automatically constructing a thesaurus for the intended problem. The thesauri for both ESWL and Polymorphism problems were constructed automatically using 2000 randomly collected documents from the PubMed database. The thesauri thus created for Polymorphism and ESWL problems are shown in Table 1 and 2 respectively.

Table 1: Automatically generated terms for Polymorphism problem

cad	vdr	evolution	drug
alpha	quot	polymorphism	loh
therapy	strains	genome	infection
plasma	risk	expression	syndrome
mapping	class	codon	genotype
carcinomas	receptor	identification	genes
repeat	liver	linkage	hla
lesions	renal	platelet	mutation
marker	markers	immune	graft
host	recipient	reject	rejection
disease	gvhd	graft-versus-host-disease	
transplant	organ	organs	physiology
mouse	mice	bone	marrow
transplantation	mhc	mthfr	

The second step involves the cluster formation and obtaining the cluster centroids. This is also a one time, off-line process where the document set collected in the previous step from PubMed database is used to create clusters. Figure 1 shows the initial clusters obtained for the Polymorphism problem (case study 1). In Figure 1, each horizontal bar represents a cluster and the terms in the cluster centroid are shown in the right side of the bar. The length of the cluster bar represents the relevance of the documents in this cluster to the user. Initially, the relevance is set equal hence the lengths are same.

Table 2. The list of terms in the thesaurus for the ESWL problem generated automatically

gallbladder	children	endoscopic
nephrolithiasis	stent	pain
recurrence	ureteroscopy	pneumatic
urolithiasis	staghorn	pole
energy	cavitation	choledocholithiasis
bladder	ct	laser
analgesia	holmium,yag,ho	electrohydraulic,ehl
pcnl,percutaneous	gallstone,gallston	bile,biliary
ureteral,ureter,ureteric	caculi,caliceal,calculus	
kidneys,kidney,renal	pancreatic,pancreas	frequency,repetition
hemolysis,haemolysis,lysis,cell lysis		lesion,hemorrage
blood pressure,hypertension		
renal blood flow,renal plasma		flow,rbf,rpfp
ancreatitispyelonephritis,pyelonephritic		

Once the cluster centroids are obtained through off-line training, the final step is to obtain a user interest profile. This is created as a vector of normalized real numbers between 0 and 1 whose dimension is equal to the number of clusters. A particular element in this vector represents the user's interest in the corresponding cluster, with '0' indicating no interest and '1' indicating the highest interest. The elements of this profile vector are continuously

updated during running of BioSifter based on user provided relevance feedback using the machine learning algorithm. Figure 2 and Figure 3 shows the state of the clusters for both problems

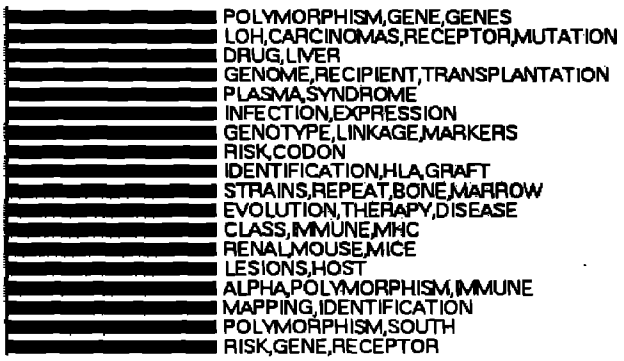


Figure 1. The initial cluster space generated for the Polymorphism problem

after approximately ten session of learning (at each learning session, 15 documents were presented to the user and the user provides a feedback). As shown in Figure 2 for the Polymorphism problem, the user profile learning algorithm discovered that cluster #4 (centroid: GENOME, RECIPIENT, TRANSPLATION) and cluster 9 (centroid: IDENTIFICATION, HLA, GRAFT) as the two most relevant clusters, while clusters 2, 3, and as the two least relevant ones. Similarly, for the ESWL problem as shown in Figure 3, the most relevant one is cluster #8 (centroid: CAVITATION, HEMOLYSIS) whereas, cluster #18 is the least relevant one.

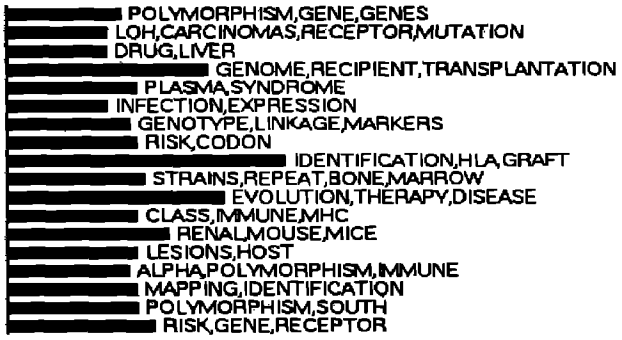


Figure 2. Cluster space after several learning sessions for the Polymorphism problem

4.2 Performance Results

For the two experiments reported here, BioSifter used a total of 500 documents obtained from PubMed database. BioSifter was tested with 10-15 sessions for each experiment. Each session consisted of filtering 15 documents using the thesaurus created as described earlier. Also, for each experiment the filtering performance was evaluated using two related criteria, normalized recall and normalized precision, described by Salton [6].

