

# Self-organizing Neural Networks for Efficient Clustering of Gene Expression Data

Ji He\*, Ah-Hwee Tan<sup>†</sup>, and Chew-Lim Tan\*

\*School of Computing, National University of Singapore,  
3 Science Drive 2, Singapore 117543  
{heji,tancl}@comp.nus.edu.sg

<sup>†</sup>Institute for Infocomm Research,  
21 Heng Mui Keng Terrace, Singapore 119613  
ahhwee@i2r.a-star.edu.sg

**Abstract**—Clustering of gene expression patterns is of great value for the understanding of the various molecular biological processes. While a number of algorithms have been applied to gene clustering, there are relatively few studies on the application of neural networks to this task. In addition, there is a lack of quantitative evaluation of the gene clustering results. This paper proposes Adaptive Resonance Theory under Constraint (ART-C) for efficient clustering of gene expression data. We illustrate that ART-C can effectively identify gene functional groupings through a case study on rat CNS data. Based on a set of quantitative evaluation measures, we compare the performance of ART-C with those of K-Means, SOM, and conventional ART. Our comparative studies on the yeast cell cycle and the human hematopoietic differentiation data sets show that ART-C produces reasonably good quantitative performance. More importantly, compared with K-Means and SOM, ART-C shows a significantly higher learning efficiency, which is crucial for knowledge discovery from large scale biological databases.

## I. INTRODUCTION

With the advances in molecular biology research, an increasing number of genes have been discovered and identified. To supplement traditional biological studies that collect expressions of individual genes, there is a natural need to analyze the gene expression data in a global fashion. Clustering of gene expressions (*gene clustering* in short) is one useful analysis technique. Specifically, the knowledge discovered by the clustering process is of great value for various molecular biological processes, such as correlating expression patterns, and mapping expressions data to sequence, structural and biochemical data [13].

While there exists a large number of clustering algorithms in the literature, only a few of them have been applied to analyze gene expression data in the recent half decade. These include K-Means [14], hierarchical clustering [5], graph theory based clustering [2], naive Bayesian clustering [1], and Gaussian mixture model based clustering [18]. There are very rare studies of neural networks on gene clustering, besides a few applications of the self-organizing map (SOM) [10], [15]. SOM however is not known as an efficient clustering algorithm, due to its high computational cost in maintaining the neighborhood relationship.

Adaptive Resonance Theory (ART) models is a family of self-organizing neural networks that performs online clustering of arbitrary input pattern sequences with a high level of efficiency [3]. However, a conventional ART model produces a varying number of output clusters in response to the distribution and the order of the inputs, mainly affected by a

global *vigilance* parameter. This behavior may not be desirable for gene clustering, as a biologist would want to control the number of output clusters directly for the purpose of validation and inspection.

To address the above problem, this paper proposes a relatively new ART variance, known as Adaptive Resonance Theory under Constraint (ART-C), for efficient clustering of gene expression data. ART-C is capable of incorporating user-defined constraint when learning its category representation. Specifically it has been shown to produce clustering results equivalent to those of ART, with the added capability of self-adjusting its vigilance parameter to generate the desired number of categories (output clusters) [8]. We consider this capability of great value as it relieves the trial-and-error process of the user in suggesting a proper vigilance parameter.

The rest of this paper is organized as follows. Section II reviews several clustering algorithms used in our benchmark. Section III summarizes the ART-C learning algorithm. Section IV illustrates ART-C's capability in rediscovering gene functional groupings through case study on rat CNS data. Section V reports our comparative benchmark on the yeast cell cycle and the human hematopoietic differentiation data sets. The last section summarizes our conclusions and proposes the future work.

## II. CLUSTERING ALGORITHMS: A BRIEF REVIEW

This section briefly reviews the clustering algorithms used in our benchmark. Detailed information of the algorithms can be found through the various references.

### A. K-Means

K-Means [16] has been extensively studied and applied in the clustering literature due to its simplicity and robustness. The objective of the K-Means clustering method is to minimize the intra-cluster compactness of the output, in terms of the summed squared error. The algorithm randomly initializes  $k$  reference clusters and iteratively adjusts the prototype of each cluster as the mean of its cluster members. Learning is repeated until the cluster assignment of each input stabilizes. Though simple, K-Means can produce satisfactory clustering results by reaching one of its local optima. The deficiency of K-Means lies in its dependency on the availability of the entire input data set for batch processing, which could be memory intensive. In addition, K-Means is sensitive to the initialization of the reference clusters and the input noises.

