PORT SAID ENGINEERING RESEARCH JOURNAL

Faculty of Engineering - Port Said University
Volume 27 No. 1 pp: 87-97

Gat2Get: A Novel Approach to Infer Gene Regulatory Network from Gene Activity using Dynamic Bayesian Network learning.

Safa'a S. Saleh [1*], Iman S. Alansari[2], Waleed Ead[3] , Hatem A. Khater[4]

[1] *Information Systems Department, Alexandria Academy for management and Accounting, Alex, Egypt, email: Safaa34@gmail.com*

[2] *Computer Science Department, college of Computer Science and Engineering, Taibah University, City, SA, email: iansari@taibahu.edu.sa,*

[3] IS *Department, Faculty of Computer and AI, Beni Suef University, Beni Suef 62511,* Egypt, *email: waleedead@bsu.edu.eg,*

[4] *Electrical Department, Faculty of Engineering, Horus University, New Damietta 34518,* Egypt, *email: hatem.a.khater@gmail.com*

**Corresponding author Safaa34@gmail.com,* *DOI: 10.21608/PSERJ.2023.186705.1213*

Received 11-1-2023,
Revised 2-3-2023,
Accepted 27-3-2023

ABSTRACT

Discovering Gene Regulatory Network (GRN) gives some idea about gene pathways and helps many potential applications in medicine. The essential source of data for this task is the gene expression data. High complexity and poor quality of gene expression data acquired by high throughput methods like microarray provide many difficulties in the context of the current issue. A promising method for evaluating gene expression noisy data to characterize processes made up of locally interacting components is Bayesian Network. In fact, because of the intricacy of the inputs and results of the cellular mechanism, inferring GRN from expression data presents numerous difficulties. This work proposes a new approach for inferring GRNs from time series gene expression data. The present work extends the existing Bayesian Network methods to include the regulation properties of genes to improve the process of capturing natural classes during inferring the relations between genes. The proposed approach is evaluated in comparing to the corresponding techniques of the related works, and the results show the ability of the present approach is efficient to some level to deal with such high dimensional data even without dimension reduction, but in the presence of regulatory information.

Keywords: Gene Regulatory Network, Gene expression, Bayesian network, Gene Regulation Ontology

1 INTRODUCTION

Gene Regulatory Network (GRN) is known as a group of genes that indirectly regulate the activity rates of each other through their protein or RNA products [1]. Discovering and understanding this regulation processes that underlie the vital operations of human disease is one of the major goals in bioinformatics. This understanding gives some idea about gene pathways and helps many potential applications in medicine. Drug discovery and toxicology analysis are the most important examples of these applications in addition to complex genetic diseases [2].

The Gene Ontology (GO) is a biological repository specific to genes. It is designed to encapsulate the known relationships between biological terms and all genes that are related to these terms [3]. An expanded version of GO's proposed framework for the field of gene regulation is the Gene Regulation Ontology (GRO). GO displays the common knowledge about gene regulation in extremely fine-grained classes whereby using (GRO) the common knowledge for gene regulation is represented in a formal manner [4]. A significant new tool for measuring the gene expression profile of known coding sequences in a particular tissue and time point is microarray innovation. The recent advances of this technology help to propose many approaches to infer the gene networks from these experimental data [1].

Many computational methods are developed to support the GRN inferring from many types of biological data and the analysis of GRN functionality. Recently an increasing interest in constructing GRN using gene expression profiles is growing. Data mining is one of many techniques that try to do that. In order to predict regulatory networks from gene expression patterns, a variety of algorithms, including Bayesian network algorithms [1], neural networks [2], and Boolean

networks [2], are utilized. In general, there are two approaches to build and model GRN, the first approach is based on finding the master regulators set of GRN and obtaining GRN. Then, a mathematical model is employed to help in clarifying the possible regulatory mechanism. This method requires a lot of time and is susceptible to human mistakes in data analysis and interpretation. The second approach is based on applying bioinformatics techniques using genomics data to conclude GRNs. These information methods are suitable to draw an overview about the general structure of gene-gene interactions.

However, the traditional bioinformatics techniques that are used to infer GRN cannot be constructed as a part of an integrated/enterprise functional dynamical system. Therefore, a combination between bioinformatical tools and mathematical modeling tools to employ genome databases with the gene expression data (GED) for constructing GRN is needed. This combination aids to discover existing interactions between genes and reveal the regulatory interactions in specific cell types or experimental samples. As decided in many previous works, Bayesian network is a promising technique to analyze gene expression noisy data. The present work introduces a new method that modifies Bayesian network to use gene regulation properties that can be retrieved from GRO with gene expression profiles to improve the performance of Bayesian Network (BN) method which is used by the most related work in [4]. This research proposes the use of a Bayesian network to infer a GRN from gene expression data using an intelligent computational system called Gat2Get.

The remainder of this article is structured as follows: The problem that the current work tries to solve is defined in section 2. A background about GRN, and microarray methodologies, Gene regulatory ontology and Bayesian network learning technique is provided in section 3. The related works of our approach are presented in section 4 surveys related works. Section 5 presents the proposed approach and the data used. Section 6 shows the experimental amongst these methods, while Section 7 discusses the results. Finally, Section 8 presents conclusions as well as several recommendations for further research.

2 PROBLEM DEFINITION

Numerous gene expression levels are measured in microarray research under various settings or samples. A matrix of actual values serves as the representation for the gene expression set (Figure 1). M = $\{X_{ij} | 1 \le i \le p, 1 \le j \le n\}$ where the rows ($R = \{\overrightarrow{g_1}, \dots, \overrightarrow{g_p}\}$) from the genes activities, the columns ($X = \{\overrightarrow{x_1}, \dots, \overrightarrow{x_n}\}$) defined the expression outlines of samples, and every cell X_{ij} is the evaluated expression level of gene i in sample j. Where, p is genes, n is number of time series samples, $\vec{g}$ is a gene vector, $\vec{S}$ is a sample vector.

	gene$_1$	gene$_2$	gene$_3$		gene$_P$
Time$_1$	X_{11}	X_{12}	X_{13}		X_{1p}
Time$_2$	X_{21}	X_{22}	X_{23}		X_{2p}
Time$_3$	X_{31}	X_{32}	X_{33}		X_{35}
. . .	. . .	. . .	. . .		. . .
Time$_n$	X_{n1}	X_{n2}	X_{n3}		X_{np}

Figure 1. A gene activity matrix (edited from[4]).