In the following equations, N represents the total documents in the collection, and REL represents the total number of relevant documents. Each of these criteria takes values between 0 and 1,

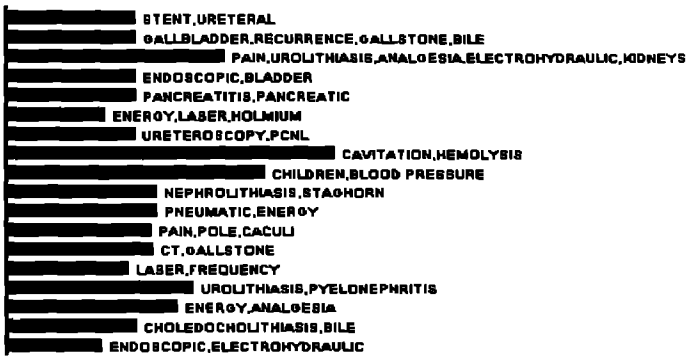
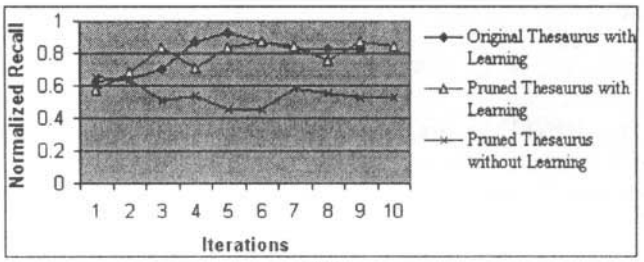


Figure 3. Cluster space after several learning sessions for the ESWL problem

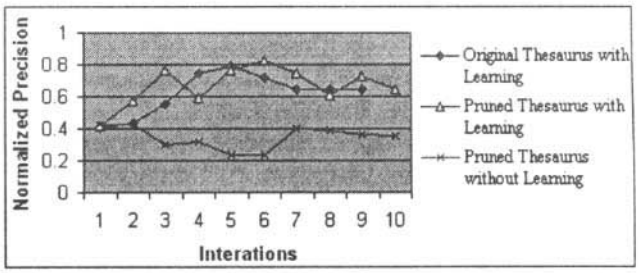
with 1 representing the best performance and 0 representing the worst performance. Qualitatively speaking, *recall* refers to the percentage of relevant documents that are retrieved, while *precision* refers to the percentage of retrieved documents that are relevant.

$$\text{Recall}_{norm} = 1 - ((\sum_{i=1}^{REL} RANK_i - \sum_{i=1}^{REL} i)/(REL(N - REL)))$$

$$\text{Precision}_{norm} = 1 - ((\sum_{i=1}^{REL} \log Rank_i - \sum_{i=1}^{REL} \log i)/\log i(N!/(N - REL)!REL!)$$



(a)

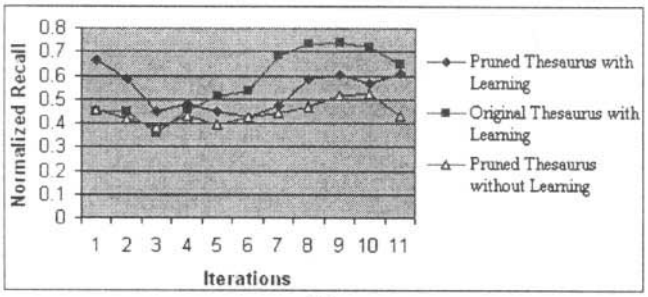


(b)

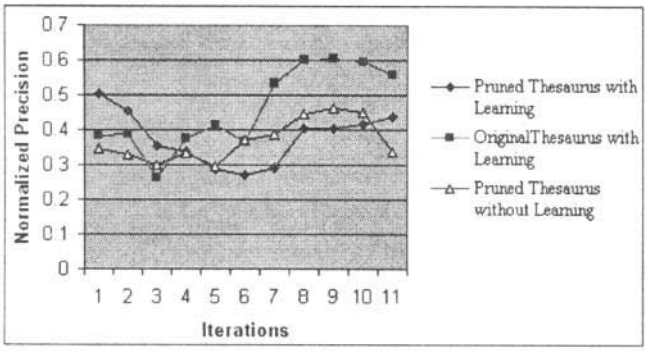
Figure 4. (a) Normalized recall and (b) normalized precision for the Polymorphism problem

The graphs in Figure 4 shows the normalized recall and precision of the document as it was presented to the user for the Polymorphism problem. The experiments were conducted using the original thesaurus that was created automatically as well as using a "pruned" thesaurus. Editing and eliminating some of redundant or insignificant terms that was present in the original thesaurus resulted in the pruned thesaurus. These figures also

show the graphs of recall and precision if no "learning" was included (i.e., without filtering).