## B. Self-organizing Map

Self-organizing Map (SOM) proposed by Kohonen [12] is a family of self-organizing neural networks widely used for clustering and visualization. Similar to K-Means, the reference clusters in SOM are randomly initialized. SOM follows a *winner-take-part* competitive learning process. For each incremental input, the network identifies the winner cluster that is most similar to the input, and updates the weights of the winner as well as the winner's neighbors in order to incorporate the input. Although SOM adopts an online learning paradigm, it is not a fast clustering algorithm, due mainly to the extra computational cost in maintaining the neighborhood relationship and the slow learning rates. The advantage of SOM is the spatial map output in which similar clusters are placed close to each other. Specifically, 2-dimensional maps are widely used in various data visualization tasks.

## C. Adaptive Resonance Theory

Besides SOM, Adaptive Resonance Theory (ART) [7] is another family of self-organizing neural architectures well-known by its *stability-plasticity* property. Unlike SOM that initializes a pre-specified number of reference clusters, the recognition categories (clusters) in ART are dynamically created using input samples. The network essentially follows a *winner-take-all* competitive learning process, with extra binary decision that triggers the network's state to either *resonance* or *reset* in response to each input, guided by a global *vigilance* parameter. If an input is dissimilar enough to the existing recognition categories in the network, such that the network fails to find a winner to reach a resonance, the input is inserted into the network as a new recognition category. This mechanism is effective in encoding distinct inputs and guarantees a high learning efficiency. The deficiency of ART is that the number of its output clusters is not directly determinable. In order to obtain a specific number of clusters over the input space, prior knowledge on the distribution of the data set is required to suggest a proper vigilance parameter.

There are a large number of ART variances, mainly depending on the pattern similarity measures used and the network's search process. The ART 2A [4] module, which uses dot product as the pattern similarity measure, is applied in our work due to its close relationship with K-Means and SOM.

## III. ADAPTIVE RESONANCE THEORY UNDER CONSTRAINT

Adaptive Resonance Theory Under Constraint (ART-C) [8] is a relatively new ART variance that addresses the "undeterminable recognition field size" problem of ART. Unlike a conventional ART network that mainly controls its learning activity with a vigilance threshold, ART-C's learning is mainly guided by an intuitive constraint on the maximal number of recognition categories in the network. The solution introduces an extra *constraint reset* mechanism to the ART network, which self-adjusts the vigilance threshold through an adaptive estimation of the input distribution in response to the constraint. The dynamically adjusted vigilance threshold in turn drives the learning activities to satisfy the user-defined

constraint. For a better understanding of this paper, the ART-C 2A learning paradigm, which is based on ART 2A [4], is reviewed with more details below.

### Parameters

The ART-C 2A dynamics are determined by the constraint  $C$  on the maximal number of recognition categories and the learning rate  $\eta \in [0, 1]$ .

### Network initialization

The category recognition layer is initialized with the null set  $\emptyset$  (i.e. contains no recognition category). The vigilance  $\rho$  is initialized as 1.0.

### Learning of each input representation

Learning of each input presentation follows the ART 2A learning paradigm [4].

### Constraint checking

Constraint checking is performed after the learning of each input representation by comparing the number of existing recognition categories  $N$  with the predefined constraint  $C$

$$\hbar = \begin{cases} 1 & \text{if } N > C \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

With  $\hbar = 0$ , the constraint is said to be satisfied, upon which the network carries on to learn the next input representation. Otherwise, *constraint reset* occurs, as described below.

### Constraint reset

Constraint reset re-organizes the recognition categories in the network towards the satisfaction of the constraint and adjusts the  $\rho$  value based on the current category distribution. The process is introduced as follows.

- 1) *Searching of the nearest category pair*: For each category pair  $(i, j)$  in the category recognition layer, their similarity is defined by the dot product of their corresponding weights  $\mathbf{w}_i$  and  $\mathbf{w}_j$  such that

$$T_{(i,j)} \equiv \mathbf{w}_i \cdot \mathbf{w}_j. \quad (2)$$

The *nearest neighbor* of each category  $i$ , indexed as  $J(i)$ , is the category that has the maximal similarity with  $i$ :

$$T_{(i,J(i))} = \max\{T_{(i,j)} : j = 1, \dots, N, j \neq i\}. \quad (3)$$

The *nearest neighbor similarity* of category  $i$ , marked as  $\tau(i)$  then refers to the similarity between category  $i$  and its nearest neighbor  $J(i)$ :

$$\tau(i) \equiv T_{(i,J(i))}. \quad (4)$$

The *nearest category pair*, indexed as  $(I, J)$ , is identified by the category  $I$  that has the minimal nearest neighbor similarity to its nearest neighbor  $J$ :

$$\tau(I) = T_{(I,J(I))} = \max\{T_{(i,J(i))} : i = 1, \dots, N\}. \quad (5)$$

- 2) *Adjustment of the vigilance*: The vigilance value  $\rho^{(new)}$  for subsequent learning is decreased according to:

$$\rho^{(new)} = \max\{\tau(i) : \text{all } i \text{ whose } \tau(i) < \rho^{(old)}\}. \quad (6)$$