Using these expression data, this work needs to infer GRN standing to fact that the number of expressed gene in a specific tissue is enhanced or inhibited (increased or decreased) by the effect of the products of other genes (Figure 2). Here, this work needs to see if vice versa is true. The authors need to know how accurately gene regulation network can be predicted based on its gene expression profiles.

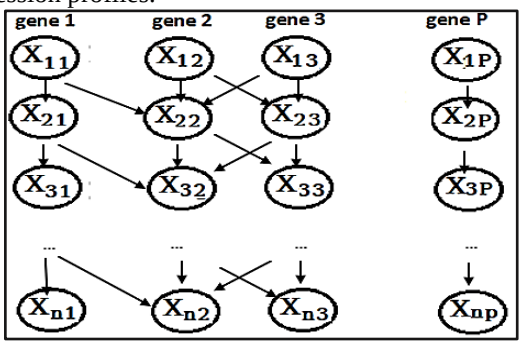


Figure 2: Graphical view of GRN (edited from [3])

For two genes W and Z, the common data of X and Y can be expressed as in Eq. 1 [2] where $\mathcal{I}(w)$ and $\mathcal{I}(z)$ are the entropy of the random values of w and z, respectively; $\mathcal{I}(w, z)$ is the joint entropy of w and z.

$$\mathcal{C}(W, Z) = \mathcal{I}(w) + \mathcal{I}(z) - \mathcal{I}(w, z) \quad (1)$$

$\mathcal{I}(w)$ can be computed as in equation 2 [2], where p(w) is the probability that the W takes x:

$$\mathcal{I}(\mathbf{w}) = -\sum_{\mathbf{w}\in\mathbf{W}} \mathbf{P}(\mathbf{w})\mathbf{logp}(\mathbf{w}) \quad (2)$$

The joint entropy of W and Z is represented by equation 3 [2], where p(w, z) is probability of W and Y:

$$\mathcal{I}(\mathbf{w}, \mathbf{z}) = -\sum_{\mathbf{w}\in\mathbf{W}, \mathbf{z}\in\mathbf{Z}} \mathbf{P}(\mathbf{w}, \mathbf{z})\mathbf{logp}(\mathbf{w}, \mathbf{z}) \quad (3)$$

In general, the process of inferring GRN is a complex task due to the complex nature by which genes interact with each other. This complexity nonlinear increases in case of using expression data which is huge and noisy data.

We attempt to reflect the light on the analysis of high-throughput gene expression data from two distinct domains in this work: gene expression and gene regulation ontology. Bayesian network as a promising data mining technique is the most suitable method to deal with this type of data source. But there is a need to improve its results in terms of accuracy and the quality

of the resulting GRN. This work introduces a new approach for mining gene regulation network based on quantitative analysis of gene expression. This work claims that extending the Bayesian Network technique to include gene regulation properties will improve the performance and the accuracy of results.

This work proposes a novel platform, Gat2Get (gene activity to gene regulation network using Bayesian network) for inferring GRNs from time series GED. For accurate GRN, Gat2Get implements Mutual information (MI) to measure the relationship between each two related genes for reconstructing GRN. The threshold is used to estimate the candidate genes of each target gene. Ignoring the low-related genes acts to decrease the dimensionality of the data approach. The candidate genes are used to estimate the regression factors as regulatory forces to reconstruct GRN using Bayesian network.

3 BACKGROUND

3.1 GRN and GED

A collection of genes, proteins, small molecules, and their interconnections is referred to as the Gene Regulatory Network (GRN). Through the production of protein and RNA, these connections can either directly or indirectly control each other's expression rates. In fact, GRN discovering and understanding these regulation processes is one of ultimate goals of bioinformatics. This understanding gives some idea about gene pathways and helps many potential applications in medicine. Drug discovery, and toxicology analysis are the most important examples of these applications in addition to the complex genetic diseases [5]. Gene regulation networks can be modeled as graphs (see Figure 3) where Nodes demonstrate the functional units (genes, proteins, metabolites, etc.) and Edges represent dependencies (represent the molecular reactions between the nodes) [6].

The traditional techniques of GRN discovery are expensive and time-consuming experiments. In fact, reverse engineering determines the probable link between genes from gene expression data (GED). However, such approaches represent a bottleneck that restricts the understanding of biological systems. Numerous answers to this challenge are provided by biotechnology, such as microarray, which counts the number of mRNA copies of each known coding sequence in a given tissue and time point [6]. The levels of mRNA expression in the tissue sampled can be determined by a single microarray experiment. The absolute expression level in one sample is important, but the topic of how expression varies between samples is more intriguing. Large-scale transcriptional alterations that reflect separate active processes in the various circumstances can be found using this type of experiment. A gene expression experiment's usual dataset contains thousands of genes and dozens of conditions (Ament et al., 2018). The challenge here is to develop acceptable models that accurately predict the interactions between genes from GED [7]. Many modern techniques are introduced to employ machine learning algorithms, mathematical optimization techniques and data mining techniques in inferring GRN from GED.

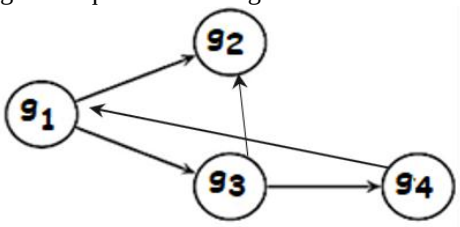


Figure 3. Graph consisting of 4 nodes and 5 edges.

The traditional techniques of GRN discovery are expensive and time-consuming experiments. In fact, reverse engineering determines the probable link between genes from gene expression data (GED). However, such approaches represent a bottleneck that restricts the understanding of biological systems. Numerous answers to this challenge are provided by biotechnology, such as microarray, which counts the number of mRNA copies of each known coding sequence in a given tissue and time point [6]. The levels of mRNA expression in the tissue sampled can be determined by a single microarray experiment. The absolute expression level in one sample is important, but the topic of how expression varies between samples is more intriguing. Large-scale transcriptional alterations that reflect separate active processes in the various circumstances can be found using this type of experiment. A gene expression experiment's usual dataset contains thousands of genes and dozens of conditions (Ament et al., 2018). The challenge here is to develop acceptable models that accurately predict the interactions between genes from GED [7]. Many modern techniques of bioinformatics are introduced to employ machine learning algorithms, mathematical optimization techniques and data mining techniques in inferring GRN from GED.

