Research Journal of Computer Systems and Engineering

ISSN: 2230-8571, 2230-8563

Volume 02 Issue 01 - 2021 (January to June)

Page 23:27

Clustering and Optimization Based on Hybrid Artificial Bee Colony and Differential Evolution Algorithm in Big Data

Dr. S.A.Sivakumar
Associate Professor, Head-ECE,
Ashoka Women's Engineering College,
Kurnool - 518 218, Andhra Pradesh, India
drsasivakumar@gmail.com

Article History

Received: 22 January 2021
Revised: 14 April 2021
Accepted: 19 May 2021

Abstract

Big data is increasingly being employed in a variety of fields because it can handle the difficulties associated with processing massive amounts of data, including business, financial transactions, medical, and more. In this paper, an approach for optimizing for searching big data is developed using Hybrid Artificial Bee Colony and Differential evolution algorithm (HABCO_DE) and finally the optimized search data is classified by using optimized, The suggested HABCO DE technique then classifies each data sample based on posterior probability of data and probability index table. Using Multi-Layer Perceptron (MLP) neural networks, a real-time, precise search target classifier was designed. Simulation results indicate that the HABCO_DE can effectively be used for data clustering.

Keywords: Big data, classification, clustering, optimization, search targets classifier

1. Introduction

In practically every field, including science and engineering, such as genomics, remote sensing, medical imaging, finance, and people's personal life with the rise of social media, recent technological breakthroughs and globalisation have enhanced data collecting capability [1].Certainly, the micro-array and other comparable techniques, which have advanced quickly and significantly over the past few years, contributed to the effect of clustering gene expression data [2]. The study of methods for selecting the optimum solution(s) among a set of choices constrained by a set of constraints falls under the umbrella of optimization, a branch of mathematics, computer science, and operations research [3].There are many similarities between data mining and optimization, despite the fact that they are two distinct fields of study. For instance, a classification problem can be thought of as an optimization problem because the objective is to maximise classification accuracy and minimise complexity while meeting certain constraints [4].

2. Related Works

This section reviews numerous methods now in use. First, VAT [5] discusses clustering through the dissimilarity matrix to create a modified matrix that displays different clusters as dark blocks diagonally, as is done in dark matter haloes, however this method only functions for large cluster data. Dendrograms were employed in [6] to create a clustering by anomaly detection method, which was then applied to many taxonomy applications [7]. Similar to this, the author in [8] suggested single

linkage-based clustering to divide time series data into sections for patient monitoring. Security was provided by the VAT commercial application [9], and it is further noted that K-means promises to efficiently cluster data. One of the main study areas in recent years has been deep learning; supervised learning tasks that have produced good results in big data clustering do not yield results among the raw data, which has an impact on accuracy. Researchers in [10] created a hierarchical method that combines fuzzy logic and neural networks for robust grouping, minimising ambiguity. In the literature [11], a fuzzy-based CNN model was created for classification and clustering. Here, CNN was first utilised to automate feature extraction from each given input image and then the FCM technique was used to cluster the data in a specified feature space.

3 Methodology:

3.1 Improvised Fuzzy C-Means approach

The FCM algorithm works by assigning each data point's membership to a corresponding CC based on data point and cluster distance; its key benefit is that it consistently produces excellent results even when the data are overlapped, and it also assigns each data point to several clusters.Let us consider the dataset $Z = \{z_1, z_2, \dots, z_q\}$ with cluster set$X = \{x_1, x_2, \dots, x_p\}$ and membership set W=$\begin{Bmatrix} \text{wkl} \mid 1 \le \text{k} \le \text{e}, 1 \le \text{l} \le \\ \text{p} \end{Bmatrix}$ further considering these three FCM can be formulated. Suggested mechanism's general premise is to combine an IFCM with a double NN. To train the example, we further create an efficient auto-encoder in eq. (1).

$$\min: \sum_{k=1}^{e}\sum_{l=1}^{p} w_{kl}^{o}\|z_l - x_k\|^2 \sum_{l=1}^{e} w_{kl} = 1, w_{kl} \ge 0 \qquad (1)$$

We create modified-FCM, or Modified FCM, in the equation below to prevent spurious clustering by eq. (2).

$$L_o(W, X) = \sum_{k=1}^{e}\eta_i \sum_{k=1}^{p}(1 - u_{kl}^{o})^o + \sum_{k=1}^{e}\sum_{k=1}^{p} w_{kl}^{m}\|z_l - z_i\|^2 \qquad (2)$$

In order to update the membership matrix and cluster centres, the equation can be optimised as eq. (3):

$$x_k = \sum_{l=1}^{p} w_{kl}^{o} z_l / \sum_{l=1}^{p} w_{kl} \qquad (3)$$

Membership matrix by eq. (4)

$$w_{kl} = \left(1 + \left(\frac{e_{kl}}{\eta_k}\right)^{-1/(o-1)}\right)^{-1} \qquad (4)$$

The term "ekl" in the equation above denotes the separation between the membership matrix and the cluster.

