A New Multitask Learning Method for Multiorganism Gene Network Estimation

Marcel Nassar, Rami Abdallah, Hady Ali Zeineddine, Elias Yaacoub, and Zaher Dawy

Electrical and Computer Engineering Department, American University of Beirut, Riad El-Solh
P.O. Box 11-0236, Beirut 1107 2020, Lebanon
Email: {mhn04, raa57, hma41, eey00, zaher.dawy}@aub.edu.lb

***Abstract*— A new method for multitask learning in a Bayesian network context is presented for multiorganism gene network estimation. When the input datasets are sparse, as is the case in microarray gene expression data, it becomes difficult to separate random correlations from actual edges in the true underlying Bayesian network. Multitask learning takes advantage of the similarity between related tasks, in order to construct a more accurate model of the underlying relationships represented by the Bayesian networks. The proposed method is tested on synthetic data to illustrate its validity. Then it is iteratively applied on real gene expression data to learn the genetic regulatory networks of two organisms with homologous genes (human and yeast).**

***Index Terms*— Bayesian networks, evolutionary information, genetic regulatory networks, multitask learning**

I. Introduction

The modeling of gene interactions as networks is becoming increasingly widespread for gaining insight into various cellular properties, such as cellular state dynamics and transcriptional regulation [1]. The resulting networks provide a high level description of the gene expression system by predicting how genes interact with each other through regulatory actions and translational feedback.

Bayesian genetic regulatory networks are attractive in analyzing gene expression data for their ability to describe complex stochastic processes [2]. A Bayesian network can be regarded as a task; a more general term used in the machine learning literature. Multitask learning is an approach that learns a problem together with other related problems, using a shared representation. This often leads to a better model for the main task, because it allows the learner to use the common information among the tasks [3]. Multitask learning is well-suited in domains where the cost of collecting more data is prohibitive or there is a limited training data associated with each task as in the case of DNA microarrays [4]. DNA microarrays measure gene expression levels, and provide data samples that are sparse compared to the number of interacting genes that need to be modeled [5].

In multitask learning, the tasks are usually not identical, and hence cannot be simply treated as a single task by merging the available datasets. Instead the tasks are assumed to be dependent; consequently, treating them as independent tasks, as in the single task learning, might cause loss of useful information. The motivation for assuming dependencies between tasks in the gene network context arises from the conservation of translational machinery across species at the expression level [6], [7]. Learning a task requires inductive biases towards some elements of the search space [3]. In multitask learning, the extra tasks serve as additional inductive bias for the learning of the main task on top of the bias provided by the model distribution and available data [3]. This paper proposes a new multitask learning method targeted for applications with sparse datasets as in the case of constructing gene networks. Results for synthetic data and real gene expression profiles are presented to validate the proposed method.

This paper is organized as follows. Background information on Bayesian networks and orthologous genes across organisms is provided in Section II. The proposed method is explained in Section III. Results are presented and discussed in Section IV. Finally, conclusions are drawn in Section V.

II. Background

A. Bayesian Networks

A Bayesian network is a graphical model for representing conditional independencies between a set of random variables. This representation consists of two components G and Θ. G is a directed acyclic graph (DAG) whose n nodes (genes) correspond to the random variables $X_1, X_2, ..., X_n$, while Θ describes the conditional probability distribution of each node X_i, given its parents[1] $P_a(X_i)$, i.e. $\{P(X_i|P_a(X_i)), X_i \in G\}$. A basic assumption in Bayesian networks is the applicability of the Markov assumption: each variable X_i is independent of its non-descendants, given its parents in G. This property allows the joint probability distribution $P(X_1, X_2, ..., X_n)$ to be decomposed into the following product form [2]:

$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i|P_a(X_i)) \qquad (1)$$

Modeling a gene network as a Bayesian network is motivated by the fact that the causal relations between genes can be described qualitatively through the presence of edges, and quantitatively through the conditional probability distributions. Biological factors, measurement inaccuracies, and data sparsity prohibit the prediction of deterministic relations between genes, and act as a noise source that results in the probabilistic nature of relations. Bayesian gene networks can also be extended to incorporate temporal order to account for a direct causal effect of one gene on another through different time stages.