According to [4], these techniques are divided into two kinds: equation-based techniques and dependence-based techniques. The first type is the equation-based techniques, where GRN is defined using equations to catch the interactions between genes using optimization methods which effects their ability of parameter assessment for the high dimensionality of candidate controllers [8]. Various computational techniques are proposed to deal with GRN inferring problems, the typical methods involve network component analysis, linear programming, Bayesian networks and random forests. The Bayesian network is based on the joint likelihood distribution of GED to organize a directed acyclic graph of GRN. With the second type, GRN is predicted based on identifying the dependencies among genes by determining the linear and nonlinear correlations. However, the results of these methods

include many redundant links. Examples of this type are Pearson correlation coefficient, mutual information (MI), and the Granger method [4]. Mutual information (MI) is an assessment technique of the relationship between genes.

3.2 The Curse of Dimensionality

The high dimensionality and noisy characteristics of microarray data represent great challenges to scientists who attempt to work with these data. These characteristics play an important role in determining the machine learning algorithms that will be utilized and can drive the extension of existing techniques [9]. The high dimensionality is one of the chief tasks with microarray information. Using a vector to represent a 10,000 gene microarray experiment and forcing it to operate in a 10,000-dimensional space. A method must be able to deal with the dimensionality of this feature space reliably in order to be effective [10].

Microarray data always suffers from a high percentage of noise due to technical limitations. Anyone wishing to work with this data must normalize it by scaling the array results by the suitable parameter due to the noise. There is a group of techniques that can be applied to fix the bias and inaccuracies caused by microarray experiments, like using logarithms to analyze and normalize the unnormalized data [11].

3.3 Gene Regulation Ontology

Gene Regulation Ontology (GRO) is an extended model from Gene Ontology specially for the gene regulation domain. It explores processes and entities (transcription factors or genes) that are related to the gene expression regulation, in terms of ontology classes and relations between classes [12]. The GRO terms are typically generated from external ontological resource such as : Sequence Ontology-SO (sequence regions and attributes of sequence regions, such as gene, binding site, promoter, DNA, RNA), and Gene Ontology-GO (molecular functions, biological processes, cellular components, such as regulation of transcription, nucleus).

3.4 Bayesian Network

A directed acyclic graph (DAG) named G is the basis of the probabilistic statistical method known as a Bayesian network (BN). G is used to describe the probability dependencies. DAG G can be thought of as a collection of nodes N and links E that join the nodes. BN as a classifier has the capacity to forecast the likelihood of class members, i.e., the likelihood that a specified set of data is part of a specific class. Ajmal and Madden in [1] discuss the Bayesian Network (BN) as a straightforward diagram where every node is a data member, and each edge denotes a probabilistic relationship as (see Figure 3) [11].

Given that there is a set classes A_1, A_2, . . . , A_i where the classifier forecasts the class of an unknown sample S with the highest posterior probability. This means that, S is assigned to class Ai if and only if $P(Ai|S)>P(Aj|S)$ for $1 \leq j \leq I$ and $j \neq i$. Bayes Theorem implies that: S is assigned to class Ai if and only if $P(Ai|S)>P(Aj|S)$ for $1 \leq j \leq i$ and $j \neq i$. Bayesian Rule [1]:

$$\mathbf{P(h|S)} = \frac{\mathbf{P(S|h)P(h)}}{\mathbf{P(S)}} \quad \textbf{(4)}$$

The class-conditional probability will be [1]

$$\mathbf{P(A_i|S)} = \frac{\mathbf{P(S|A_i)P(A_i)}}{\mathbf{P(S)}} \quad \textbf{(5)}$$

Where each attribute set $S = \{s_1, s_K, \ldots, s_n\}$ consists of d attributes. The relative frequency of instances with value aj as the j^{th} feature in class Ci is approximated as P(ajAi) if the i^{th} parameter is classified. If the j^{th} feature is uninterrupted, on the other hand, P(ajAi) is often approximated using a Gaussian density function. Both situations are easily calculable. Compared with other classifiers, BN has some advantages: 1- BN is easy to generate, as the structure is given a priori. 2- it has a very effective classification process. 3- it gives the best accuracy and performance in terms of training time. 4- can use data with missing values as inputs, during classification, In time that decision tree and neural network cannot [3].

Dynamic Bayesian network (DBN) is a BN which introduced a time-variant network in which the present situation is affected by the last situation. The distribution over n timeslots is expressed by equation 6:

$$\boldsymbol{P(S_{1:n})} = \prod_{t=1}^{n} \prod_{i=1}^{p} \boldsymbol{P(S_1^i \,|Pa(S_t^i))} \quad \textbf{(6)}$$

Using DBN, each node at a specific timeslot (t) is based on the state of its parent nodes at the preceding timeslot (t-1). One gene can be represented by several nodes related to the number of its samples on gene expression microarray. Here, the states of a gene are separated into a number of timeslots during modeling a cyclic regulation by DBNs.

4 THE RELATED WORKS

A lot of works tried previously to discover GRN from the expression data using Bayesian network. They were varying in the method of applying Bayesian network. The following subsections describe these trials. For instance, [13] modified the K2 method and created a BN architecture learning algorithm using Friedman's scoring function. They evaluated it using the REVEAL method to rebuild both artificial and actual yeast networks. Friedman's scoring function exhibits greater precision and recall when a stationary correlation is introduced throughout two successive time slices. They concluded that Friedman's score measurements for BN may be utilized to recreate transition networks and have a significant potential to increase the precision of gene regulatory network structure prediction with time series gene. However, the obtained accuracy is still low (0.76) and BN can obtain more accuracy than that already

obtained from this work. Also, we believe that this work has ignored the important information about genes during training their algorithm and this may be the cause of producing non- meaningful regulatory network.

