

Automatic Teaching–Learning-Based Optimization: A Novel Clustering Method for Gene Functional Enrichments

Ramachandra Rao Kurada, K. Karteeka Pavan and Allam Appa Rao

Abstract Multi-objective optimization emerged as a significant research area in engineering studies because most of the real-world problems require optimization with a group of objectives. The most recently developed meta-heuristics called the teaching–learning-based optimization (TLBO) and its variant algorithms belongs to this category. This paper provokes the importance of hybrid methodology by illuminating this meta-heuristic over microarray datasets to attain functional enrichments of genes in the biological process. This paper persuades a novel automatic clustering algorithm (AutoTLBO) with a credible prospect by coalescing automatic assignment of k value in partitioned clustering algorithms and cluster validations into TLBO. The objectives of the algorithm were thoroughly tested over microarray datasets. The investigation results that endorse AutoTLBO were impeccable in obtaining optimal number of clusters, co-expressed cluster profiles, and gene patterns. The work was further extended by inputting the AutoTLBO algorithm outcomes into benchmarked bioinformatics tools to attain optimal gene functional enrichment scores. The concessions from these tools indicate excellent implications and significant results, justifying that the outcomes of AutoTLBO were incredible. Thus, both these rendezvous investigations give a lasting impression that AutoTLBO arises as an impending colonizer in this hybrid approach.

R.R. Kurada (✉)

Department of Computer Science and Engineering, Shri Vishnu Engineering
College for Women, Bhimavaram, India
e-mail: ramachandrarao.kurada@gmail.com

K. Karteeka Pavan

Department of Information Technology, RVR & JC College of Engineering,
Guntur, India
e-mail: kanadamkarteeka@gmail.com

A.A. Rao

CRRao AIMSCS, UoH, Hyderabad, India
e-mail: apparaoallam@gmail.com

© The Author(s) 2015

N.B. Muppalaneni and V.K. Gunjan (eds.), *Computational Intelligence
Techniques for Comparative Genomics*, Forensic and Medical Bioinformatics,
DOI 10.1007/978-981-287-338-5_2

Keywords Automatic clustering • Teaching–learning-based optimization • Gene functional enrichments • Cluster validity indices

1 Introduction

Evolutionary algorithms (EA) are generic meta-heuristic optimization algorithms that use techniques inspired by nature's evolutionary processes. EA maintains a whole set of solutions that are optimized at the same time instead of a one single solution. The inherent randomness of the emulated biological processes enables them to provide good approximate solutions nevertheless. The recently emerged nature-inspired multi-objective meta-heuristic optimization algorithms teaching–learning-based optimization (TLBO) [1, 2] and its variations Elitist TLBO [3, 4] belong to this category. Both these algorithms aim to find global solutions for real-world problem with less computational effort and high reliability. The principle idea behind TLBO is the simulation of teaching–learning process of a traditional classroom in to algorithmic representation with two phases called teaching and learning. Elitist TLBO was pioneered with a major modification to eliminate the duplicate solutions in learning phase.

Clustering is the subject of active research in several fields such as statistics, pattern recognition, machine learning, data mining, and bioinformatics. The purpose of clustering is to determine the intrinsic grouping in a set of unlabeled data, where the objects in each group are indistinguishable under some criterion of similarity. Clustering is used to partition a dataset into groups, so that the data elements within a cluster are more similar to each other than data elements in different clusters. Automatic clustering addresses the challenge of determination the appropriate number of clusters or partitions mechanically.

Most of the existing clustering techniques, based on EA, accept the number of classes (k) as an input instead of determining the same on the iteration. Nevertheless, in many practical situations, the appropriate number of groups in a previously unhandled dataset may be unknown or impossible to determine even approximately. To avoid the algorithm struck in such blockage, automatic assignment of (k) value by the algorithm in each run was made tangible in this work. These automatic clusters are again endorsed with cluster validity indices (CVIs), which combine compactness and separability for assessing the quality of clusters. Cluster validity criteria are of three types external, internal, and relative. External indexes require a priori data for the purposes of evaluating the results of a clustering algorithm, whereas internal indexes do not. Internal indexes evaluate the results of a clustering algorithm using information that involves the vectors of the datasets themselves. The relative index evaluates the results by comparing the current cluster structures with other clustering schemes. The CVIs that are used in this work are rand index (RI) [5], advanced rand index (ARI) [5], Hubert index (HI) [6],

silhouettes (SIL) [7], Davies and Bouldin (DB) [8], and Chou (CS) [9] measures, primarily finds the best partitioning in the underlying data.

This paper impersonate k -means clustering algorithm, procedures for automatic clustering, CVIs, visualization, and elitism techniques into TLBO. The objective of the novel AutoTLBO algorithm was to cluster the *Saccharomyces cerevisiae* categorized microarray datasets, and the expected multiple outcome was to attain optimal number of automatic clusters, mean values of CVIs, dendrograms and cluster profiles of co-expressed genes. These outcomes are assumed as summative assignment-I and is shown as Experiment 1 in Sect. 5. The obtained cluster profiles of AutoTLBO are used as inputs into Bioinformatics tools FatiGO for first opinion and database for annotation, visualization, and integrated discovery (DAVID) for second opinion. This verification procedure is named as summative assignment-II, primarily used to re-validate the results given by this novel AutoTLBO. Figure 1 unveils a broad road map of the proposed work in this paper. The input is fed into the tool in such a manner that the first list holds the gene-IDs of one of the cluster and the other list holds the gene-IDs of all the remaining clusters generated by this novel AutoTLBO algorithm. Two-stage preprocessing is imposed on the lists by applying statistical techniques such as Fisher exact test and duplicate elimination. Finally, these clean lists are used in gene ontology (GO) biological process to find the significant terms, term annotations % per list, p -value, FDRs, enrichment scores, etc. This entire set of test results of both the tools are publicized as Experiment 2 in Sect. 5. The re-validate techniques adopted in the tools manifest a positive sign that the novel AutoTLBO is power-packed in obtaining optimal number of automatic clusters and discrete gene cluster profiles. This silver lining absolutely ratify that the novel algorithm proposed in this work can be used for attaining gene functional enrichments.

The rest of the paper is formed as follows. Section 2 exposes a basic background to the theme concepts used by other researchers, TLBO, and its variations.

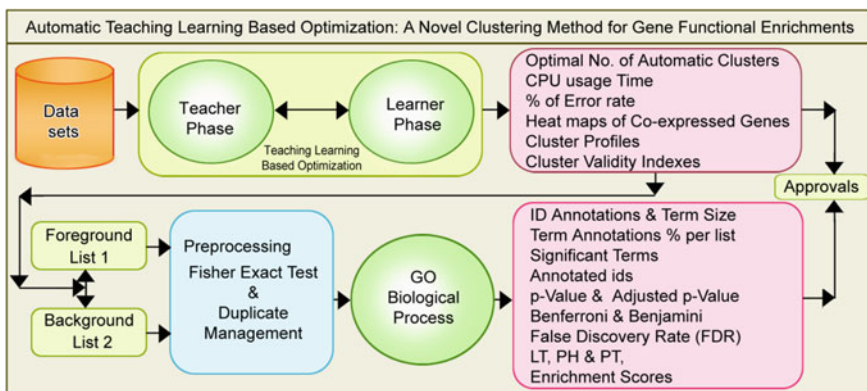


Fig. 1 Road map of AutoTLBO: a novel clustering method for gene functional enrichments

Section 3 summarizes the general framework of the TLBO algorithm. Section 4 presents the novel algorithm formulation. Results of a comparative study are presented in Sect. 5. Finally, conclusions are provided in Sect. 6.