[1] All nodes that have an edge directed towards X_i

978-1-4244-2571-6/08/$25.00 ©2008 IEEE

B. Genes across Organisms

Since microarray gene expression data do not contain sufficient information for estimating accurate complex gene networks, other biological information is being considered to improve the estimation process. Recent studies have revealed highly conserved proteins that exhibit similar expression patterns in different organisms, and play a role in gene regulation. Therefore, this evolutionary information can be used to refine regulatory relationships among genes, which are estimated from gene expression data and bias the learning process. In genomics, sequence homology implies that the given sequences share a common ancestor. Orthology is a type of sequence homology that results from a speciation event; i.e. a parent species diverging into two new species. Orthologous genes usually preserve protein expression and regulation functions and can be used to quantify the evolutionary information between two organisms [7].

The authors in [8] utilized evolutionary information between two organisms to estimate the individual gene networks. Assuming two organisms A and B with respective gene expression data D_A and D_B, the networks of the two organisms G_A and G_B are built simultaneously by a hill-climbing algorithm that maximizes the posterior probability function $P(G_A, G_B|D_A, D_B, H_{AB})$ where H_{AB} models the evolutionary information between A and B. As part of their algorithm, the authors calculate $P(H_{AB}|G_A, G_B)$ based on gene expression data of the orthologous gene pairs between A and B in addition to two free parameters ζ_N and ζ_P that need to be set. However, they do not give a systematic approach for choosing ζ_N and ζ_P. Instead they choose them empirically by comparing the obtained networks to established biological pathways. This is a main shortcoming because such pathways are normally not available with the exception of few organisms.

In this work, we follow an approach similar to [8] and focus on the structure learning problem of the Bayesian network. This improves parameter learning indirectly since parameter learning depends on structure learning [2]. We propose a new score function that captures the evolutionary information between A and B by a parameter β (see Section III) without requiring any additional free parameters and that can be localized[2] easily for reduced computational complexity (K2 algorithm vs. hill-climbing algorithm).

III. The Proposed Structure Learning Algorithm

In order to exploit multitask learning, a quantitative link between the Bayesian networks of the different tasks (organisms) must be determined. To model this link, we propose a similarity parameter β. This parameter represents the similarity of the underlying true Bayesian networks; i.e. the Bayesian networks that would be learned if we had complete information. Let $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ be two DAGs of B_1 and B_2, the underlying true Bayesian networks of organisms 1 and 2, respectively, where V is the set of vertices (genes) and E_i is the set of edges (regulatory relations). We define the similarity parameter β as

$$\beta \triangleq P(e \in E2 \mid e \in E1) = P(e \notin E2 \mid e \notin E1) \quad (2)$$

where we assume that the probability of the presence of an edge e in both networks and its absence is the same. Since $\in$ is a binary relation, we can deduce that $P(e \notin E2 \mid e \in E1) = P(e \in E2 \mid e \notin E1) = 1 - \beta$. Based on the given definition, it is difficult to obtain a numerical value for the parameter β based on general evolutionary biological information. To alleviate this problem, we make use of the known common orthologous genes between the two organisms in order to estimate the value of β. The following estimation procedure is proposed:

Step 1 : Isolate the orthologous genes between the two organisms.

Step 2 : Construct the Bayesian network for the orthologous genes for each organism separately using single network learning algorithms such as the K2 algorithm discussed in Section III-A.

Step 3 : Obtain an estimate for β using the following expression:

$$\beta = \frac{2N_c}{|E_1| + |E_2|} \quad (3)$$

where N_c is the number of common edges between the two DAGs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$, and $|E_i|$ is the total number of edges in G_i.