- 3) *Merging of the nearest category pair*: Merging of the nearest category pair  $(I, J)$  is done by inserting a new

category  $L$  with the weight vector as the mean of these two categories:

$$\mathbf{w}_L = \mathfrak{R}(0.5\mathbf{w}_I + 0.5\mathbf{w}_J), \quad (7)$$

where  $\mathfrak{R}$  is the Euclidean normalization as given by

$$\mathfrak{R}\mathbf{x} \equiv \frac{\mathbf{x}}{\|\mathbf{x}\|}. \quad (8)$$

In addition, the categories  $I$  and  $J$  are deleted from the network after the creation of the new category.

#### IV. CASE STUDY: INTERPRETATION OF ART-C 2A CLUSTERS

We applied ART-C 2A to the clustering problem of the rat CNS data set [17]. The rat CNS data set contains 112 gene expressions on nine time points, covering the embryonic development phase (E11, E13, E15, E18, and E21, time in days), the postnatal development phase (P0, P7, and P14) and the adult phase (A). Prior study by Wen et al [17] on the data set using the FITCH software summarized five major waves of the gene expression patterns. With the exception of the *constant wave*, they have shown high correlation with the four major functional categories identified using biological domain knowledge, namely *Neuroglial Markers*, *Neurotransmitter Receptors*, *Peptide Signaling*, and *Diverse*.

We replicated Wen et al's experiment with ART-C 2A and compared the output of ART-C 2A with that of the FITCH software as reported in Wen's study. The small size and the relatively distinct expression patterns of this data set enabled us to validate ART-C 2A's clustering results via visual inspection. We adopted a standard set of parameters for ART-C 2A.  $C = 12$  is used for the purpose of analyzing the output clusters in satisfactory details. Figure 1 depicts the mean expression pattern of each cluster generated by ART-C 2A. The gene expressions grouped in each cluster are observed to have close similarity to each other (error bars corresponding to the deviations to the mean expressions are not plotted for a clearer illustration). Each pattern has showed a distinct group of gene expressions, identified by the variances of the expressions across all time points and the time point corresponding to the peak level. The ART-C 2A output is observed to have close relevancy with the output of the FITCH software as reported by Wen et al. The mapping of the clusters generated by ART-C 2A to the five major waves discovered by FITCH is summarized in Table I.

To further validate the ART-C 2A clusters, we investigated the correlation of genes in each cluster (Table II) with the major gene functional categories previously identified through human inspection (Table III) [17]. With the exception of cluster 1, which encodes a max of genes in *Peptide Signaling* and *Diverse* categories with relatively constant expressions (constant expressions normally are not of interests to biologists), the majority of the most clusters are dominated by genes of a single functional category. The best result is given by clusters 2 - 5, which clearly recognize the functional group *Neurotransmitter Receptors*. It is noted that, although cluster 6 and cluster 8 show relatively similar patterns (grouped into *wave1* in Wen et al's study), the majority of the genes in them

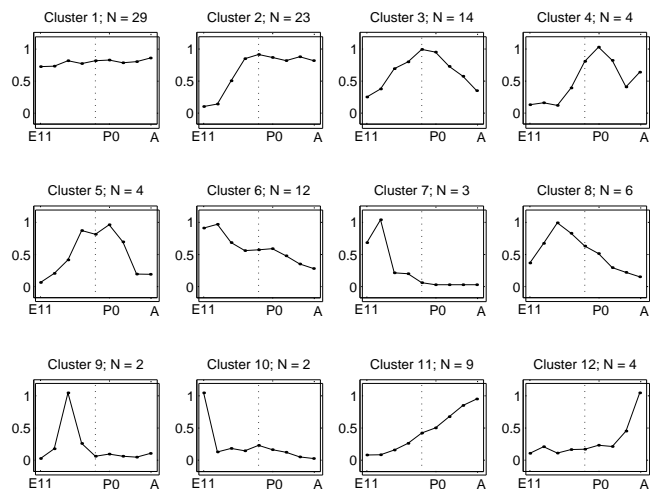


Fig. 1. The gene expression patterns of the rat CNS data set generated by ART-C 2A.  $N$  indicates the number of genes in each cluster.