2 Literature Survey

One of the most recent advancements in nature-inspired population-based meta-heuristics was the TLBO algorithm. This algorithm was initially proposed by Rao et al. in 2011 [1]. The worthiness of TLBO was successfully proved when it was applied over constraint real-time engineering optimization problems. In 2012, the same set of authors Rao et al. [2] have once again proven the efficiency on continuous non-linear large-scale problems with respect to the criterion best solution, average solution, convergence rate, and computational effort. A way to extend the TLBO algorithm to solve specific engineering optimization problems was also shown to the novel researchers by Rao et al. Hence, the authors of this paper embedded the concept of automatic clustering into TLBO and proved its effectiveness in clustering microarray datasets.

The methodology of this algorithm rolls between two entities teacher and learner. The outcome of the class learners is prejudiced by the teacher. Also, the superiority of the teacher is assessed by the learner's concert in requisites of marks and rating. The other significant factor that persuades the teacher's quality was the learner's progress among themselves by communication in the class. The same orientation was carry forwarded in this work to automatically cluster the genes stored in the form of microarray datasets.

The concept of elitist was introduced in to TLBO, and this modified version was released by Rao and Patel in 2012 [3] to solve constrained problems. The objective of this embedment was to eliminate the duplicate solutions and to get efficiency of algorithm when common variables such as population size and number of generations were used in the commencement of the algorithm. Again in 2013, Rao and Patel [4] pioneered elitism and common variable in TLBO to solve unconstrained engineering problems with different benchmark functions. The authors were successful in both [3, 4] to produce valuable results and prove the efficiency of TLBO with elitism. The inspiration for this present work was from the aforementioned articles [1–4], and the hopeful extensions to this work were implementing the new methodologies stated in articles [10, 11].

The enhancement incorporated in to TLBO by Rao and Patel in 2013 [10] was to exploit the capabilities of multiple teachers into classrooms (learners), adaptive teaching factor, tutorial training and self motivated learning. All these characteristics were thoroughly assessed in solving unconstrained multi-dimensional, linear, and non-linear problems. The most up-to-date work of Rao and Waghmare was in 2014 [11] which evaluated and produced efficient results by introducing multi-objective optimization with multiple trade-off in to TLBO over a set of the constraint and unconstrained functions.

TLBO relevance to cluster analysis was shown in 2012 by Amiri [12]; this study was accomplished by testing on quite a few numbers of datasets. Automatic clustering in multi-objective optimization framework using differential evolution was shown by Suresh et al. in 2011 [13]. The experimental result over different datasets proves the variations of DE that are desired for doing automatic clustering. Automatic clustering using genetic algorithms and generating optimality with Pareto front is well demonstrated by the same set of authors Suresh et al. in 2009 [14]. Cluster evaluation, ranking, and validation using CVIs are effectively shown by Liu et al. in 2005 [15]. The conceptualization toward fitting in automatic clustering into TLBO in the paper was from [13, 14].

Satpathy et al. in 2013 [16] brought an improved version of TLBO by using orthogonal design. This change was proved as a statistically effect method to generate an optimal offspring in EA. In the recent past, automatic clustering in TLBO was shown by Naik et al. in 2012 [17] using fuzzy c means. The results were well demonstrated over artificial and real datasets. In 2014, Murthy et al. [18] used automatic clustering in TLBO to find optimal number of clusters and shown potential results proving the efficiency of algorithm.

The proposal toward using automatic clustering in TLBO over microarray datasets was from the article published by Suresh et al. in 2009 [19] and Pavan et al. in 2011 [20]. Both these articles use a test suite to compare results over the gene datasets. The acquired optimal numbers of clusters are verified by using CVIs.

3 TLBO

TLBO algorithm is a teaching–learning methodology-motivated population-based algorithm, proposed by Rao et al. [1–4, 10, 11] which focused around the impact of a teacher on the after effect of learners in a class. In this optimization algorithm, the faction of learners are assumed as population and diverged configuration of variables are treated as distinctive subjects accessible to the learners, and their result is comparable to the fitness estimation value of this optimization issue. In the whole population, the best solution is treated as the teacher.

Teacher phase: It is included as the first segment of TLBO, where learners gain knowledge from the teacher. In this phase, the teacher attempts to increase the mean value of the class room from any value mean_1 to his or her echelon I_A . But sensibly it is not promising and a teacher can move the mean of the class room mean_1 to any other value mean_2 which is healthier than mean_1 depending on his or her competence. Considered mean_j be the mean and I_i be the teacher at any iteration i . Now, teacher I_i will try to improve the existing mean mean_j toward it so the new mean will be I_i designated as mean_{new} , and the difference between the existing mean and new mean is given as

$$\text{diverged_mean}_i = r_i (\text{mean}_{\text{new}} - T_F * \text{mean}_j) \quad (1)$$

where T_F is the teaching factor that fixes the value of mean to be changed, and r_i is the random number in the range $[0, 1]$, that is used to support the teaching factor. Value of T_F can either 1 or 2 which is an interrogative step, which is determined randomly with equivalent probability as:

$$T_F = \text{round}[1 + \text{rand}(0, 1)\{2 - 1\}] \quad (2)$$

The teaching factor is produced arbitrarily in TLBO within the scope of 1–2, in which 1 compares to no increase in the learning level and 2 relates to inclusive exchange of knowledge, and intermediate values indicate the exchange of knowledge. The shifting level of knowledge can be any depending on the learner competence.

Based on diverged_mean , the existing solution is updated according to the following expression:

$$\alpha_{\text{new},i} = \alpha_{\text{old},i} + \text{diverged_mean}_i \quad (3)$$

Learner phase: It is included as the second segment of the algorithm, where learners improve their knowledge by communication among themselves. A learner adapts new things if the other learner has more knowledge than him. Precisely, the learning trend of this phase is articulated as follows:

At any iteration i , consider two distinct learners α_i and α_j where $i \neq j$.

$$\alpha_{\text{new},i} = \alpha_{\text{old},i} + r_i (\alpha_i - \alpha_j) \quad \text{if } f(\alpha_i) < f(\alpha_j) \quad (4)$$

$$\alpha_{\text{new},i} = \alpha_{\text{old},i} + r_i (\alpha_j - \alpha_i) \quad \text{if } f(\alpha_j) < f(\alpha_i) \quad (5)$$

3.1 Elitist TLBO Procedure

Step 1: Initialization Stage

Initialize the population (learners), design variables (numbers of subjects offered to the learners) with random generation, threshold values, and termination criterion.