A. Basic Algorithm Structure

To find the best matching graph using the single network approach (i.e. using the data of just one organism), the scoring functions must be proportional to $P(G|D)$, which can be expressed as

$$P(G|D) = \frac{P(D|G)P(G)}{P(D)} \propto P(D|G)P(G) \quad (4)$$

where G is the DAG under evaluation, D is the available dataset, and $P(G)$ is assumed to be uniformly distributed if no prior knowledge is available. Maximizing $P(G|D)$ is equivalent to maximizing $P(D|G)$ that in turn can be decomposed into maximizing local scores by the Markov property. This allows the use of local search techniques which for each node X_i search for the parent set $P_a(X_i)$ that maximizes the local scoring function.

The scoring function that is derived in Section III-B, can be used with various existing structure learning algorithms such as the K2, hill-climbing, and MCMC algorithms. In this work, we use the basic structure of the K2 algorithm, a widely used greedy algorithm, because it provides a better comparison between different score functions as it does not have a random component. Moreover, the K2 algorithm benefits from the local decomposability of the Bayesian score (BDe) across the parents of every node to locally search for the best DAG, thereby minimizing the search space of the problem. This in turn reduces the required computational complexity. For each node, the K2 algorithm adds a parent to a node as long as this parent causes a positive change in score ($\triangle(e)$).

The following is a general description of the K2 algorithm:

[2]A local score is a function of the node X_i and its parents $P_a(X_i)$ only

- for $i = 1$ to n do
  - while number of parents of X_i is less than u (a specified upper bound) and there exist $\triangle(e) > 0$, add an edge from X_j to X_i, such that the edge addition e yields the maximum $\triangle(e)$.
  - end while
- end for

The K2 algorithm tries to find the most probable structure given the data. In other words, it tries to find the structure that maximizes $P(G|D)$, through maximizing $P(D|G)$ since all networks are assumed to have equal priors. This maximization reduces to the following form:

$$\max_G P(D|G) = \prod_{i=1}^{n} \max_{\pi_i} f(i, \pi_i) \tag{5}$$

where n is the number of nodes of G, i is a node index corresponding to node x_i of G, and π_i is the set of parents of x_i. $f(i, \pi_i)$ evaluates the probability that the set π_i is the real set of parents of x_i, denoted by P_{a_i}, given the data D. In other words, $f(i, \pi_i) \propto P(P_{a_i} = \pi_i|D)$. For detailed information about the K2 algorithm refer to [5].

B. Score Function Derivation and Algorithm Explanation

In this section, we explain the proposed new algorithm that makes use of evolutionary relations between two organisms to estimate their genetic regulatory networks. The inputs to the algorithm are: data samples D of given organism, input DAG G_{in} of the other organism, and similarity parameter β. The output of the algorithm is a learned improved DAG structure G_{out} of given algorithm. This procedure can be iteratively repeated between both organisms using the output DAG of one organism as an input to the other organism.

Generalizing the Bayesian score approach (BDe), the objective is to find a structure that maximizes $P(G|D, G_{\text{in}})$ instead of $P(G|D)$:

$$P(G|D, G_{in}) = \frac{P(G, D|G_{\text{in}})}{P(D|G_{\text{in}})} \tag{6}$$

Using the fact that $P(D|G_{\text{in}})$ is independent of G, we obtain

$$P(G, D|G_{\text{in}}) = P(D|G, G_{\text{in}})P(G|G_{\text{in}}) \tag{7}$$

In addition, D is independent of G_{in} given G, since G_{in} provides only structural information, and D depends only on the structural information of its network. Therefore,

$$P(G, D|G_{\text{in}}) = P(D|G, G_{\text{in}})P(G|G_{\text{in}}) = P(D|G)P(G|G_{\text{in}}) \tag{8}$$