The study by [14] tried to resolve the difficulties in implying GRN from gene expression data. A unique non-parametric learning strategy depending on nonlinear dynamical systems was also presented. These non-parametric techniques more precisely infer network architectures than do conventional methods for broader GRNs encompassing numerous genes, but at a powerful processing expense. They stated that their approach had produced low performance and it didn't perform better than their previous work. Again, we still believe that this work ignored important information about genes during training its algorithm and this may result in non-meaningful regulation network.

A new approach for identifying connections involving genes based on numerous expression data was developed in [15]. The foundation of this approach is the representation of statistical interdependence using Bayesian networks. Both a revolutionary search algorithm (Friedman) and a method for assessing statistical data served as the foundation of their approach. They used their techniques on the actual expression data without any previous information, and they were able to identify causal connections and additional gene relationships besides strong correlation. But with significantly low performance. So, they planned to apply a learning approach (that is what this work tries to do). Many authors, including the proposed work, think that such methods have numerous shortcomings. (1) A regulatory link between two genes may not always exist just because their expression levels are identical. (2) Despite the existence of a real connection, it can be difficult to determine which is the regulator and which is the aim. (3) Because of the complexity and various layers of gene regulation in addition to the potential resulting latency, these strategies can only diagnose a restricted number of regulatory connections, which makes the regulatory connection between a regulator (an activator or a repressor) and its objective very elusive and challenging to identify.

The work by [16] employed BN to describe interactions among co-clustered genes following clustering relying on Gene Ontology to increase performance. Additionally, they presented a novel technique for adding time data to BN via cross-correlation among co-clustered genes. Reconstructing the regulatory network of 84 yeast genes using this strategy. Their approach improved the accuracy from 66% to 72%. We still see that the accuracy resulted by this work is unpromising. We still aspire to better accuracy. It is believed that the cause of low accuracy is the absence of regulation information during training BN algorithm, and this may result in non- meaningful regulation network.

The work by [17] is the most related work to Gat2Get, although it didn't use Bayesian network. Target pattern recognition is a key component of their strategy for determining gene regulatory connections. Two stages were taken to complete the pattern recognition: Finding genes with recognized target genes (KTGs) of each analyzed regulator that have expression patterns identical to their own was done using a first approach. By looking for regulator-certain binding positions in their promoter sequences, the chosen genes were further screened. They used the method to the recognized target genes of 18 yeast regulator genes and found 267 additional regulatory connections. 36.1% of the newly found target genes shared or were comparable to a KTG of the regulator. Although they used suitable gene regulation properties, the achieved accuracy was not at all promising, this work aims to achieve better accuracy. This work is the most related to our proposed approach as it included the gene regulation properties in their mining techniques. The current work proposes to use these properties in a different form, and with the most suitable classifier (Bayesian Network). They introduced two Bayesian information criterion (BIC)-based BN scoring functions. They found a schema and raised the scores by combining these evaluation metrics with the DBN architecture. In comparison to the BIC score, their BN scoring functions greatly increase the learned graphs' accuracy. Also, the work by [18] is another most related work to Gat2Get. It made an effort to pinpoint the central GRN in charge of a biological activity' choice. They developed an assessing framework for building core GRN using transcriptomics data and biological databases. Their work shows outperforming existing algorithms in inferring GRN.

5 THE PROPOSED FRAMEWORK

This work depends mainly on machine learning using BN. The proposed approach also uses a BN scoring technique that is based on the scoring method by [1]. The used scoring method is derived from Bayesian Information Criterion (BIC) to learn Bayesian network during searching candidate GRN.

Based on Figure 1, X is a matrix of p genes measured across n timeslots, P is a set of measured genes where ($1 \le a \le p$) and N is a set of samples in different timeslots where ($1 \le t \le n$). p^a denotes a set of measured genes excluding a. i.e., $p^a = \{a\}$, and S_{ts}^a is the activity of gene (a) at timeslot t. E_{ab} is the edge between node X_t^a and node X_{t+1}^b . $Par(S^a)$ is the parent of S_t^a in G where G(E) is a set of all edges of GRN. The edges between all pairs of genes are scored using BIC where foreach pair of gene activities (X_t^a , X_{t+1}^b) the possible edge E_{ab} is scored using the conditional dependence between them at t and t+1.

$$\boldsymbol{Score(E) = \sum_{a=1}^{p} \sum_{b \in Par(X^a)} Score(E_{ab})} \quad \textbf{(7)}$$

And the score for the edge E_{ab} that presents in the graph G is expressed in Equation (8).

$$Score(E(G)) = \sum_{a=1}^{p} \sum_{b \in Par(X^a)} Score(E_{ab}(G)) \quad (8)$$

The value of α_{ab} can be used as a score of the edge between the gene activity X_t^a andX_{t+1}^b. A small value of α_{ab} means that there is a conditional dependency between the two genes (no null hypothesis). So, the value of (1-α_{ab}) can reflect the score strength of the edge between X_t^a andX_{t+1}^b. Using the value of α_{ab} to be assigned to edge E_{ab} which is equivalent to $Max_{k \neq b}(P_{ab|k})$ for each pair of genes (p-1 genes), i.e., the highest score. The edge score can be computed using Equation (9).

$$Score(E(G)) = \sum_{a=1}^{p} \sum_{b \in Par(X^a)} log(1 - \alpha_{ab} + \gamma) \quad (9)$$

Where γ is constant that used to push the probability far from zero. This value can be chosen randomly and verified through practical experiments.

As outlined from the related works, this work is motivated to enhance the significant low performance or accuracy and see that the wealthy input data and the absence of the regulation information about genes during training phase is the cause of the resulted non-meaningful regulation network. The proposed framework is using a flow diagram (Figure 4). It goes in five main phases in logical order to optimize the classification performance and accuracy. The operations of the proposed approach are described in Figure 4 as the following :

- Prepare normalized input data by reduction its complexity and dimensionality by using dimension reduction method. (Performance optimization step)
- Reformat the output data by converting it into binary matrix between the known genes and their regulators.(accuracy optimization step)
- Construct predictor corresponding to gene expression data input and corresponding gene regulators as a target.

In addition, the core operation of proposed framework as depicted in Figure 4 is divided into two parts, the GRN construction subsystem (process # 5 in Figure 4) and the GRN learning subsystem (process # 4 in Figure 4).