Step 2: Elitist Teaching Phase

Select the best learners of each subject as a teacher for that subject and calculate mean result of learners in each subject.

- (a) Keep the elite solution
- (b) Calculate the mean of each design variable
- (c) Select the best solution
- (d) Calculate the diverged_mean and modify the solutions based on best solution

- Step 3: *Elitist Teaching Phase—Update procedure amid with duplicate elimination*
 Evaluate the difference between current mean result and best mean result according to Eq. (1) by utilizing the teaching factor T_F
- (a) If the new solution is better than the existing solution, then accept or else keep the previous solution
 - (b) Select the solutions randomly and modify them by comparing with each other
 - (c) Modify duplicate solution via mutation on randomly selected dimensions of duplicate solutions before executing the next generation
- Step 4: *Elitist Learners Phase*
 Update the learner's knowledge with the help of teacher's knowledge according to Eq. (3)
- (a) If the new solution is better than the existing solution, then accept or else keep the previous solution
 - (b) Replace worst solution with elite solution
- Step 5: *Elitist Learners Phase—Update procedure amid with duplicate elimination*
 Update the learner's knowledge by utilizing the knowledge of some other learners according to Eqs. (4) and (5).
- (a) Modify duplicate solution via mutation on randomly selected dimensions of duplicate solutions before executing the next generation
- Step 6: *Stoppage Criterion*
 Repeat the procedure from Step 2 to Step 5 till the termination criterion is met.
- (a) If termination criterion is fulfilled, then we get the final value of the solution or else repeat from Step 2 to Step 5.

4 Automatic Clustering Using Elitist TLBO (AutoTLBO)

The proposed automatic clustering using Elitist TLBO algorithm (AutoTLBO) follows a novel integrated approach by assimilation of elitism and cluster evaluation implanted into TLBO algorithm. Elitism is a mechanism used in this algorithm to preserve the best individuals from generation to generation. By this way, the algorithm never loses the best individuals found during the optimization process. In this algorithm, replacing the worst solutions with elite solutions is done at the end of learner phase. In the present work, duplicate solutions are modified by mutation on randomly selected dimensions of the duplicate solutions before executing the

next generation. At the same time, the solutions are updated both in teacher phase and learner phase. The cluster evaluation procedure adopted in this algorithm is used to appraise the generated cluster with CVIs. The internal and external CVIs such as RI, ARI, HI, SIL, DB, and CS are used in this algorithm as an objective function to evaluate the cluster engendered.

4.1 AutoTLBO Algorithm

Step 1: *Initialization Phase*

Initialize each learner to contain Max k number of selected cluster centers and Max k (randomly chosen) activation thresholds in $[0, 1]$. Let α is a given dataset with n elements. The population α is initialized randomly. The dataset is generated with n rows and d columns using the following equation.

$$\alpha_{i,j}(0) = \alpha_j^{\min} + \text{rand}(1) * (\alpha_j^{\max} - \alpha_j^{\min}) \quad (6)$$

where $\alpha_{i,j}$ creates a population of learners or individuals. The i th learner of the population α at current generation t with d subjects is as follows:

$$\alpha_i(t) = [\alpha_{i,1}(t), \alpha_{i,2}(t), \dots, \alpha_{i,d}(t)] \quad (7)$$

Step 2: *Teaching Phase*

Find the active cluster centers with value greater than 0.5, in each learner, and keep it as a elite solution as mentioned in Eq. 1

Step 3: *Teaching Phase—Update procedure amid with duplicate elimination*

For $t = 1$ to t_{\max} do

- (a) For each data vector α_p , calculate its distance from all active cluster centers using Euclidean distance or Euclidean metric
- (b) Assign α_p to nearby cluster using simple k -means algorithm
- (c) Modify duplicate solution via mutation on randomly selected dimensions of duplicate solutions before executing the next generation as described in Eqs. 2 and 3.

Step 4: *Learner Phase*

Evaluation of Clusters engendered with CVIs

- (a) Evaluate each learner quality and find teacher, the best learner using RI or other indices
- (b) Replace worst solution with elite solution

Step 5: *Learners Phase—Update procedure amid with duplicate elimination*

- (a) Update the learners according to the TLBO algorithm described in Eqs. 4 and 5
- (b) Modify duplicate solution via mutation on randomly selected dimensions of duplicate solutions before executing the next generation

Step 6: *Stoppage Criteria*

- (a) Repeat the procedure from Step 2 to Step 5 till the termination criterion Step 6 is met.
- (b) Report the final solution obtained by the globally best learner (one yielding the highest value of the fitness function) at time $t = t_{\max}$.

In this experiment, Deb's heuristic constrained handling method [21] is used to handle the constraints. The rules are implemented at the end of the teacher phase (Step 3) and the learner phase (Step 5). Deb's method uses a tournament selection operator in which two solutions are selected and compared with each other. The following three heuristic rules are implemented on them for the selection:

- If one solution is feasible and the other is infeasible, then the feasible solution is preferred.
- If both the solutions are feasible, then the solution having the better objective function value is preferred.
- If both the solutions are infeasible, then the solution having the least constraint violation is preferred.

5 Experimental Analysis

The durability of AutoTLBO algorithm is assessed by compared various sized yeast datasets. The algorithm is implemented in MATLAB R2008a and run on a PC with Windows as OS and a core i3 processor operating at 2.93 GHz with 4 GB of RAM. All the results appeared in the analysis were the upshot of each dataset iterated with AutoTLBO algorithm after 50 independent runs. The following results correspond to the best solution attained by each algorithm, with respect to the coverage of assumed performance measures in the algorithm. The best results are displayed in boldface.

5.1 Experiment-1

The substantiated experimental analysis of the proposed algorithm is given in Table 1. Table 1 consolidates the results of *Saccharmyces cerevisiae* categorized yeast datasets with distinct sizes and dimensions on AutoTLBO. The optimality and efficiency of the algorithm is estimated by 4 parameters, i.e., the core concept of

Table 1 Results of automatic clustering using Elitist TLBO in microarray datasets

Datasets	Size	Dim	No. of auto clusters	CPU time (s)	% of error rate	Mean value of cluster validity indices (CVIs)						
						ARI	RI	MIM	HIM	SIL	CSM	DB
Yeast238	238	19	4.2	25.07	5.801	0.921	0.972	0.928	0.944	0.951	1.646	1.069
Yeast384	384	19	5.4	40.84	8.937	0.952	0.907	0.893	0.914	0.829	1.696	1.088
Yeast2885	2885	19	6.4	1,700.75	38.411	0.821	0.810	0.890	0.721	0.891	2.912	1.505
Yeast2946	2946	18	5.6	1,100.80	35.477	0.851	0.879	0.821	0.859	0.726	2.325	1.502
Yeast4382	4382	25	6.2	3,510.24	44.278	0.721	0.663	0.737	0.826	0.747	3.702	2.016

engendering optimal automatic clusters, cluster profiles, minimum utilization of CPU time, low percentage of error rate, and mean values of CVIs marching toward its thresholds to justify automatic clustering accuracy.