$P(D|G)$ is the BDe score used in the K2 algorithm, whereas the new factor $P(G|G_{\text{in}})$ will perform the biasing towards the input network. The above scores are decomposed to local scores, involving only nodes and their parents, as is done in the K2 algorithm. The local score corresponding to $P(D|G)$ is represented by the $f(i, \pi_i)$ terms in the K2 algorithm. The local score of $P(G|G_{\text{in}})$ is $P(P_{a_i}^{(1)} = \pi|P_{a_i}^{(2)} = \alpha)$ that is the probability that the set of parents $P_{a_i}^{(1)}$ of x_i in G is π given that the set of parents $P_{a_i}^{(2)}$ of x_i in G_{in} is α. To calculate $P(P_{a_i}^{(1)} = \pi|P_{a_i}^{2} = \alpha)$, we resort to the definition of the similarity parameter β. The set of parents of node x_i can be viewed as the set of edges incident to the node x_i.

We define P_{s_i} to be the set of all possible parents of x_i, and we denote by E_i the set of all possible edges that are incident to the node x_i. Similarly, we extend this definition to any set of parents of x_i. That is, given π to be a set of parents of node x_i, we define $E_{i\pi}$ to be the set of all incident edges corresponding to the set π. In other words, if $x_j \in \pi$ then the edge $(x_j, x_i) \in E_{i\pi}$ and vice versa.

Clearly, $P_{a_i}^{(1)} \subseteq P_{s_i}$ and $P_{a_i}^{(2)} \subseteq P_{s_i}$. To decompose $P(P_{a_i}^{(1)} = \pi|P_{a_i}^{(2)} = \alpha)$, the probability that the node x_i has the set π as parents given that the prior (input) structure has α as a set of parents, we rely on the corresponding edge probabilities. In other words, we check every possible edge e in E_i, and determine its absence or presence in the sets $E_{i\pi}$ and $E_{i\alpha}$. The following decomposition illustrates this idea:

$$\begin{aligned} P(P_{a_i}^{(1)} = \pi|P_{a_i}^{(2)} = \alpha) = & \\ & \prod_{e \notin E_{i\pi}, E_{i\alpha}} P(e \notin E_{i\pi}|e \notin E_{i\alpha}) \\ & \cdot \prod_{e \in E_{i\pi}, E_{i\alpha}} P(e \in E_{i\pi}|e \in E_{i\alpha}) \\ & \cdot \prod_{e \notin E_{i\pi}, e \in E_{i\alpha}} P(e \notin E_{i\pi}|e \in E_{i\alpha}) \\ & \cdot \prod_{e \in E_{i\pi}, e \notin E_{i\alpha}} P(e \in E_{i\pi}|e \notin E_{i\alpha}) \end{aligned} \tag{9}$$

where $e \in E_i$. Based on the definition of the similarity parameter β, (9) reduces to

$$\begin{aligned} P(P_{a_i}^{(1)} = \pi|P_{a_i}^{(2)} = \alpha) = & \prod_{e \notin E_{i\pi}, E_{i\alpha}} \beta \cdot \prod_{e \in E_{i\pi}, E_{i\alpha}} \beta \\ & \cdot \prod_{e \notin E_{i\pi}, e \in E_{i\alpha}} (1-\beta) \cdot \prod_{e \in E_{i\pi}, e \notin E_{i\alpha}} (1-\beta) \end{aligned} \tag{10}$$

To solve (10), let $H = E_{i\pi} \oplus E_{i\alpha}$ be the set of all elements that are not common in the two sets. Defining $m = |H|$, the number of elements in H, and $N = |Ps_i|$, (10) reduces to

$$P(P_{a_i}^{(1)} = \pi|P_{a_i}^{(2)} = \alpha) = (1-\beta)^m \beta^{N-m} = \left(\frac{1-\beta}{\beta}\right)^m \beta^N \tag{11}$$

Since N is common to all the structures being evaluated, the final score expression can be expressed as

$$P(P_{a_i}^{(1)} = \pi, D|P_{a_i}^{(2)} = \alpha) \propto \left(\frac{1-\beta}{\beta}\right)^m \tag{12}$$

As a result, the following steps should be used to calculate a new score function to replace $f(i, \pi_i)$ in the K2 algorithm.