TABLE I  
MAPPING OF THE GENE PATTERNS GENERATED BY ART-C 2A TO THE PATTERNS DISCOVERED BY FITCH.

| Cluster Pattern  |               | Interpretation  |
|------------------|---------------|---|
| ART-C 2A         | FITCH         |   |
| Cluster 1        | Constant Wave | Relatively constant levels across all time points.                                    |
| Cluster 2        | Wave 2        | Ascending levels during E phase and relatively constant levels during P and A phases. |
| Clusters 3 - 5   | Wave 3        | Peak level during late E and early P phases.  |
| Clusters 6 - 10  | Wave 1        | Peak level during early E phase.  |
| Clusters 11 - 12 | Wave 4        | Ascending levels across all time points.  |

actually corresponded to two different gene functional groups *Peptide Signaling* and *Neuroglial Markers*. This shows that ART-C 2A is capable of identifying subtle differences between the sets of two patterns.

It is interesting that although cluster 8 and cluster 11 present very different patterns, they actually corresponded to the same functional group *Neuroglial Markers*. In addition, several small clusters, especially cluster 7, 9, and 10 clearly identify a number of noises who did not follow the correlation between their functions and expressions in the main stream. This reflected the underlying complexity of the gene expression data.

#### V. COMPARATIVE EXPERIMENTS

##### A. Cluster Validity Measures

The *quantitative* assessment of clustering results did not receive much attention from the biologists until recently. Among the few known studies that quantitatively evaluate the gene clustering results, Yeung et al. [19] used generalized Jaccard and Hubert indices [11] to compare the performance of several clustering algorithms. The adjusted Rand index was

TABLE II  
LISTING OF GENES GROUPED IN THE CLUSTERS GENERATED BY ART-C  
2A.

| Cluster | Genes in the cluster   |
|---------|--|
| 1       | GAP43, GAT1, ODC, GRa1, GRb3, BDNF, CNTF, trkB, trkC, CNTFR, PTN, PDGFa, FGFR, TGFR, Ins2, IGF I, IGFR1, CRAF, IP3R1, IP3R2, cyclin A, H2AZ, cjun, TCP, actin, DD63.2, SOD, CCO1, CCO2 |
| 2       | MAP2, synaptophysin, neno, S100, pre-GAD67, GAD67, ACHE, GRa2, GRa3, GRa5, GRb1, GRg2, GRg3, mGluR3, mGluR5, mGluR7, NMDA1, NMDA2B, nAChRa7, mAChR2, 5HT1c, 5HT2, statin               |
| 3       | L1, NFL, GAD65, NOS, GRa4, mGluR8, NMDA2D, nAChRa3, nAChRa4, mAChR3, 5HT1b, EGFR, InsR, SC2  |
| 4       | mGluR4, NMDA2C, nAChRa2, EGF   |
| 5       | GRb2, mGluR2, mGluR6, 5HT3   |
| 6       | cellubrevin, nAChRa6, NT3, MK2, GDNF, PDGFR, IGF II, IGFR2, IP3R3, cyclin B, Brm, SC1  |
| 7       | nAChRd, PDGFb, SC6   |
| 8       | nestin, G67I80/86, G67I86, TH, nAChRa5, SC7  |
| 9       | nAChRe, trk  |
| 10      | keratin, Ins1  |
| 11      | NFH, GFAP, MOG, ChAT, GRg1, mAChR4, bFGF, aFGF, cfos   |
| 12      | NFM, mGluR1, NMDA2A, NGF   |

TABLE III  
CORRELATION BETWEEN THE GENE CLUSTERS DISCOVERED BY ART-C  
2A AND THE FUNCTIONAL GENE CATEGORIES IDENTIFIED THROUGH  
HUMAN INSPECTION.  $N$  IS THE TOTAL NUMBER OF GENES IN EACH  
CLUSTER.  $NM$ ,  $NR$ ,  $PS$ ,  $DV$  ARE THE NUMBERS OF GENES IN  
FUNCTIONAL CATEGORY *Neuroglial Markers*, *Neurotransmitter Receptors*,  
*Peptide Signaling*, AND *Diverse* RESPECTIVELY. BOLDFACE NUMBERS  
IDENTIFY THE DOMINATE FUNCTIONAL CATEGORY IN EACH CLUSTER.