In case of yeast238 dataset, the optimal number of automatic clusters is 4.2 which is close to assumed value, the percentage of error rate and CPU time is low. The mean value of CVIs SIL has a high optimality value and DB has a moderate value. This justifies that AutoTLBO can be applied for microarray datasets.

Figure 2 articulates the four cluster profiles with 39, 26, 43, and 129 respective gene-IDs in yeast238 dataset, and Fig. 3a fortitudes to a hierarchal representation of clustered expression profiles in yeast238 dataset as heat maps. This output exhales a result-oriented concrete outcome to impart new momentum and energy to AutoTLBO for clustering microarray datasets.

The eligibility of the algorithm happens true when the dataset yeast384 is practiced over AutoTLBO. The algorithm attains a value of 5.4 as an optimal number of automatic clusters, with high favorable rate in ARI and DB indices' thresholds. An important observation was that the algorithm sustains the same minimal values both at error rate and CPU time. AutoTLBO is retrospective with belying expectations on yeast384 by leveraging five automatic clusters profiles with 68, 131, 45, 40, and 37 respective gene sizes. Figure 3b is fortitude in representing hierarchal cluster expression profiles on yeast384 dataset as a dendrogram. The result on yeast384 is a piecemeal but pragmatic with a promising growth that AutoTLBO is a landside in gene ontology with a corrective action.

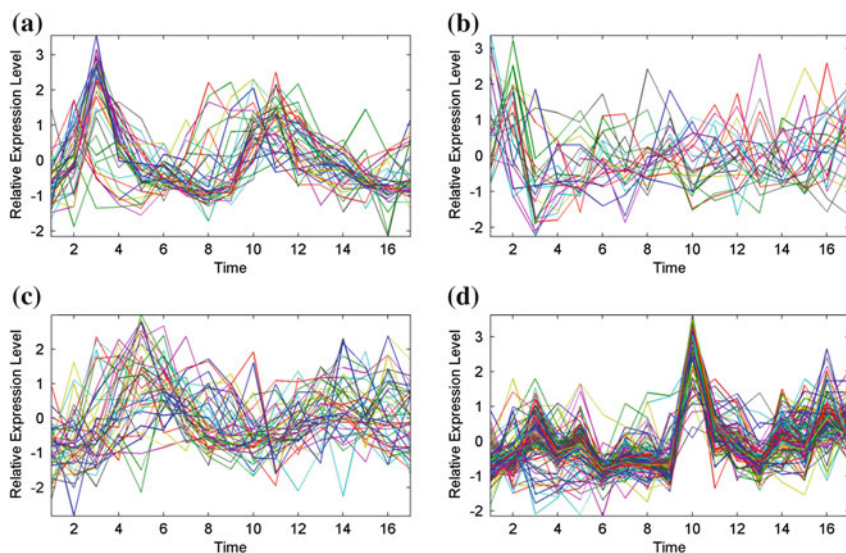


Fig. 2 Cluster profiles of co-expressed genes in yeast238 dataset. **a** cluster 4 of yeast238—cluster 1, **b** cluster 4 of yeast238—cluster 2, **c** cluster 4 of yeast238—cluster 3, **d** cluster 4 of yeast238—cluster 4

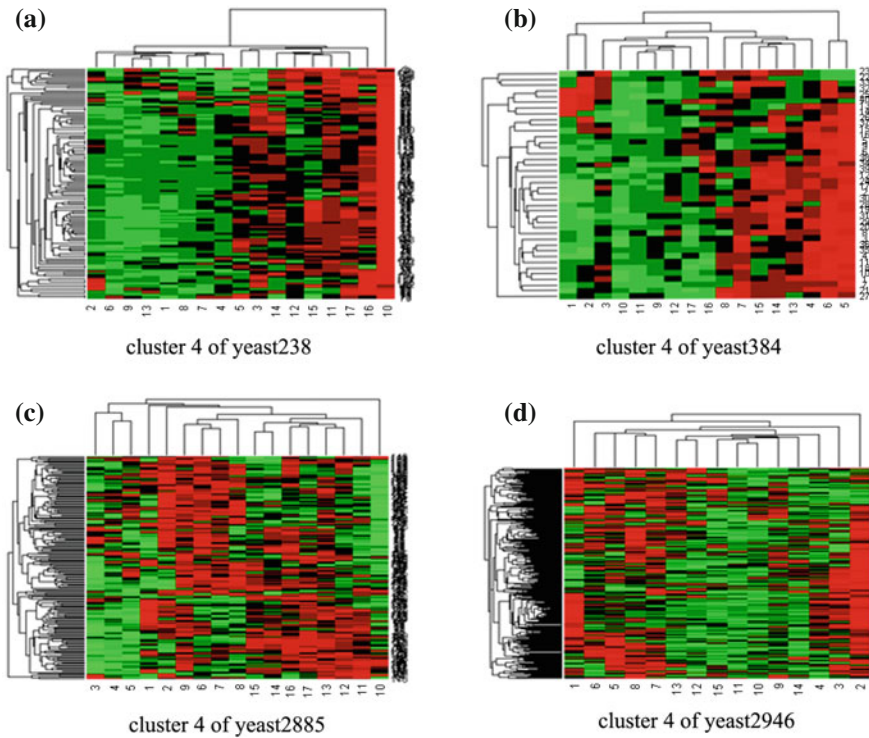


Fig. 3 Hierarchical expression profile of *S. cerevisiae* categorized datasets. **a** cluster 4 of yeast238, **b** cluster 4 of yeast384, **c** cluster 4 of yeast2885, **d** cluster 4 of yeast2946

The quality of the algorithm is inspected by applying with a huge quantity dataset called yeast2885 with 2,885 gene-IDs and 19 dimensions. The efficiency of algorithm is exhibited by producing a value 5.4 as an optimal number of auto clusters, low usage of CPU time, and low percentage of error rate. Optimal threshold values are recorded at SIL and MIM indices to highlight the competence of clustering accuracy. Hence, this outcome launches AutoTLBO as a mere and potential immigrant in clustering microarray datasets. The core ideology of generating the automatic cluster when entrenched over yeast2885 is visualized as Fig. 4. This figure justifies that the five respective constellations of gene-IDs 1,422, 367, 877, 388, and 1,327 are prudent and consummated to the actual. Figure 3c is impinged with a heat map over the fourth cluster of yeast2885. The analysis from the heat map [22] was the data matrix holds the color information of microarray dataset along with numeric data. The red color is evidence for higher expression level of the gene, whereas green indicates low expression level and black indicates the absence of expression level. The perception was so bedazzle that the heat maps generated by AutoTLBO have higher expression level since most of the cluster is marked in red color.