5.1 GRN Construction Subsystem

This subsystem is responsible for the construction of GRN from the gene activity data. In other words, the main role of this subsystem is to convert gene expression data to creation of a directionless gene interaction network. The process of this subsystem includes five steps that are depicted in Figure 5 as the following:

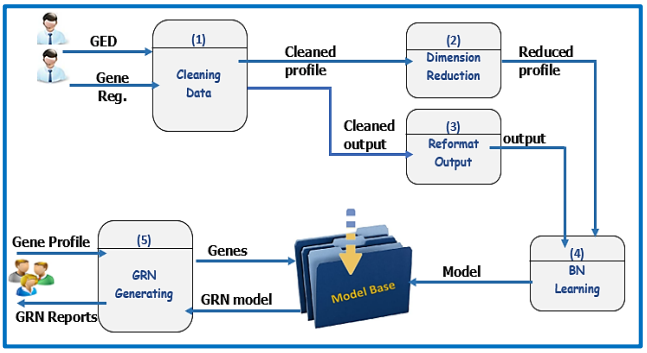


Figure 4. Flow diagram of Gat2Get

Step 1: Convert genes-samples matrix to gene-gene matrix and reduce the dimension using the mutual information algorithm for reducing the false negative rate (FPR) and increase true positive rate (TPR). As shown in Figure 5, MI technique using different thresholds divides the candidate genes into three categories (low, mid, and high) dependent genes. Ignoring independent genes will reduce the dimensionality.

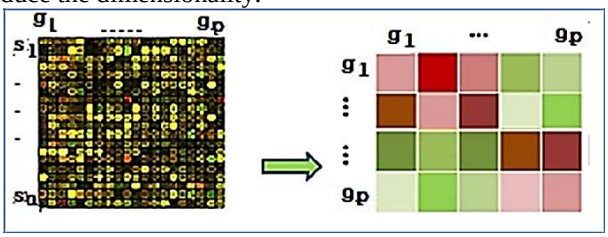


Figure 5. Step 1 in GRN Construction.

Step 2: For each gene in the high-dependent genes category, estimate the score parameters with the remaining genes as an indicator for regulatory strengths. Furthermore, the high-dependent genes are used as constraint of the model (See Figure 6).

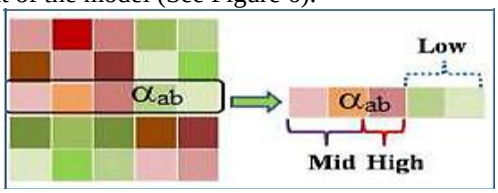


Figure 6. Step 2 in GRN Construction.

Step 3: Figure 7 depicts the core technique of Gat2Get approach. According to the strength of dependency of the given gene current gene, the regulatory genes are split into three regulatory types: weak, or mid, and strong.

5.2 GRN Learning Subsystem

As described in Figure 4, Bayesian network Learning process receives two input datasets, the gene expression profile, and gene regulator vector.

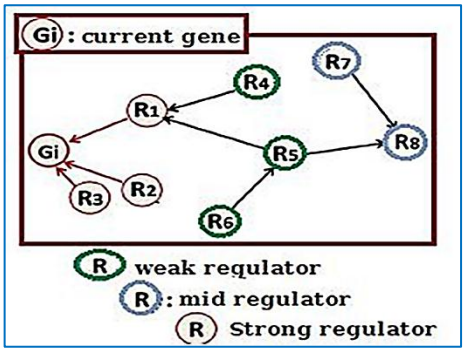


Figure 7. step 3 in GRN Construction

This process uses the first input to determine the similar gene profile and the second to identify the related genes. The following algorithm describes the process of BN Learning subsystem in Figure 8:

Inputs→ GED, GRN

Step (1): Prepare the entire datasets of all experiments and regulators to extract the input/output matrix.
Step (2): Randomly partition the set of data into 80% training set and 20% testing set.
Step (3): Segment training samples into three groups randomly
Step (4): For each group ∈ training samples groups do:
    Choose one group for validation operation.
    Reserve other two groups for calibration
    Train Bayesian network classifier
    Test the model.
    For each model using all of 3 groups for validation
    Optimize the calibration.
Loop

Outputs: optimized Bayesian network Model for inferring network G

Figure 8. BN Learning algorithm

6 EXPERIMENTAL STUDY

The proposed approach (Figure 4) is implemented in Java using an edited version of Jbayes class and MS Naive Bayes algorithms of MS SQL Server Analysis Services (SSAS). Data Cleaning is performed using the Info Sphere Information Server for Data Quality from IBM. Dimension reduction is implemented using PCA of Microsoft SQL server. Re-formatting is implemented via ProM version 6 ,MS SQL server, MS Excel also used.

6.1 Gene expression Profiles

The source of dataset that used by the present work is an (Array Express) which generates gene expression matrix by normalized row data file. The following Table (1) presents a properties summary of the real dataset.

Table 1: The used gene expression properties

Name	Mouse gene dataset
Number of samples	55
Number of genes	7388
Type of source	Microarray

The dataset that is studied by the works by Friedman et al. in [15], Haoi et al. [13] and Hairong et al. [17] is chosen for this work to simplify the result comparison. The dataset is first normalized by eliminating the maximum calculations that differ than the above two units (to reduce false-positive in results). The scores are thus an average of and Median-subtracted. Percentages under zero are often not biologically significant, according to prior research Hovattam et al. [19]. Therefore, any percentages less than one are adjusted to one. Less than 0.01% of all data points are lost, and those scores are equated to zero. 5016 genes are eventually kept and entered into the database for research. Table 2 presents a sample (a few rows & columns) of the resulted normalized expression dataset. The first column represents the genes' codes and other columns determine the expression level at different points of time with a header row. Each row signifies the gene expression levels of a specific gene at different time points or different conditions.