| Cluster | $N$ | Gene Class Distribution |           |           |           |
|---------|-----|-------------------------|-----------|-----------|-----------|
|         |     | $NM$                    | $NR$      | $PS$      | $DV$      |
| 1       | 29  | 3                       | 2         | <b>12</b> | <b>12</b> |
| 2       | 23  | 7                       | <b>15</b> | 0         | 1         |
| 3       | 14  | 4                       | <b>7</b>  | 2         | 1         |
| 4       | 4   | 0                       | <b>3</b>  | 1         | 0         |
| 5       | 4   | 0                       | <b>4</b>  | 0         | 0         |
| 6       | 12  | 1                       | 1         | <b>6</b>  | 4         |
| 7       | 3   | 0                       | 1         | 1         | 1         |
| 8       | 6   | <b>4</b>                | 1         | 0         | 1         |
| 9       | 2   | 0                       | 1         | 1         | 0         |
| 10      | 2   | 1                       | 0         | 1         | 0         |
| 11      | 9   | <b>4</b>                | 2         | 2         | 1         |
| 12      | 4   | 1                       | <b>2</b>  | 1         | 0         |
| Total   | 112 | 25                      | 39        | 27        | 21        |

used in Yeung et al's recent work to evaluate their model-based clustering algorithm [18]. These evaluation measures are all based on the so-called *external* criteria, i.e. they require a *known optimal partition* of the data set as the reference result. Such an optimal partition however may not be available without extensive human knowledge on the problem domain. Therefore in our studies, we assess the clustering results

through the distribution of the output clusters directly.

Since the target of clustering is to re-organize the input samples such that data points in the same cluster are more similar to each other than to points in a different cluster, it is natural to evaluate the intra-cluster homogeneity and the inter-cluster separation of the clustering output in a global fashion. We used two quantitative measures, namely *cluster compactness* and *cluster separation* for this purpose. The definitions of these two measures are summarized below. More discussions on these measures can be found in [9].

#### Cluster compactness

The cluster compactness measure is based on the generalized definition of the *variance* of a data set given by

$$v(\mathbf{X}) = \sqrt{\frac{1}{N} \sum_{i=1}^N d^2(\mathbf{x}_i, \bar{\mathbf{x}})} \quad (9)$$

where  $d(\mathbf{x}_i, \mathbf{x}_j)$  is a distance metric between two vectors  $\mathbf{x}_i$  and  $\mathbf{x}_j$ ,  $N$  is the number of members in  $\mathbf{X}$ , and  $\bar{\mathbf{x}} = \frac{1}{N} \sum_i \mathbf{x}_i$  is the mean of  $\mathbf{X}$ . The cluster compactness for the output clusters  $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_C$  generated by a system is then defined as

$$Cmp = \frac{1}{C} \sum_i \frac{v(\mathbf{c}_i)}{v(\mathbf{X})} \quad (10)$$

where  $C$  is the number of clusters generated on the data set  $\mathbf{X}$ ,  $v(\mathbf{c}_i)$  is the variance of the cluster  $\mathbf{c}_i$ , and  $v(\mathbf{X})$  is the variance of the data set  $\mathbf{X}$ .

#### Cluster separation

The cluster separation of a clustering system's output is defined by

$$Sep = \frac{1}{C(C-1)} \sum_{i=1}^C \sum_{j=1, j \neq i}^C \exp\left(-\frac{d^2(\mathbf{x}_{c_i}, \mathbf{x}_{c_j})}{2\sigma^2}\right) \quad (11)$$

where  $\sigma$  is a Gaussian constant,  $C$  is the number of clusters,  $\mathbf{x}_{c_i}$  is the centroid of the cluster  $\mathbf{c}_i$ , and  $d(\mathbf{x}_{c_i}, \mathbf{x}_{c_j})$  is the distance between the centroid of  $\mathbf{c}_i$  and the centroid of  $\mathbf{c}_j$ .

It is understandable that for both measures, a smaller value indicates a better output quality.

#### B. Evaluation Paradigm

All the four clustering algorithms used a standard set of parameters. The reference clusters of K-Means and SOM were initialized with random vectors that slightly vary from the mean vector of the input set. Both K-Means and SOM utilized Euclidean distances. On SOM, different neighborhood degrees and various neighborhood decaying methods were tested. We obtained relatively faster convergence and marginally better quality when the neighborhood degree was zero. The results with such parameters are reported in this paper. Readers shall note that, as discussed in [6], with a zero neighborhood degree, SOM becomes equivalent to an online version of K-Means.

We utilized the Euclidean distance for the evaluation of cluster compactness ( $Cmp$ ) and cluster separation ( $Sep$ ).  $2\sigma^2 = 1.0$  as in Equation 11 was used to simplify our evaluation. Multiple runs of experiments, each using randomly reshuffled sequence of input, are conducted in order to obtain a

statistically valid comparison.  $t$  statistics was used to evaluate the statistical significance of our comparison observation when appropriate.