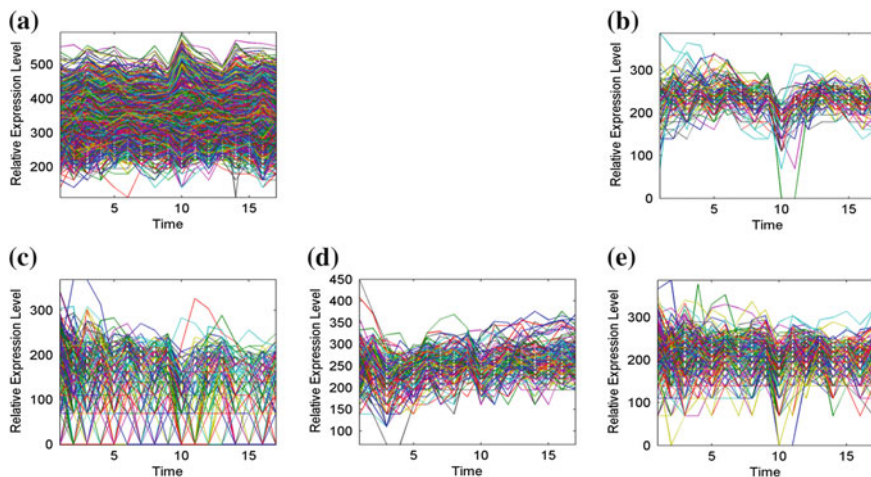


Fig. 4 Cluster profiles of co-expressed genes in yeast3 dataset with 2,885 instances. **a** dataset yeast2885—cluster 1, **b** dataset yeast2885—cluster 2, **c** dataset yeast2885—cluster 3, **d** dataset yeast2885—cluster 4, **e** dataset yeast2885—cluster 5

The investigation to estimate the capability of AutoTLBO is again protracted with almost on the same-sized dataset yeast2946, but with 18 dimensions. The proposed work makes its impact by producing the desired 5.6 automatic clusters with minimum percentage of error rate. The optimal mean CVI value is quoted in RI and HIM, proving the accuracy of automatic clusters despite the vast size of dataset. An important observation was yeast2885 consumes approximately 600 s of CPU time less when compared with yeast2946 dataset. The reason was the reduction in dimension of the microarray dataset makes that difference.

Figure 5 exhibits the inclusiveness of yeast2946 with six distinct sets of gene-ID of sizes 651, 193, 1,023, 656, and 422, respectively; also, the dispensation of fourth cluster expression profile is laid in as Fig. 3d. Both the figures exploit the importance of the proposed auto clustering method in the field of bioinformatics and recommends a revival strategy for the researchers to snoop into AutoTLBO.

The royalty of the work is exhibited when the algorithm is expended over the enormous volume of microarray dataset yeast5 with 4,382 instances of genes and 25 dimensions. Despite the size of the dataset the cluster accuracy sticks to the range of CVIs thresholds, by depicting an optimal value of 0.826 and 0.721, respectively, at HIM and ARI. AutoTLBO had a consistent eye watch over the percentage of error rate, but a moderate value of 44 % is recorded. The CPU time is coherent with the size of the dataset by producing accurate number of high volume clusters profiles. It is also noteworthy from AutoTLBO, to culminate five groups of discrete gene-IDs of 913, 967, 976, and 1,525 with respective sizes and obtain a tangible heat map. The hierarchical expression profile of the fourth cluster is visualized as shown in Fig. 6.

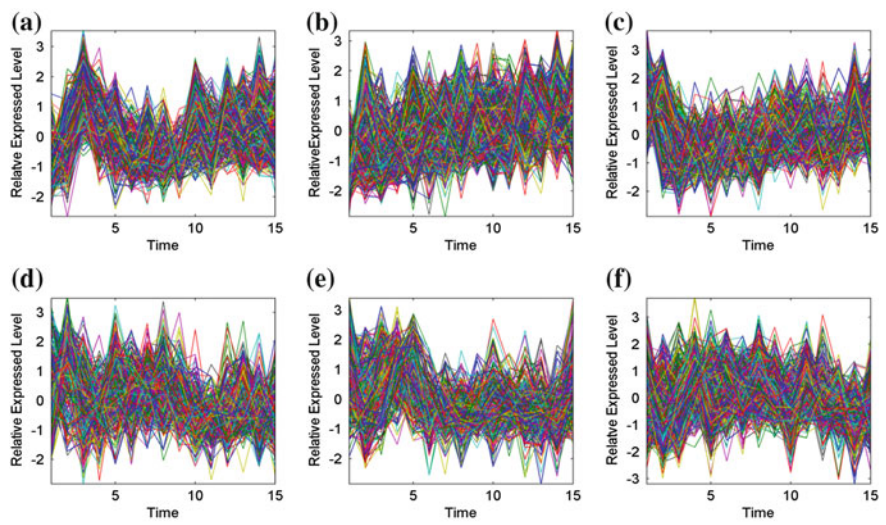
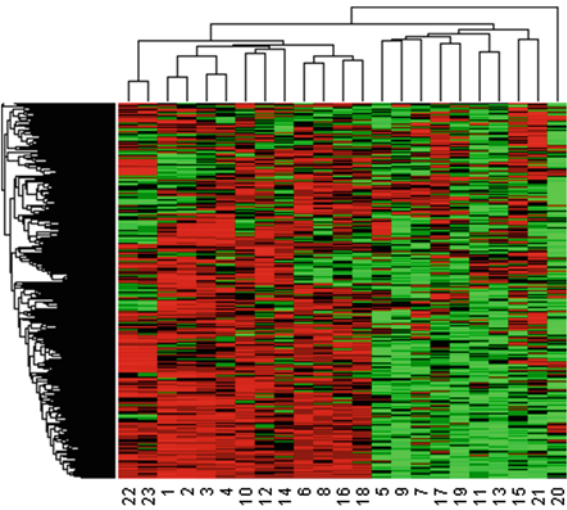


Fig. 5 Cluster profiles of co-expressed genes in yeast2946 dataset. **a** dataset yeast2946—cluster 1, **b** dataset yeast2946—cluster 2, **c** dataset yeast2946—cluster 3, **d** dataset yeast2946—cluster 4, **e** dataset yeast2946—cluster 5, **f** dataset yeast2946—cluster 6

Fig. 6 Hierarchical expression profile of yeast4382 dataset in the form of heat map



The upshots of the summative assessment-I are pragmatic by actioning all the constraints laid in Table 1. Hence, the promising results obtained by AutoTLBO over different-sized microarray datasets justify that the proposed work can be adopted in the field of bioinformatics to automatically cluster the gene profiles.