Table 2 : Sample of GED after modification

IN_XM	Epididym	Eye	Adrenal	Aorta	Colon	Digit	Cortex
XM_1219781	0.000	0.637	0.149	0.259	0.000	2.618	0.685
XM_1219801	0.166	0.113	0.669	2.151	1.192	0.000	0.330
XM_1219811	0.000	0.000	0.000	0.000	0.337	0.000	0.556
XM_1219831	0.092	0.100	0.000	0.502	0.422	0.214	0.091
XM_1219851	0.722	2.154	0.000	1.317	1.746	4.142	0.000
XM_1219861	0.000	0.000	0.042	0.497	0.741	0.058	0.000
XM_1219871	0.000	0.174	0.089	0.000	1.371	0.686	0.000
XM_1219881	0.836	0.749	0.232	0.177	2.005	1.331	0.000
XM_1219891	0.000	0.346	0.000	0.115	0.656	0.000	0.000
XM_1219901	0.285	0.142	0.441	0.682	2.308	0.300	0.000

6.2 Gene Regulators Vectors

Mouse GO-Regulators that used by [17] are obtained from the Gene Ontology [20] based on the knowledge from EBI [21] and both are mapped (as the expression profiles) to XM gene to the regulators. Regulators with less than 100 genes among the expressed genes are not included in our investigation because the statistical tests do not apply to them.

Note that the utilized regulation information that we obtained is not found directly in any ontology or annotation store. The first step is to recognize GO terms with appropriate GRO terms. Then use a quick tool from EMBL-EBI as fast browser for Regulation Gene Ontology terms and annotations. The regulator chosen if its affected genes include any gene in the expression matrix.

6.3 Applying Gat2Get Method

To train the classifier, the regulator dataset is prepared. The regulators with a gene participation of 140 or less are removed. The result is twenty regulators set. Table 3 summarizes a sample of these regulators. Three-fold cross validation experiments were used on an Intel core(TM)2 Duo PC with an individual 2.66 GHz processor and a 2 GB RAM. The used software is Microsoft Windows 8 with data mining service from Microsoft .NET SSAS. These analysis Services provide an integrated platform that incorporates data mining to create business intelligence solutions with predictive analytics.

The implementation of Bayesian network with SSAS is straightforward. It is needed to set some parameters to ensure completing the process with reasonable accuracy and quality. Because SSAS-Bayesian computes the likelihood of each situation of every entry column with every probable situation of the expected column, the *minimum dependency probability* between input and output attributes must be specified between(0, 1).

Table 3: Sample of selected regulators

Regulator	Used Code	+ve genes
YDL210W (UGA4)	R 1	632
YJR048W (CYC1)	R 2	427
YEL039C (CYC7)	R 3	411
YGL009C (LEU1)	R 4	356
YLR355C(ILV5)	R 5	332
YPR124W (CTR1)	R 6	288
YDR146C (SWI5)	R 7	272
YDR516C (EM12)	R 8	219
YOR074C (CDC21)	R 9	202
YDL102W (CDC2)	R 10	199
YML123C (PHO84)	R 11	191
YBR093C (PHO5)	R 12	186
YGL008C (PMA1)	R 13	179
YDR064W (RPS13C)	R 14	170
YGR254W (ENO1)	R 15	169

By this way, the size of generated content is controlled. Larger value reduces the number of attributes in the model. Dependency probability is set to 0.5 to return only those inputs that are more likely than random to be correlated with the output. Class priorities are another parameter that needs to be set specify. It refers to the probability that any given input will be in class regardless of other information. This parameter is set automatically to allow SSAS-Bayesian to estimate it from the training set by computing the fraction of training records that belong to each class.

7 RESULTS

7.1 Evaluation of Classification Techniques.

Finding the accuracy rate is an important part of any model evaluation. Confusion matrix, which is a convenient metric to understand the experimental results, is used for this task. A confusion matrix displays the enumeration of the real against forecast class scores. It displays in what manner the approach forecasts and offerings the facts almost effects might have gone incorrect. Table 4 is a model confusion matrix. The classifier assigns TP + FP illustrations to the positive class(True Positives, False Positives) and TN + FN illustrations to the negative class (True Negatives, False Negatives). Accuracy and Precision of the classifier are usually used to measure the quality of classification [1]. They are calculated from the following equations (11-16)

Table 4: Binary classifiers confusion matrix

		Predicted Class	
		False	**True**
Actual Class	**False**	**TN** True Negative	**FP** False Positive
	True	**FN** False Negative	**TP** True Positive

P = T P + FN (11) N = TN + FP (12)

$$\text{FPrate} = \frac{\text{FP}}{\text{N}} \quad (13) \qquad \text{TPrate} = \frac{\text{TP}}{\text{P}} = \text{Recall} \quad (14)$$

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{P+ N}} \quad (15) \qquad \text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (16)$$

Table 5 presents the results that are obtained from the introduced model based on the previous equations . The following observations are made from the previous results that are obtained during the training model with the present approach are noticed:

Table 5: Selected gene Regulators used for comparisons.

Reg.	+ve$_s$	TP	TN	FP	FN	P	N	Precision	Accuracy	FPrate	Recall
R 1	632	31	791	124	54	85	915	0.20	0.822	0.14	0.36
R 2	427	20	811	122	47	67	933	0.14	0.831	0.13	0.30
R 3	411	10	888	57	45	55	945	0.15	0.898	0.06	0.18
R 4	356	22	839	110	29	51	949	0.17	0.861	0.12	0.43
R 5	332	0	958	3	39	39	961	0.00	0.958	0.00	0.00
R 6	288	5	930	30	35	40	960	0.14	0.935	0.03	0.13
R 7	272	15	835	124	26	41	959	0.11	0.85	0.13	0.37
R 8	219	9	927	53	11	20	980	0.15	0.936	0.05	0.45
R 9	202	15	868	100	17	32	968	0.13	0.883	0.10	0.47
R 10	199	7	884	90	19	26	974	0.07	0.891	0.09	0.27
R 11	191	6	940	36	18	24	976	0.14	0.946	0.04	0.25
R 12	186	11	875	99	15	26	974	0.10	0.886	0.10	0.42
R 13	179	9	840	142	9	18	982	0.06	0.849	0.14	0.50
R 14	170	4	961	18	17	21	979	0.18	0.965	0.02	0.19
R 15	169	8	921	54	17	25	975	0.13	0.929	0.06	0.32
R 16	162	0	961	16	23	23	977	0.00	0.961	0.02	0.00
R 17	146	6	884	91	19	25	975	0.06	0.89	0.09	0.24
R 18	145	2	955	26	17	19	981	0.07	0.957	0.03	0.11
R 19	141	5	916	61	18	23	977	0.08	0.921	0.06	0.22
R 20	106	1	965	9	25	26	974	0.10	0.966	0.01	0.04
Average								**0.11**	**0.91**	**0.07**	**0.26**