Note that a valid comparison of clustering systems using the two evaluation measures requires them to output the same (or at least comparable enough) number of clusters. In order to control the number of ART 2A output clusters, we manually tried various  $\rho$  values on one random input sequence, then used the  $\rho$  value which produced the pre-specified number of output clusters on this input sequence in the remaining runs. While the actual number of ART 2A output clusters may slightly vary from the pre-specified number using the same  $\rho$  value on different input sequences, we found the variance was within an acceptable level.

### C. Data Sets

Our first batch of the experiments compared the performance of these four algorithms on two gene expression data sets, namely the yeast cell cycle data set (YEAST) and the human hematopoietic differentiation data set with features under mixed conditions (HL60\_U937\_NB4\_Jurkat).

The yeast cell cycle data set (YEAST)<sup>1</sup> consists of 6,601 gene expression data in 17 conditions. The 17 conditions are evenly divided into two panels, each one corresponding to a cell cycle, with the 9th condition as the intermediate point between the two cell cycles. Following a common procedure [15], we employed a variance filter to select genes with significant changes over conditions. The variance filter adjusted observation values into units range  $[min, max]$  and eliminated genes which did not show a relative change of  $x$  times and an absolute change of  $y$  units across all conditions. Using the parameter settings  $min = 20, max = 20,000, x = 3$ , and  $y = 150$ , a sub-set of 1,109 genes was generated for use in our experiments. Expression levels were first normalized using the standard normal distribution with a mean of 0 and a standard variance of 1 within each panel. The intermediate 9th condition was excluded from our experiments for the ease of normalization. The remaining 16-dimensional vectors were further Euclidean normalized for our benchmark.

The human hematopoietic differentiation data set (HL60\_U937\_NB4\_Jurkat)<sup>2</sup>, which consisted of 7,229 gene expression data in 17 conditions, was preprocessed with the same variance filter. Using settings  $min = 20, max = 20,000, x = 3$ , and  $y = 100$ , 1,423 genes passed through the variance filter. Expression levels were first normalized using the standard normal distribution over all conditions and then Euclidean normalized.

### D. Results and Discussions

Both the two gene expression data sets are small scale, have a small number of features, and are densely distributed. Prior studies are capable of identifying a few number of expression patterns on these data sets only. Therefore on each data set, we set the target number of the output clusters to be relatively small. Table IV reports the four algorithms' cluster validity

measures, together with the CPU time costs, when the target number of output clusters is 10 and 20 respectively.

TABLE IV

EXPERIMENTAL RESULTS FOR ART-C 2A, ART 2A, SOM, AND K-MEANS ON THE YEAST AND THE HL60\_U937\_NB4\_JURKAT DATA SETS, WHEN THE NUMBER OF CLUSTERS  $C$  WAS SET TO 10 AND 20.  $I$ ,  $T$ ,  $Cmp$  AND  $Sep$  INDICATE THE NUMBER OF LEARNING ITERATIONS, THE COST OF TRAINING TIME (IN  $ms$ ), *cluster compactness* AND *cluster separation* RESPECTIVELY. ALL VALUES ARE SHOWN WITH THE MEAN AND THE STANDARD DEVIATION OVER TEN RUNS.

| YEAST, $C = 10$                |          |             |                      |                      |
|--------------------------------|----------|-------------|----------------------|----------------------|
| Method                         | $I$      | $T$ (ms)    | $Cmp$                | $Sep$                |
| ART-C 2A                       | 2.6±0.5  | 12.0±4.2    | <b>0.7469±0.0301</b> | 0.1424±0.0057        |
| ART 2A                         | 2.0±0.0  | 10.0±0.0    | 0.7514±0.0135        | <b>0.1408±0.0053</b> |
| SOM                            | 7.9±1.4  | 52.1±10.3   | 0.7670±0.0071        | 0.1579±0.0044        |
| K-Means                        | 13.0±2.2 | 72.3±13.9   | 0.7583±0.0065        | 0.1639±0.0069        |
| YEAST, $C = 20$                |          |             |                      |                      |
| Method                         | $I$      | $T$ (ms)    | $Cmp$                | $Sep$                |
| ART-C 2A                       | 3.0±0.0  | 19.0±8.9    | 0.7157±0.0107        | <b>0.1587±0.0035</b> |
| ART 2A                         | 3.0±0.0  | 18.0±4.2    | 0.7167±0.0123        | 0.1607±0.0024        |
| SOM                            | 14.3±0.9 | 196.5±14.6  | 0.6887±0.0250        | 0.1836±0.0048        |
| K-Means                        | 14.8±2.6 | 216.8±37.4  | <b>0.6861±0.0136</b> | 0.1858±0.0059        |
| HL60_U937_NB4_Jurkat, $C = 10$ |          |             |                      |                      |
| Method                         | $I$      | $T$ (ms)    | $Cmp$                | $Sep$                |
| ART-C 2A                       | 2.6±0.5  | 11.0±3.2    | 0.7262±0.0316        | 0.1525±0.0073        |
| ART 2A                         | 2.0±0.0  | 10.0±0.0    | 0.7209±0.0161        | <b>0.1462±0.0060</b> |
| SOM                            | 10.3±1.8 | 84.8±16.5   | <b>0.6983±0.0049</b> | 0.1907±0.0033        |
| K-Means                        | 13.5±3.2 | 119.7±68.4  | 0.6985±0.0201        | 0.1963±0.0101        |
| HL60_U937_NB4_Jurkat, $C = 20$ |          |             |                      |                      |
| Method                         | $I$      | $T$ (ms)    | $Cmp$                | $Sep$                |
| ART-C 2A                       | 3.0±0.0  | 23.0±4.8    | 0.6543±0.0192        | <b>0.1675±0.0031</b> |
| ART 2A                         | 3.0±0.0  | 20.0±4.7    | 0.6632±0.0197        | 0.1711±0.0048        |
| SOM                            | 17.0±2.6 | 279.7±45.6  | 0.6508±0.0194        | 0.2181±0.0090        |
| K-Means                        | 15.1±2.5 | 232.9±120.8 | <b>0.6348±0.0123</b> | 0.2174±0.0089        |