5.2 Experiment 2

This experiment focuses mainly to attain the gene functional enrichment of the incurred clusters of yeast datasets in Table 1 and re-validate the outcomes of AutoTLBO in experiment 1. The outputs of experiment 1 are used as inputs to the experiment 2. The Web-based functional annotation tools such as FatiGo [23] and DAVID [24] are used, particularly for gene-enrichment analysis. The clustered gene-IDs of each yeast datasets is segmented individually into two lists of genes, i.e., a group of interest as foreground and rest of genes as background lists. These lists are passed as inputs to the aforementioned tools for gene functional enrichment. The GO biological process is triggered on the gene lists to obtain significant results in range of selected level of gene ontology, gene annotations, *p*-value, etc. Since experiment 1 uses the yeast datasets, the genomics organism is treated as *S. cerevisiae*.

5.2.1 Outputs of FatiGo

Table 2 is inferred with the GO biological process applied between the levels 3–9 on all the five comparing datasets. The percentages of annotations produced by FatiGo in list 1 and list 2 of yeast234 and yeast384 datasets are outstanding. In the list 1 of yeast3 and yeast5, the percentage of annotations are 72 and 78.23 %, respectively, and in the list 2 of yeast3 and yeast5, the percentage of annotations are 73.48 and 77.66 %, respectively. This indicates that FatiGo has marked a reasonable good percentage of annotations on the list. The reason that was most looming was of the enormous size of data. The experimental results of Yeast2946 are not shown in this paper since the obtained results are merely equivalent to yeast2885. The clustering accuracy of TLBO was once again proven from column 6 and 7 of Table 2. The percentage of misclassification of the algorithm is zero in both in

Table 2 GO biological process on yeast datasets

GO biological process (levels from 3 to 9)						
Dataset	Total genes	ID annotations		Duplicate management		Number of significant terms
		List 1 annotations	List 2 annotations	List 1 duplicates	List 2 duplicates	
Yeast1	238	109 of 113 (96.4 %)	101 of 104 (97.12 %)	16 of 129 (0.12 %)	4 of 108 (0.04 %)	75
Yeast2	384	10 of 131 (83.97 %)	155 of 190 (81.58 %)	0 of 131 (0 %)	0 of 190 (0 %)	11
Yeast3	2,885	956 of 1,327 (72 %)	1,144 of 1558 (73.48 %)	0 of 1,327 (0 %)	0 of 1,558 (0 %)	6
Yeast5	4,382	1,193 of 1,525 (78.23 %)	2,218 of 2,856 (77.66 %)	0 of 1,525 (0 %)	0 of 2,856 (0 %)	4

Term	Term size	Term size (in genome)	Term annotation % per list	Annotated ids	Odds ratio (log e)	pvalue	Adjusted pvalue
translation (GO:0006412)	310	416	list 1: <div><div></div></div> 4.75% list 2: <div><div></div></div> 8.09%	list 1: YAL516W,YCR003W... list 2: YBR048W,YBR116W,YBR1...	-0.5684	0.00005055	0.01377
ribosome biogenesis (GO:0042254)	294	379	list 1: <div><div></div></div> 8.89% list 2: <div><div></div></div> 5.76%	list 1: YAL536C,YBL079W... list 2: YBR048W,YBR167C,YBR1...	0.4675	0.0002218	0.02278
RNA metabolic process (GO:0016070)	828	1139	list 1: <div><div></div></div> 23.81% list 2: <div><div></div></div> 16.76%	list 1: YAL511W,YBL066C... list 2: YAL529C,YAL599W,YBL5...	0.4394	7.223e-8	0.00003936
response to DNA damage stimulus (GO:0006974)	196	272	list 1: <div><div></div></div> 6.33% list 2: <div><div></div></div> 3.67%	list 1: YAR007C,YBL003C... list 2: YBL051C,YBL066C,YBR5...	0.5739	0.0001692	0.02278
alcohol metabolic process (GO:0006066)	163	217	list 1: <div><div></div></div> 2.19% list 2: <div><div></div></div> 4.39%	list 1: YBR056C,YCR036W... list 2: YAR059W,YBR196C,YBR2...	-0.7198	0.0002507	0.02278
RNA processing (GO:0006396)	328	462	list 1: <div><div></div></div> 9.87% list 2: <div><div></div></div> 6.45%	list 1: YBR247C,YCL054W... list 2: YAL529C,YBR167C,YBR2...	0.4628	0.0001311	0.02278

Fig. 7 Gene functional enrichment in yeast2 dataset with 2,885 genes

Term	Term size	Term size (in genome)	Term annotation % per list	Annotated ids	Odds ratio (log e)	pvalue	Adjusted pvalue
translation (GO:0006412)	310	416	list 1: <div><div></div></div> 4.79% list 2: <div><div></div></div> 8.3%	list 1: YAL516W,YCR003W... list 2: YDL033C,YDL083C,YDL2...	-0.5878	0.00001046	0.00285
RNA metabolic process (GO:0016070)	828	1139	list 1: <div><div></div></div> 22.69% list 2: <div><div></div></div> 16.88%	list 1: YAL511W,YBL066C... list 2: YBL008W,YBL052C,YBR0...	0.3684	0.000003771	0.002055
RNA modification (GO:0009451)	50	70	list 1: <div><div></div></div> 1.97% list 2: <div><div></div></div> 0.7%	list 1: YCL054W,YDL014W... list 2: YDL033C,YDL036C,YDL1...	1.0457	0.0002771	0.03776
folic acid and derivative metabolic process (GO:0006760)	8	14	list 1: <div><div></div></div> 0.52% list 2: <div><div></div></div> 0%	list 1: YER193C,YGR067C... list 2: no ids	1.7976931349e+304	0.000213	0.03776

Fig. 8 Gene functional enrichment in yeast2 dataset with 4,800 genes

yeast2885 and yeast4382 and very low in yeast238 and yeast384. The number of significant terms quoted by FatiGo was very low, since the clusters submitted by the algorithm were very precise. Hence, all these evidences establish AutoTLBO as a top-drawer in associating biological phrases.

Figures 7 and 8 present the significant terms of yeast2885 and yeast4382 when p -value is less than 0.05. The nominal p -value forecasts the significance of the enrichment score for a single gene set.

5.2.2 Outputs of DAVID

The approach adopted in FatiGo tool is also practiced in this DAVID bioinformatics tools. The p -value generated on the yeast dataset via this tool is much similar to the outputs attained in FatiGo. This tool is used to judge the dataset with few more additional parameters. The results of yeast2885 dataset in DAVID tool are shown as Fig. 9. An optimal functional enrichment score of 0.93 is obtained for the annotation cluster. In the preprocessing stage, the Fisher exact statistical test is used to obtain the clean lists. Multiple comparison solutions between the list of foreground and background gene-ID is given as the false discovery rate. The significant value of 0.1 is acceptable for screening, and a list of independent, continuous dependent, normal gene-ID list is prepared. The Benjamini test type holds a cutoff of p -value to 0.05 and expect 0.05 genes to be significant by chance, and Bonferroni holds a cutoff of