(a)



(b)

Figure 5. (a) Normalized recall and (b) normalized precision for the ESWL problem

Similar experiments were conducted for the ESWL problem. Graphs shown in Figure 5 depict the recall and precision respectively. These experiments were also conducted with and without pruning the thesaurus and with and without learning.

Both experimental results indicate the significance of filtering as a mechanism for data → information mapping. Given adequate time for learning, BioSifter can filter out such documents and provide the user only with relevant documents. The key point is that improved performance was achieved when the system learned a user profile and applied to filter the incoming data.

5. DISCUSSION

The current practices of manually specifying search patterns, coordinating search activities from multiple sources, and analyzing the retrieved data for information value, are tedious, and time-consuming even for a highly motivated individual.

In this paper, we specifically addressed the first stage of IM, i.e., mapping data to relevant information. BioSifter provides automated and efficient methods as well as a working system to provide biological researchers with active, personalized and integrated information delivery with minimal user interaction. The concept of using thesaurus-based profiles allows the method to be both adaptable and scalable. Another important feature of the proposed approach is that the first stage of information management process, i.e., *data* to *information*, significantly reduces the information space. The filtered data obtained through BioSifter is relevant as well as much smaller in dimension compared to all the retrieved data. This would in turn significantly reduce the complexity associated with the next level transformation, *information* to *knowledge*.

Our future plans are to extend the first phase of the information management process to include sequence and structural biological data, as well as methods for integrating these multi-format data. Work is also already under way for the second phase of the problem of mapping information to knowledge.

6. ACKNOWLEDGEMENTS

This project is supported in part by a grant from the Eli Lilly & Co., and by National Science Foundation ITR-Grant #NSF-IIS/ITR 0081944. The authors would like to thank Matthew Stephens, Jeremy Doherty, Sheri Groenroberts, John Fieber and Ruth Allen for their assistance with the experiments.

7. ADDITIONAL AUTHORS

Rajeev Raje, IUPUI Indianapolis, Indiana; Mathias N'Cho and Santosh Mishra, Eli Lilly & Co., Indianapolis, Indiana.

8. REFERENCES

[1] Butler, A. Sequence Analysis using GCG, in Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, Baxevanis & Ouellette (eds.), John Wiley, NY, 1998.

[2] Andrade, M.A. and Valencia, A. Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. *Bioinformatics*. 14, 600-607, 1998.

[3] Pazzani, M., Muramatsu J., and Billsus, D. Syskill and Webert: Identifying interesting web sites. Proceedings of the National Conference on Artificial Intelligence, Portland, OR, 1996.

[4] Bigus, P., Bigus, J., and Bigus, J. Constructing Intelligent Agents Using Java: Professional Developer's Guide, 2001.

[5] Mostafa, J., Mukhopadhyay, S., Lam, W., and Palakal, M. A Multi-level Approach to Intelligent Information Filtering: Model, System, and Evaluation. ACM Transactions on Information Systems, **15(4)**, 368-399, 1997.

[6] Salton, G. and McGill, M. J. Introduction to Modern Information Retrieval. McGraw-Hill, NY, 1983.

[7] Lewis, D. Text representation for intelligent text retrieval: A classification-oriented view. In P.S. Jacobs (Ed.), Text-based intelligent systems: Current research and practice in information extraction and retrieval, pp. 179-197, Hillsdale, NJ: Erlbaum, 1992.

[8] Levis, D. D. Representation and learning in information retrieval. Dissertation, Dept. of Computer and Information Science, Univ. of Massachusetts, Amherst, MA, 1992.

[9] Tou, J. T. and Gonzalez, R. C. Pattern Recognition Principles. Addison-Wesley, Reading, MA, 1974.

[10] Salton, G. Automatic Text Processing. Addison-Wesley, Reading, MA, 1989.

[11] Narendra, K. S. and Thathachar, M. A. L. Learning Automata -- an Introduction. Prentice Hall, NJ, 1989.

[12] Thathachar, M. A. L., and Sastry, P. S. A New Approach to the Design of Reinforcement Schemes for Learning Automata. *IEEE Transactions on Systems, Man, and Cybernetics*, 15, 168-175, 1985.