- The input matrix with the regulator data is hard to classify. This may be due to the presence of some noise in expression values.
- Bayesian inference within the introduced method success to achieve 97% accuracy for some regulators with an average of 91% for all. As expected before, Bayesian Network with regulator information outperforms the previous work that obtained maximum 76% of accuracy.
- The closest works to compare against the present work are by [15] -A, [13]-B and [17]-C. Taking the regulation dataset size in addition to the expression dataset size into consideration, the presented approach results give a more promising performance and accuracy. (Figure 9)
- False negative results from machine learning are not necessarily biological false negative [18] It may be reasonable when a gene that was assigned to a regulator based on profile similarity has a specific activity that demands a different regulation strategy.

7.2 Lift Chart Analysis

Lift chart analysis is a popular graphical technique to assess the performance of classification approaches. Lift chart is a beneficial procedure for visualizing, establishing, and choosing classifiers depending on their accuracy [18].

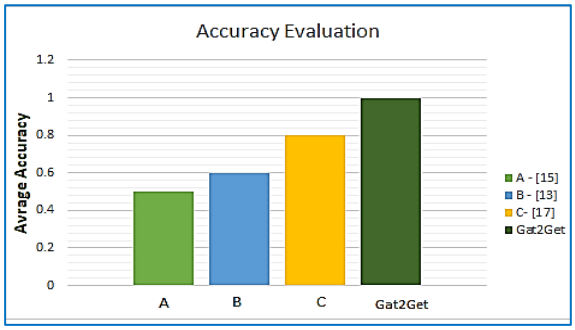


Figure 9: The accuracy of Gat2Get and related works

The likelihood that a classifier would score a randomly selected positive occurrence better than a randomly selected negative occurrence is represented by the area under the curve (AUC) in lift charts. A better AUC hence denotes greater overall success. For various sized random samples in Lift charts, diagonal lines yield the anticipated outcome. In the instance of the Lift chart, the precise coordinates of each indicated point that appears on the graphical region of the two illustrations match the likelihood of the goal value of the related cases. Figure 10 shows the lift charts from the present experiment that give a better impression about the performance of the introduced method [18][22-24]. The better performance is due to the dependence of Bayesian learning on well reduced profile in addition to the use of gene regulatory.

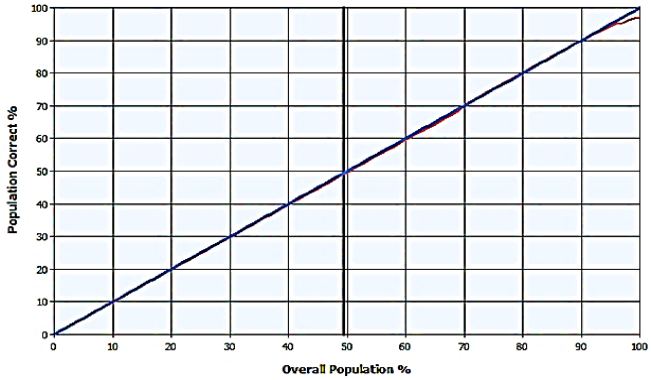
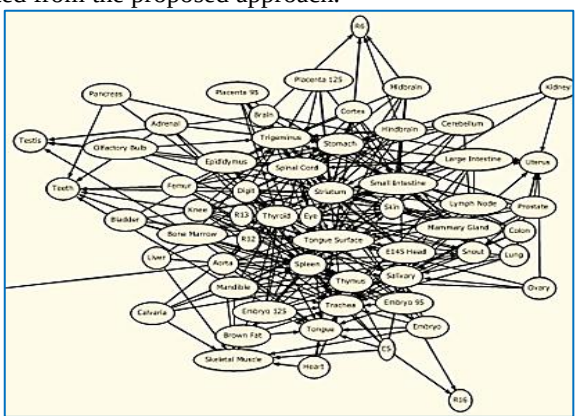


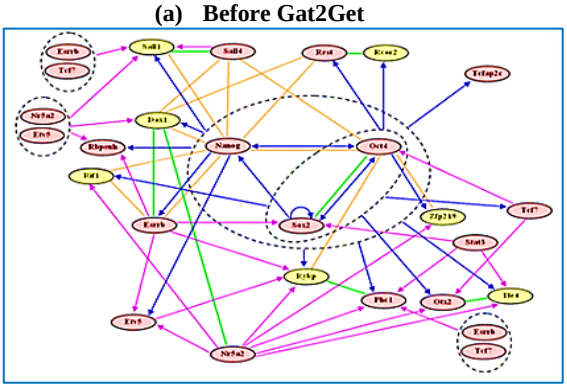
Figure 10 : lift charts from Gat2Get experiment

7.3 From Biological View

Gat2Get mainly tries not only to obtain better accuracy, but also to improve the quality of resulting gene regulation network from the biological view. The idea depends on including the regulation properties in learning steps instead of depending only on the expression data as a source of genes relations. From biological view, the presented approach with including the regulation properties to learning phase has outperformed other works that depend only on the expression information. The resulting GRN is less complex, more readable, more meaningful and the relations between genes are clearer. Figure 11 (a, b) illustrates the complex GRN that resulted without the proposed approach compared with meaningful GRN that resulted from the proposed approach.