Annotation Cluster 1	Enrichment Score: 0.93			Count	P_Value	Fold Change	Bonferroni	Benjamini	FDR	LT_PH,PT
GOTERM_CC_FAT	cytosolic ribosome	RT	↓	3	3.5E-2	8.5E0	6.5E-1	6.5E-1	2.6E1	8,97,2209
GOTERM_BP_FAT	regulation of translation	RT	↓	3	4.5E-2	7.6E0	9.9E-1	9.9E-1	4.0E1	9,105,2382
GOTERM_BP_FAT	posttranscriptional regulation of gene expression	RT	↓	3	5.3E-2	7.0E0	1.0E0	9.3E-1	4.5E1	9,114,2382
GOTERM_BP_FAT	regulation of cellular protein metabolic process	RT	↓	3	5.5E-2	6.8E0	1.0E0	8.5E-1	4.7E1	9,117,2382
KEGG_PATHWAY	Ribosome	RT	↓	3	6.2E-2	5.5E0	3.2E-1	3.2E-1	2.6E1	5,87,792
GOTERM_CC_FAT	cytosolic part	RT	↓	3	6.2E-2	6.2E0	8.5E-1	6.2E-1	4.2E1	8,133,2209
GOTERM_CC_FAT	ribosomal subunit	RT	↓	3	7.1E-2	5.8E0	8.9E-1	5.2E-1	4.7E1	8,144,2209
SP_PIR_KEYWORDS	ribosome	RT	↓	3	7.6E-2	6.0E0	9.5E-1	9.5E-1	5.1E1	13,115,2968
GOTERM_MF_FAT	structural constituent of ribosome	RT	↓	3	1.0E-1	4.8E0	9.6E-1	9.6E-1	5.9E1	9,141,2046
SP_PIR_KEYWORDS	ribosomal protein	RT	↓	3	1.1E-1	4.9E0	9.8E-1	7.5E-1	6.4E1	13,140,2968
GOTERM_CC_FAT	ribosome	RT	↓	3	1.3E-1	4.1E0	9.8E-1	6.4E-1	6.9E1	8,201,2209
SP_PIR_KEYWORDS	protein biosynthesis	RT	↓	3	1.4E-1	4.1E0	1.0E0	7.6E-1	7.5E1	13,167,2968
SP_PIR_KEYWORDS	ribonucleoprotein	RT	↓	3	1.6E-1	3.9E0	1.0E0	7.2E-1	7.8E1	13,176,2968
GOTERM_CC_FAT	cytosol	RT	↓	3	1.6E-1	3.6E0	9.9E-1	6.5E-1	7.7E1	8,230,2209
GOTERM_MF_FAT	structural molecule activity	RT	↓	3	1.8E-1	3.4E0	1.0E0	8.6E-1	8.2E1	9,200,2046
GOTERM_CC_FAT	ribonucleoprotein complex	RT	↓	3	2.6E-1	2.6E0	1.0E0	7.8E-1	9.3E1	8,315,2209
GOTERM_BP_FAT	translation	RT	↓	3	3.6E-1	2.1E0	1.0E0	1.0E0	9.9E1	9,373,2382
GOTERM_CC_FAT	non-membrane-bounded organelle	RT	↓	3	5.4E-1	1.5E0	1.0E0	9.6E-1	1.0E2	8,537,2209
GOTERM_CC_FAT	intracellular non-membrane-bounded organelle	RT	↓	3	5.4E-1	1.5E0	1.0E0	9.6E-1	1.0E2	8,537,2209

Fig. 9 Gene functional enrichment in yeast2 dataset with 2,885 genes in DAVID Tool

Annotation Cluster 1	Enrichment Score: 1.29			Count	P_Value	Fold Change	Bonferroni	Benjamini	FDR	LT_PH,PT
GOTERM_BP_FAT	regulation of translation	RT	↓	4	8.4E-3	8.1E0	6.4E-1	6.4E-1	9.2E0	11,100,2224
GOTERM_BP_FAT	posttranscriptional regulation of gene expression	RT	↓	4	1.0E-2	7.5E0	7.2E-1	4.7E-1	1.1E1	11,108,2224
GOTERM_BP_FAT	regulation of cellular protein metabolic process	RT	↓	4	1.1E-2	7.4E0	7.3E-1	3.6E-1	1.2E1	11,110,2224
GOTERM_CC_FAT	ribosome	RT	↓	4	3.1E-2	4.8E0	6.1E-1	6.1E-1	2.3E1	9,191,2071
SP_PIR_KEYWORDS	protein biosynthesis	RT	↓	4	4.3E-2	4.6E0	8.7E-1	8.7E-1	3.4E1	15,160,2782
GOTERM_CC_FAT	ribonucleoprotein complex	RT	↓	4	9.9E-2	3.0E0	9.6E-1	4.6E-1	5.9E1	9,303,2071
GOTERM_BP_FAT	translation	RT	↓	4	2.0E-1	2.3E0	1.0E0	1.0E0	9.2E1	11,350,2224
GOTERM_CC_FAT	non-membrane-bounded organelle	RT	↓	4	3.2E-1	1.8E0	1.0E0	8.1E-1	9.6E1	9,517,2071
GOTERM_CC_FAT	intracellular non-membrane-bounded organelle	RT	↓	4	3.2E-1	1.8E0	1.0E0	8.1E-1	9.6E1	9,517,2071
Annotation Cluster 2	Enrichment Score: 0.92			Count	P_Value	Fold Change	Bonferroni	Benjamini	FDR	LT_PH,PT
GOTERM_CC_FAT	cytosolic ribosome	RT	↓	3	4.9E-2	7.3E0	7.8E-1	5.3E-1	3.5E1	9,95,2071
KEGG_PATHWAY	Ribosome	RT	↓	3	6.7E-2	5.2E0	3.4E-1	3.4E-1	2.8E1	5,85,742
GOTERM_CC_FAT	cytosolic part	RT	↓	3	8.1E-2	5.5E0	9.2E-1	5.7E-1	5.1E1	9,126,2071
GOTERM_CC_FAT	ribosomal subunit	RT	↓	3	9.5E-2	5.0E0	9.5E-1	5.3E-1	5.7E1	9,138,2071
SP_PIR_KEYWORDS	ribosome	RT	↓	3	1.0E-1	5.1E0	9.9E-1	9.2E-1	6.5E1	15,110,2782
SP_PIR_KEYWORDS	ribosomal protein	RT	↓	3	1.4E-1	4.2E0	1.0E0	9.1E-1	7.7E1	15,134,2782
GOTERM_MF_FAT	structural constituent of ribosome	RT	↓	3	1.5E-1	3.9E0	1.0E0	1.0E0	8.0E1	11,135,1918
GOTERM_CC_FAT	cytosol	RT	↓	3	2.0E-1	3.2E0	1.0E0	6.7E-1	8.5E1	9,217,2071
SP_PIR_KEYWORDS	ribonucleoprotein	RT	↓	3	2.1E-1	3.3E0	1.0E0	9.4E-1	8.9E1	15,170,2782
GOTERM_MF_FAT	structural molecule activity	RT	↓	3	2.7E-1	2.7E0	1.0E0	1.0E0	9.5E1	11,196,1918

Fig. 10 Gene functional enrichment in yeast2 dataset with 4,382 genes

p -value equals to 0.05, to identify 5 % of the genes. The statistically significant values of these parameters are shown as column 8 and 9 in Fig. 9. The list total, population list, and population total are genes are shown as column 11 in Fig. 9.