In all the four batches of experiments, the cluster validity measures produced by ART-C 2A, in terms of both cluster compactness and cluster separation, were very similar to those of ART 2A. Specifically,  $t$ -test did not suggest any significant difference between our observations on each evaluation measure. It is interesting that the performances of SOM and K-Means were rather similar to each other as well. This is surprising as some prior studies reported that SOM was notably worse than K-Means [6]. We should highlight that the good performance of SOM may be due to our use of zero neighborhood in the controlled experiments, which was extensively discussed in [6]. It is also noted that the initial reference clusters affect the outputs of both K-Means and SOM significantly. In our experiments, we used the same strategy to initialize the reference clusters for K-Means and SOM. This also partly explains the similarities in their outputs.

In terms of cluster compactness, the validity measures of these four algorithms did not show significant differences in experiments on YEAST with  $C = 10$  and HL60\_U937\_NB4\_Jurkat with  $C = 20$ , although K-Means produced slightly lower scores than the remaining trio on HL60\_U937\_NB4\_Jurkat with  $C = 20$ . However, for YEAST with  $C = 20$  and HL60\_U937\_NB4\_Jurkat with  $C = 10$ , both SOM and K-Means produced significantly lower scores than those of ART-C 2A and ART 2A. In general, across these four batches of observations, the cluster compactness scores of K-Means were slightly lower than those of SOM and significantly lower than those of ART-C 2A and ART 2A in some of our experiments.

<sup>1</sup><http://genomics.stanford.edu>.

<sup>2</sup><http://www-genome.wi.mit.edu/cgi-bin/cancer/datasets.cgi>.

This sounds reasonable, as K-Means, which uses the global distribution of the data set to reduce the overall error, should tend to outperform competitive learning systems that look at the input patterns one at a time. In addition, ART-C 2A and ART 2A tend to encode noises and produce clusters of uneven sizes. This in turn decreases the compactness of the large output clusters.

In terms of cluster separation, the validity measures of both ART-C 2A and ART 2A were significantly lower than those of SOM and K-Means. It may be that, whereas SOM and K-Means modify existing, randomly initialized cluster prototypes to encode new samples, ART-C 2A and ART 2A adaptively inserts recognition categories to encode new input samples that are significantly distinct from existing prototypes. The distinct reference clusters in turn make the output clusters to be more dissimilar to each other. This unique neuron initialization mechanism appears to be effective in representing diverse data patterns in the input set.

To sum up, the output quality of all the three self-organizing networks are quite satisfactory, comparable to that of K-Means. ART-C 2A and ART 2A tend to produce clusters that are more separated from each other, with slightly compromised cluster compactness.

In terms of efficiency, ART-C 2A incurred slightly more computational cost than ART 2A. However, the difference was not significant. More importantly, both ART-C 2A and ART 2A showed a significantly higher efficiency than SOM and K-Means. In all experiments, SOM and K-Means used three to five times more iterations and CPU times than those of ART-C 2A and ART 2A.

In addition to the above observations, it is necessary to highlight that ART-C 2A has a higher usability than ART 2A in our opinion. Given an unknown gene expression collection, it is desirable that the number of output clusters generated by a system is within a human controllable scale (and preferably fixed) for the ease of human inspection. Specifically, ART-C 2A removes the trial-and-try process of ART 2A in estimating a proper vigilance parameter for this purpose.