Step 1 : Find m, the number of edges that are different between the parents under consideration and the parents of the input structure.

Step 2 : Multiply $f(i, \pi_i)$ by $(1 - \beta/\beta)^m$.

IV. Results and Discussion

Before applying the proposed algorithm to real gene expression data, it is first tested on synthetic data provided by the

ALARM network. This synthetic network is considered one of the standard benchmark networks for comparison between Bayesian learning algorithms.

A. Synthetic Data: ALARM Network

The proposed method is applied first on two similar networks derived from the ALARM network. The network has 37 random variables (nodes) and 46 edges, and each random variable has up to four possible values and a maximum of five parents.

In order to generate similar networks that resemble biological networks, we used a similar approach as in [9]: Starting from a base network, we generate two similar networks according to an input parameter $P_{\rm del}$. Similar networks are generated by passing through all edges of the base network and deleting any given edge with probability $P_{\rm del}$. This procedure can be repeated to generate different networks that are similar to the order of $P_{\rm del}$. Any two generated networks contain common edges in addition to edges that are present in one and not in the other. The proposed multitask learning algorithm is compared to the single network K2 algorithm in terms of the following metrics:

1) False positives: the number of edges absent in the real underlying structure but present in the learned structure
2) False negatives: the number of edges present in the real underlying structure but absent in the learned structure
3) Total error: the difference between the real and the learned structures, i.e. the sum of false positives and false negatives

Fig. 1 presents the results based on two similar networks of equal size datasets. The y-axis represents the average number of occurrences over multiple runs and the x-axis represents the dataset size. The proposed method, denoted by "New", clearly outperforms the single network K2 algorithm in terms of false positives. This illustrates the fact that the new method reduces the effect of noisy observations, i.e. the statistical deviations in the data. However, the performance in terms of false negatives is comparable. This is due to the fact that real edges that are weakly supported by the data might not be identified by the proposed method. Nevertheless, in terms of total error, the advantage of the proposed method is evident. Finally, it can be seen that as the dataset size increases, the error rates decrease due to a reduction in the estimation noise.

B. Real Gene Expression Data

The proposed method is used to simultaneously estimate two gene networks of two distinct organisms, with a Bayesian network model utilizing the evolutionary information so that gene expression data of one organism improves the estimation of the gene network of the other organism. We demonstrate the effectiveness of the proposed algorithm through the analysis on Saccharomyces cerevisiae and Homo sapien cell cycle gene expression data. The proposed algorithm is successful in estimating gene networks that capture many known relationships as well as some unknown relationships which are likely to be novel.

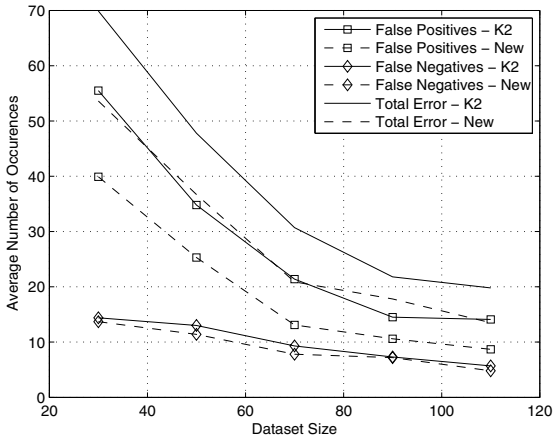

Fig. 1. Error rate of the proposed algorithm with ALARM network synthetic data.