Figure 10 displays the cluster obtained on yeast4382 with DAVID. The gene enriched in annotation term is scored as 1.29, which is an optimal value for the dataset of 4,382 gene-IDs. Figure visualizes the annotated clusters of yeast4382, whose p -value is <0.05 . The impressive outputs of different standard statistics for

multiple comparison corrections such as Benjamini, Bonferroni, LT, PH, and PT are given in column 8, 9, and 11 of Fig. 10.

The upshots of this summative assessment-II are indeed noteworthy. Realistic results are achieved by the usage of bioinformatics tools because the input quality was magnificent. Hence, this re-validation rationalizes the results produced by AutoTLBO is superlative.

6 Conclusion

This paper articulates a novel automatic clustering algorithm using TLBO for achieving gene functional enrichments. The results of AutoTLBO were again re-validated using benchmarked bioinformatics tools. Both the assessments substantiate that the AutoTLBO algorithm underscored in this paper has accurate creditability in yielding impending outputs. Thus, the hybrid approach of using this AutoTLBO algorithm of engineering studies in bioinformatics datasets is practical. As this paper is classified only to *S. cerevisiae* organism, the future envisaged scope of this work was to introspect AutoTLBO on the other assemblies of molecular functions in the gene ontology biological process.

References

1. Rao RV, Savsani VJ, Vakharia DP (2011) Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Computer Aided Des* 43:303–315. doi:[10.1016/j.cad.2010.12.015](https://doi.org/10.1016/j.cad.2010.12.015)
2. Rao RV, Savsani VJ, Vakharia DP (2012) Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems. *Inf Sci* 183:1–15. doi:[10.1016/j.ins.2011.08.006](https://doi.org/10.1016/j.ins.2011.08.006)
3. Rao RV, Patel V (2013) Comparative performance of an elitist teaching-learning-based optimization algorithm for solving unconstrained optimization problems. *Int J Ind Eng Comput* 4: 29–50. doi:[10.5267/j.ijiec.2012.09.001](https://doi.org/10.5267/j.ijiec.2012.09.001)
4. Rao RV, Patel V (2012) An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems. *Int J Ind Eng Comput* 3:535–560. doi:[10.5267/j.ijiec.2012.03.007](https://doi.org/10.5267/j.ijiec.2012.03.007)
5. Rand WM (1971) Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc* 66(336):846–850
6. Hubert Lawrence, Schultz James (1976) Quadratic assignment as a general data analysis strategy. *Br J Math Stat Psychol* 29(2):190–241
7. Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math* 20:53–65
8. Davies DL, Bouldin DW (1979) A cluster separation measure. *Pattern Anal Mach Intell IEEE Trans On* 2:224–227
9. Chou C-H, Su M-C, Lai Eugene (2004) A new cluster validity measure and its application to image compression. *Pattern Anal Appl* 7(2):205–220

10. Rao RV, Patel V (2013) An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. *Sci Iranica D* 20(3):710–720. doi:[10.1016/j.scient.2012.12.005](https://doi.org/10.1016/j.scient.2012.12.005)
11. Rao RV, Waghmare GG (2014) A comparative study of a teaching–learning-based optimization algorithm on multi-objective unconstrained and constrained functions. *J King Saud University—Comput Inf Sci* 26: 332–346. doi:[10.1016/j.jksuci.2013.12.004](https://doi.org/10.1016/j.jksuci.2013.12.004)
12. Amiri Babak (2012) Application of teaching-learning-based optimization algorithm on cluster analysis. *J Basic Appl Sci Res* 2(11):11795–11802
13. Suresh K, Kundu D, Ghosh S, Das S, Abraham A (2009) Automatic clustering with multi-objective differential evolution algorithms. In: *Evolutionary computation, 2009, IEEE Congress on CEC'09*. IEEE, pp 2590–2597
14. Kundu D, Suresh K, Ghosh S, Das S, Abraham A, Badr Y (2009) Automatic clustering using a synergy of genetic algorithm and multi-objective differential evolution. In: *Hybrid artificial intelligence systems*. Springer, Berlin, pp 177–186
15. Liu Yimin, Özyer Tansel, Alhajj Reda, Barker Ken (2005) Integrating multi-objective genetic algorithm and validity analysis for locating and ranking alternative clustering. *Informatica* 29:33–40
16. Satapathy SC, Naik A, Parvathi K (2013) A teaching learning based optimization based on orthogonal design for solving global optimization problems. *SpringerPlus* 2:130
17. Naik A, Satapathy SC, Parvathi K (2012) Improvement of initial cluster center of c-means using teaching learning based optimization. *Procedia Technol* 6:428–435. doi:[10.1016/j.protcy.2012.10.051](https://doi.org/10.1016/j.protcy.2012.10.051)
18. Murty MR et al (2014) Automatic clustering using teaching learning based optimization. *Appl Math* 5:1202–1211. doi:[10.4236/am.2014.58111](https://doi.org/10.4236/am.2014.58111)
19. Suresh Kaushik, Kundu Debarati, Ghosh Sayan, Das Swagatam, Abraham A, Han SY (2009) Multi-objective differential evolution for automatic clustering with application to micro-array data analysis. *Sensors* 9:3981–4004. doi:[10.3390/s90503981](https://doi.org/10.3390/s90503981)
20. Pavan KK, Rao AA, Dattatreya Rao AV, Sridhar GR (2011) Robust seed selection algorithm for k-means type algorithms. *Int J Comput Sci Inf Technol (IJCSIT)* 3(5). doi:[10.5121/ijcsit.2011.3513](https://doi.org/10.5121/ijcsit.2011.3513)
21. Deb Kalyanmoy (2000) An efficient constraint handling method for genetic algorithms. *Comput Methods Appl Mech Eng* 186(2):311–338
22. Wilkinson L, Friendly M (2009) The history of the cluster heat map. *The American Statistician* 63(2)
23. Al-Shahrour F, Minguez P, Tárraga J, Medina I, Alloza E, Montaner D, Dopazo J (2007) FatiGO+: a functional profiling tool for genomic data. Integration of functional annotation, regulatory motifs and interaction data with microarray experiments. *Nucleic Acids Research* 35 (Web Server issue):W91–W96
24. Dennis G, Sherman BT, Hosack DA, Yang J, Baseler MW, Lane HC, Lempicki RA (2003) DAVID: database for annotation, visualization, and integrated discovery. *Genome Biology* 4 (5):P3

Computational Intelligence Techniques for Comparative
Genomics

Dedicated to Prof. Allam Appa Rao on the Occasion of
His 65th Birthday

Muppalaneni, N.B.; Gunjan, V.K. (Eds.)

2015, XVIII, 137 p. 61 illus., 41 illus. in color., Softcover

ISBN: 978-981-287-337-8