## VI. CONCLUSIONS AND FUTURE WORK

We have presented a novel neural architecture known as ART-C (Adaptive Resonance Theory under Constraints) for efficient clustering of gene expression data. Through a serial of experiments on three real-life data sets, namely the rat CNS data set, the yeast cell cycle data set, and the human hematopoietic differentiation data set, we compared the performance of ART-C with those of K-Means, SOM, and conventional ART. Our studies and comparative experiments on the three data sets indicate that:

- Through clustering of gene expressions, ART-C is capable of identifying inherent knowledge on functional groupings with a satisfactory level of quality. In addition, ART-C showed a high capability of identifying subtle differences between two pattern sets.
- Compared with SOM and K-Means, ART 2A and ART-C 2A showed a notably higher clustering efficiency. This is a crucial advantage for mining large scale gene expression data.

- Compared with ART 2A, ART-C 2A could have a greater ease of use for gene clustering, due to its capability of allowing the user to directly control the number of output cluster.

Our studies so far on several well-known data sets have produced satisfactory results, with respect to prior biological work. It would be interesting for biologists to apply these algorithms to *new* gene expression collections and explore what *new knowledge* can be discovered.

## REFERENCES

- [1] Y. Barash and N. Friedman. Context-specific bayesian clustering for gene expression data. In *The Fifth Annual International Conference on Computational Molecular Biology (RECOMB)*, pages 12–21, 2001.
- [2] A. Ben-Dor, R. Shamir, and Z. Yakhini. Clustering gene expression patterns. *Journal of Computational Biology*, 6(3/4):281–297, 1999.
- [3] G. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image processing*, 34:54–115, 1987.
- [4] G. Carpenter, S. Grossberg, and D. Rosen. ART 2-A: An adaptive resonance algorithm for rapid category learning and recognition. *Neural Networks*, 4:493–504, 1991.
- [5] M. Eisen, P. Spellman, D. Botstein, and P. Brown. Cluster analysis and display of genome-wide expression patterns. In *Proceedings of National Academy of Science USA*, volume 95, pages 14863–14867, 1998.
- [6] A. Flexer. Limitations of self-organizing maps for vector quantization and multidimensional scaling. *Advances in Neural Information Processing Systems 9.*, pages 445–451, 1997.
- [7] S. Grossberg. Adaptive pattern classification and universal recoding. I. parallel development and coding of neural feature detector. *Biological Cybernetics*, 23:121–134, 1976.
- [8] J. He, A. Tan, and C. Tan. ART-C: A neural architecture for self-organization under constraints. In *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, pages 2550–2555, 2002.
- [9] J. He, A. Tan, C. Tan, and S. Sung. On quantitative evaluation of clustering systems. In W. Wu and H. Xiong, editors, *Information Retrieval and Clustering*. Kluwer Academic Publishers, 2002. in press.
- [10] J. Herrero, A. Valencia, and J. Dopazo. A hierarchical unsupervised growing neural network for clustering gene expression patterns. *Bioinformatics*, 17(2):126–136, 2001.
- [11] A. Jain and R. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Upper Saddle River, NJ, 1988.
- [12] T. Kohonen. *Self-Organization and Associative Memory*. Springer-Verlag, Berlin, second edition, 1997.
- [13] N. Luscombe, D. Greenbaum, and M. Gerstein. What is bioinformatics? An introduction and overview. *Yearbook of Medical Informatics*, pages 83–100, 2001.
- [14] G. Michaels, D. Carr, M. Askenazi, S. Fuhrman, X. Wen, and R. Somogyi. Cluster analysis and data visualization of large-scale gene expression data. In *Pacific Symposium on Biocomputing*, 3, pages 42–53, 1998.
- [15] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. Lander, and T. Golub. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. In *Proceedings of the National Academy of Science*, volume 96, pages 2907–2912, 1999.
- [16] J. Tou and R. Gonzalez. *Pattern Recognition Principles*. Addison-Wesley, Massachusetts, 1974.
- [17] X. Wen, S. Fuhrman, G. Michaels, D. Carr, S. Smith, G. Barker, and R. Somogyi. Large-scale temporal gene expression mapping of central nervous system development. In *Proceedings of the National Academy of Science*, pages 334–339, 1998.
- [18] K. Yeung, C. Fraley, A. Raftery, and W. Ruzzo. Model-based clustering and data transformations for gene expression data. *Bioinformatics*, 17(10):977–987, 2001.
- [19] K. Yeung, D. Haynor, and W. Ruzzo. Validating clustering for gene expression data. Technical Report UW-CSE-00-01-01, Department of Computer Science and Engineering, University of Washington, January 2000.