(a) Before Gat2Get



(b) with Gat2Get

Figure 11: The resulting GRNs

8 CONCLUSION AND FUTURE WORK

This work introduces a new approach for inferring gene regulation network from gene expression data. By implementing this proposed approach, we can conclude that the including of gene regulation properties improves the process of capturing natural classes during inferring the relations between genes from expression data. This can decrease the dimensionality of the target network. In addition to its role in improving the result accuracy. High complexity and poor quality of gene expression data acquired by high-throughput methods like microarray provide two difficulties in the context of the current issue. That is the cause of our decision to use Bayesian network. Our experiment shows that BN is efficient to some level in classifying such high dimensional data even without dimension reduction, but in the presence of regulatory information. When evaluating constrained probabilities from data, the Bayesian network can average out noise points to produce good results. It can also manage outliers by suppressing them through approach construction. In fact, the inclusion of gene regulation properties can be the main factor of improving the quality of the resulted GRN as it acts to reduce the dimensionality by avoiding the unreal relations between genes. The low poor performance in terms of the number

of False Positive (FPs) of some regulator's classes may be due to some biological causes, or the high noise in the dataset. The nature of noise and highly dimensional of the data from real-world application domain as DNA microarrays still represent particular challenges for machine learning methods. Dealing with such big and noisy data can act as a source for future research directions in machine learning. It is planned to extend this study to identify the hidden regulators and investigate the impact of these missing regulators on the gene profile and so on the accuracy of BN.

9 REFERENCES

1. Ajmal H. B. and Madden M. G., "Dynamic Bayesian Network Learning to Infer Sparse Models from Time Series Gene Expression Data," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 19, no. 5, pp. 2794-2805, 1 Sept.-Oct. 2022, doi: 10.1109/TCBB.2021.3092879.
2. Xu C. and Jackson S. A., "Machine learning and complex biological data," Genome Biol., vol. 20, 2019, Art. no. 76.
3. Yang, B., Xu, Y., Maxwell, A. et al. MICRAT: a novel algorithm for inferring gene regulatory networks using time series gene expression data. BMC Syst Biol 12 (Suppl 7), 115 (2018). https://doi.org/10.1186/s12918-018-0635-1
4. Jiang, X., Zhang, X. RSNET: inferring gene regulatory networks by a redundancy silencing and network enhancement technique. BMC Bioinformatics 23, 165 (2022). https://doi.org/10.1186/s12859-022-04696-w
5. Ament SA, Pearl JR, Cantle JP, Bragg RM, Skene PJ, Coffey SR, Bergey DE, Wheeler VC, MacDonald ME, Baliga NS, Rosinski J, Hood LE, Carroll JB, Price ND. Transcriptional regulatory networks underlying gene expression changes in Huntington's disease. Mol Syst Biol. 2018 Mar 26;14(3):e7435. doi: 10.15252/msb.20167435. PMID: 29581148; PMCID: PMC5868199.
6. Chan TE, Stumpf MPH, Babtie AC. Gene regulatory network inference from single-cell data using multivariate information measures. Cell Syst. 2017.
7. Carre C, Mas A, Krouk G. Reverse engineering highlights potential principles of large gene regulatory network design and learning. Npj Syst Biol Appl. 2017;3:17.
8. Huynh-Thu V. A. and Geurts P., "dynGENIE3: Dynamical GENIE3 for the inference of gene networks from time series expression data," Sci. Rep., vol. 8, no. 1, Feb. 2018.
9. Yeung KY, Dombek KM, Lo K, et al. Construction of regulatory networks using expression time-series data of a genotyped population. Proc Natl Acad Sci U S A. 2011;108(48):19436–41. ; 2018.
10. Katebi A, Kohar V, Lu M. Random parametric perturbations of gene regulatory circuit uncover state transitions in cell cycle. iScience. 2020;23:101150.
11. Li Y. and Ngom A., "The max-min high-order dynamic Bayesian network learning for identifying gene regulatory networks from time-series microarray data," in Proc. IEEE Symp. Comput. Intell. Bioinf. Comput. Biol., 2013, pp. 83–90.
12. Juanes Cortés B, Vera-Ramos JA, Lovering RC, Gaudet P, Laegreid A, Logie C, Schulz S, Roldán-García MDM, Kuiper M, Fernández-Breis JT. Formalization of gene regulation knowledge using ontologies and gene ontology causal activity models. Biochim Biophys Acta Gene Regul Mech. 2021 Nov-Dec. doi: 10.1016/j.bbagrm.2021.194766. Epub 2021 Oct 25. PMID: 34710644.
13. Haoni L., W. Nan, G. Ping, J. Edward, Z. Chaoyang; Learning the structure of gene regulatory networks from time series gene expression data; BMC Genomics 2011.
14. Christopher A. and L. David; How to infer gene networks from expression profiles, revisited; Interface Focus (2011) 1, 857–870.
15. Friedman N, Linial M, Nachman I, Pe'er D. Using Bayesian networks to analyze expression data. J Comput Biol. 2000;7(3-4):601-20. doi: 10.1089/106652700750050961. PMID: 11108481.
16. Yavari F., Towhidkhah F., Gharibzadeh Sh.; Gene Regulatory Network Modeling Using Bayesian Networks and Cross Correlation; IEEE; 2008.
17. Hairong W., K. Yiannis; Inferring Gene Regulatory Relationships by Combining Target–Target Pattern Recognition and Regulator-Specific Motif Examination; Wiley Periodicals Inc; 2004.
18. Su, K., Katebi, A., Kohar, V. et al. NetAct: a computational platform to construct core transcription factor regulatory networks using gene activity. Genome Biol 23, 270 (2022). https://doi.org/10.1186/s13059-022-02835-3
19. Hovattam I., Kimppa K., A. Lehmussola; DNA microarray data analysis; CSC – Scientific Computing Ltd.; Second edition ; 2005
20. European Bioinformatics Institute: Part of European Molecular Biology Lab; "http://www.ebi.ac.uk/".
21. Gene Ontology Consortium: http://geneontology.org/
22. EL-Geneedy M., Moustafa H., Khalifa F., Khater H. A., AbdElhalim E., An MRI-based deep learning approach for accurate detection of Alzheimer's disease, Alexandria Engineering Journal, Volume 63, 2023, https://doi.org/10.1016/j.aej.2022.07.062.
23. Gamal M. I., Khater H.A., E. A., Applying Artificial Intelligence Techniques to Improve Clinical Diagnosis of Alzheimer's Disease. 9th International Conference on Research in Science and Technology (RSTCONF), 20-22 March 2020 Berlin, Germany.
24. Khater H. A., Baith A. M. and Kamel S. M., A Proposed Technique for Software Development Risks Identification by using FTA Model,

International Journal of Computer and Information Engineering, World Academy of Science, Engineering and Technology, Jan 2013, vol. 73(1).