There are only a few datasets where both detailed knowledge of the regulatory pathways and the same type of expression data are available for two distinct organisms [8]. The proposed algorithm is implemented on a set of ten orthologous genes between yeast (S. cerevisiae) and human (H. sapiens). The list of the chosen genes is shown in Table I. The data for the yeast genes is available at http://genome-www.stanford.edu/cellcycle/ [10], and the data for the human genes at http://genome-www.stanford.edu/Human-CellCycle/Hela/ [11].

Yeast genes	Human genes
CDC28	CDK7
CDC6	CDC6
CDC7	CDC7
ORC1	ORC1L
MCM2	MCM2
MCM3	MCM3
CDC46	MCM5
MCM6	MCM6
CDC47	MCM7
CDC45	CDC45L

TABLE I
ORTHOLOGOUS GENES BETWEEN HUMAN AND YEAST.

The obtained results are compared to known relations in the published KEGG pathways [12]. To start the analysis, we learn the yeast gene network using the classical K2 algorithm based only on the yeast dataset. Then, the proposed algorithm is applied to learn the human gene network by using the human data from [11] in addition to the obtained yeast gene network (learned with the K2 algorithm). The results are shown in Fig. 2. The estimated similarity factor is given by $\beta = 0.8$. Having obtained the human gene network, we use it to enhance the gene network of the yeast genes; i.e. the yeast data from [10] and the learned human gene network are given as inputs to the proposed algorithm to enhance the yeast gene network. The results are also shown in Fig. 2.

It can be seen that in the first step, known relations that were not detected in the yeast network by the K2 algorithm

were detected by the proposed algorithm in the human network (e.g. relation between CDC6 and the ORC complex). In addition, the human gene network was successfully used to enhance the yeast gene network. In fact, the edge between MCM6 and CDC45, and the one between CDC47 and CDC45 were not detected by the classical K2 algorithm. However, when the learned human network was used with our proposed algorithm, these two edges were detected.

Finally, the results are compared to other results discussed in the literature, namely [8] and [12]. Many of the relations obtained using the proposed algorithm are consistent with the KEGG pathways; other obtained relations are not listed in the KEGG pathways but were also estimated in [8] (e.g. relation between CDC28 and MCM2 in yeast), which suggests that experimental effort should be done to verify these relations biologically, since they were detected by different approaches. Comparing the obtained results to [8], it can be seen that many relations listed in [8] and not yet biologically verified were not detected by the proposed algorithm (e.g. relation between CDC6 and CDC45L in human), whereas some new relations estimated by the proposed algorithm were not estimated in [8] (e.g. relation between CDK7 and the MCM complex in human).

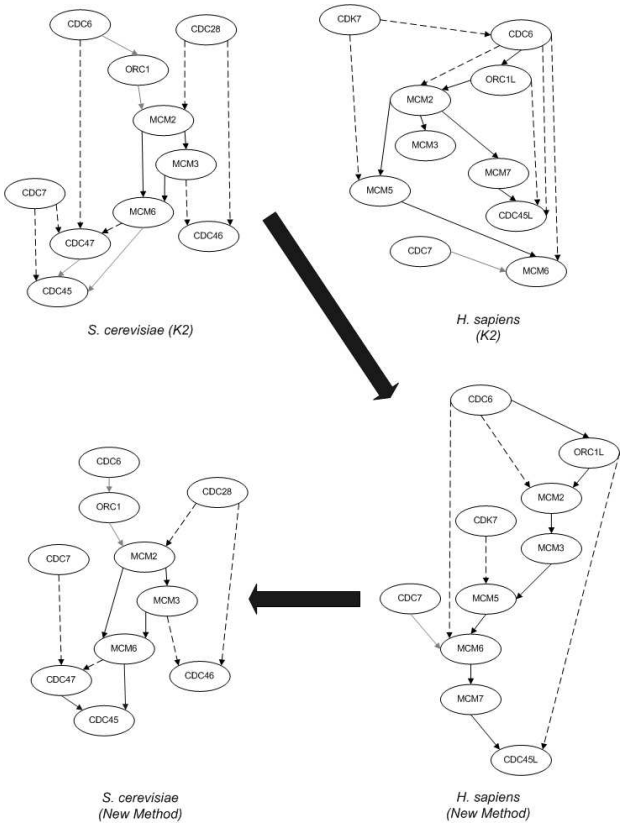

Fig. 2. Obtained gene networks: solid black edges correspond to relations consistent with the KEGG pathways, dashed edges correspond to newly estimated relations, and gray edges correspond to listed relations in the KEGG pathways that were not estimated.