Optimized NN takes input as$Z \in T^{K_1 \times K_2 \dots\dots \times K_P}$ and reconstruction of same is represented as $Z \in T^{K_1 \times K_2 \dots \times K_P}$.

3.2 Artificial Bee Colony (ABC) Algorithm and Data Collection

There are four major steps in the ABC algorithm. First, initialization Assume that population is SN in size, with N being population's initial food source. Xi = {Xi1, Xi2,...,} (i = 1, 2, . . . , N), where D is the optimization problem's vector dimension. Subsequent population is drawn at random by eq. (5).

$$X_{ij} = X_{\min j} + \text{rand}(0,1)\left(x_{\max j} - X_{\min j}\right) \qquad (5)$$

(2) Population Updating. Hired bee switches to the new food source if the amount of nectar there is higher than the previous one; otherwise, it stays with the old one by eq. (6).

$$V_{ij} = X_{ij} + \text{rand}(-1,1) \cdot \left(X_{ij} - X_{kj}\right) \qquad (6)$$

(3) Bee Source Selection.The hired bees in this stage fly in accordance with the revenue rate of their sources as determined by their fitness value. The following equation states that food sources with higher income rates are more likely to be eq. (7):

$$P_i = \frac{\text{fit}(X_i)}{\sum_{n=1}^{\text{sN}} \text{fit}\,(X_n)} \qquad (7)$$

$\text{fit}(X_n) = \begin{cases} \frac{1}{f(X_n)}, & f(X_n) \geq 0 \\ 1 + \text{abs}\left(f(X_n)\right), & f(X_n) < 0, \end{cases}$ where $f(Xn)$ is objective function value of bee source Xn. The search patterns of the followed bees are closer to the sources, which enhances the algorithm's capacity for local exploitation.

(4) Population Elimination. The related spectator bees transform into scouting bees and randomly develop a new solution to replace abandoned one if a particular solution doesn't show any discernible improvement after continuous limit cycling updates by eq. (8).

$$X_{ij} = X_{\min j} + \text{rand}(0,1)\left(x_{\max j} - X_{\min j}\right) \qquad (8)$$

3.3 Differential Evolution

When creating new vectors, DE varies from other EAs primarily in that it adds weighted difference vector (DV) between two individuals to a third individual. When viable patches are parallel to the axes, it works well. The following is a brief discussion of its primary operators.

Mutation: difference between two random vectors is multiplied by an amplification factor, F, to create a mutant vector, which is then added to a third random vector (DE/rand/1) by eq. (9).

$$\vec{v}_z = \vec{x}_{r_1,j} + F \cdot \left(\vec{x}_{r_{2,j}} - \vec{x}_{r_{3,j}}\right) \qquad (9)$$

Crossover: The binomial and exponential crossover techniques are the two most used ones. The exponential crossover begins with a random selection of an integer index, l, from range [1, D], where D is issue dimension. This index serves as starting point in target vector where the donor vector's variable exchange process starts. The donor vector provides a certain number of components to the target vector, denoted by the integer index L, so that L [1, D]. A trial vector is then calculated as eq. (10):

$$u_{z,j} = \begin{cases} v_{z,j} \text{ for } j = \langle l \rangle_D, \langle l+1 \rangle_D, \ldots, \langle l+L-1 \rangle_D \\ x_{z,j} \forall j \in [1, D] \end{cases}$$

$$u_{z,j} = \begin{cases} v_{z,j} & \text{if (rand } \leq cr \text{ or } j = j_{\text{rand}} \text{)} \\ x_{z,j} & \text{otherwise} \end{cases} \qquad (10)$$

Selection The selection method is straightforward: if a child outperforms its parent, it survives to the next generation.

3.4 modified Whale Optimization Algorithm (mWOA)

The best technique to arrive at a global minimum in population-based optimization methods happens in two main stages. The search space has to be searched for persons in the early stages of optimization. Instead of grouping in the smallest local area, they must search the entire search space. Individuals are required to apply the data gathered for minimising global convergence, which are represented by Eq., in the final phases (11).

$$\alpha = 2\left(1 - \frac{t}{T}\right) \qquad (11)$$

where T stands for greatest number of iterations and t stands for current iteration. Generally speaking, expanding your search field will lessen your chances of experiencing local optima stagnation. Different strategies can be used to speed up exploration. In this regard, some non-linear functions are proposed to reduce the values of in order to balance the exploration and exploitation phases. These non-linear functions have varying slopes and distinct curve shapes.