V. Conclusions

We propose a new algorithm for multitask learning in the context of Bayesian networks. The algorithm was successfully tested and verified on synthetic data using ALARM benchmark network. The main objective of the proposed algorithm is to utilize evolutionary preserved gene interactions in order to learn or refine the construction of genetic regulatory network models across multiple organisms. The proposed algorithm was also applied to S. cerevisiae and H. sapiens real cell cycle gene expression data. Results demonstrate that the proposed algorithm was able to improve the gene network estimation accuracy compared to published KEGG pathways. In addition, new relations were also estimated which would motivate verification via lab based biological experiments.

Acknowledgment

The authors wish to to express their gratitude for Prof. Kevin Murphy for making his BNT MATLAB toolbox available freely on his website. This toolbox provided the necessary frame to develop and test the proposed algorithm.

References

[1] M. Brun, S. Kim, W. Choi, and E. R. Dougherty, "Comparison of Gene Regulatory Networks via Steady-State Trajectories," *EURASIP Journal on Bioinformatics and Systems Biology*, January 2007.

[2] E. Neapolitan, *Learning Bayesian Networks*. Prentice Hall, 2004.

[3] R. Caruana, "Multitask Learning," *Machine Learning, Vol. 28, No. 1, pp. 41-75*, 1997.

[4] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram, "Multi-Task Learning for Classification with Dirichlet Process Priors," *Journal of Machine Learning Research, Vol. 8, pp. 35-63*, May 2007.

[5] N. Friedman, M. Linial, I. Nachman, and D. Peer, "Using Bayesian Networks to Analyze Expression Data," *Journal of Computational Biology, Vol. 7, No. 3/4, pp. 601-620*, 2000.

[6] J. L. Jimenez, M. P. Mitchell, and J. G Sgouros, "Microarray Analysis of Orthologous Genes: Conservation of the Translational Machinery across Species at the Sequence and Expression Level," *Genome Biology, Vol. 4, No. 1*, 2003.

[7] S. Bergmann, J. Ihmels, and N. Barkai, "Similarities and Differences in Genome-Wide Expression Data of Six Organisms," *PLOS Biology, Vol. 2, No. 1*, January 2004.

[8] Y. Tamada, H. Bannai, S. Imoto, T. Katayama, M. Kanehisa, and S. Miyano, "Utilizing Evolutionary Information and Gene Expression Data for Estimating Gene Networks with Bayesian Network Models," *Journal of Bioinformatics and Computational Biology, Vol. 3, No. 6, pp. 1295-1313*, December 2005.

[9] A. Niculescu-Mizil and R. Caruana, "Learning the Structure of Related Tasks," *Proceedings of Inductive Transfer NIPS Workshop*, 2005.

[10] P. T. Spellman et al., "Comprehensive Identification of Cell Cycle-Regulated Genes of the Yeast Saccharomyces Cerevisiae by Microarray Hybridization," *Molecular Biology of the Cell, Vol. 9, pp. 3273-3297*, December 1998.

[11] M. L. Whitfield et al., "Identification of Genes Periodically Expressed in the Human Cell Cycle and their Expression in Tumors," *Molecular Biology of the Cell, Vol. 13, pp. 1977-1200*, June 2002.

[12] M. Kanehisa et al., "From Genomics to Chemical Genomics: New Developments in KEGG," *Nucleic Acids Research, Vol. 34, pp. 354-357*, January 2006.