4. Performance Evaluation:

MATLAB is the simulation tool used on a desktop with Intel i7 processor at 1.8 GHz and 16GB of RAM and represents a synthetic information gathering problem in an area of 30 m × 30 m.

Table.1.Comparison of Existing (with Existing) and Proposed

Metrics	Existing Without Preprocessing (%)	Proposed With Preprocessing (Morlet Wavelet De-Noise) (%)
Accuracy	85	96
Precision	72	81
Recall	65	71
RMSE	42	46
F-Measure	55	65

The above table-1 shows comparative analysis between proposed and existing techniques in terms of accuracy, precision, recall, F_1 score, RMSE. Here the analysis has been carried out based on number of epochs. Accuracy calculation is done by the general prediction capability of projected DL method. For calculating F-score, number of images processed are EEG signal for both existing and proposed technique. The F-score reveals each feature ability to discriminate independently from other features. For the first feature, a score is generated, and for the second feature, a different score is obtained. However, it says nothing about how the two elements work together. Here, calculating the F-score using exploitation has determined the prediction performance. It is created by looking at the harmonic component of recall and precision. If the calculated score is 1, it is considered excellent, whereas a score of 0 indicates poor performance. The actual negative rate is not taken into consideration by F-measures. The accuracy of a class is calculated by dividing the total items classified as belonging to positive class by number of true positives. Probability that a classification function will produce a true positive rate when present. It is also known by the acronym TP amount. In this context, recall is described as ratio of total number of components that genuinely fall into a positive class to several true positives. How well a method can recognise Positive samples is calculated by recall. Recall increases as more positive samples are determined. RMSE is one of the most often used metrics to assess how accurately our forecasting model predicts values compared to real or observed values while training regression or time series models.MSE squared root is used to calculate RMSE. The RMSE calculates the change in each pixel as a result of processing.

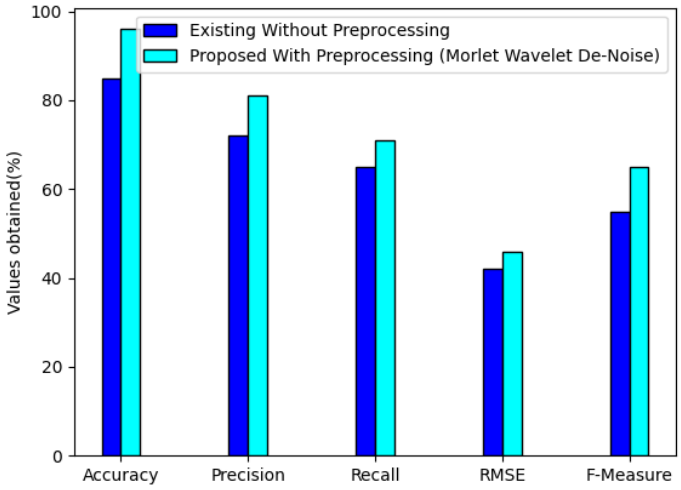

Figure 1. Comparative analysis between proposed and existing techniques

From above figure 1 shows comparative analysis between proposed and existing technique. the proposed technique attained accuracy of 96%, precision of 81%, recall of 71%, F-1 score of 65%, RMSE of 46%. While the existing ICPE attained accuracy of 82%, precision of 72%, recall of 65%, F-1 score of 55%, RMSE of 42%; CDA attained accuracy of 88%, precision of 76%, recall of 68%, F-1 score of 59%, RMSE of 44%.

Reference:

[1] Herrera, V. M., Khoshgoftaar, T. M., Villanustre, F., &Furht, B. (2019). Random forest implementation and optimization for Big Data analytics on LexisNexis's high performance computing cluster platform. *Journal of Big Data*, *6*(1), 1-36.

[2] Ilango, S. S., Vimal, S., Kaliappan, M., &Subbulakshmi, P. (2019). Optimization using artificial bee colony based clustering approach for big data. *Cluster Computing*, *22*(5), 12169-12177.

[3] Singh, N., Singh, D. P., & Pant, B. (2019). ACOCA: ant colony optimization based clustering algorithm for big data preprocessing. *International Journal of Mathematical, Engineering and Management Sciences*, *4*(5), 1239.

[4] Kushwaha, N., & Pant, M. (2018). Link based BPSO for feature selection in big data text clustering. *Future generation computer systems*, *82*, 190-199.

[5] He, R., Ai, B., Molisch, A. F., Stuber, G. L., Li, Q., Zhong, Z., & Yu, J. (2018). Clustering enabled wireless channel modeling using big data algorithms. *IEEE Communications Magazine*, *56*(5), 177-183.

[6] Wu, J., Dong, M., Ota, K., Li, J., & Guan, Z. (2018). Big data analysis-based secure cluster management for optimized control plane in software-defined networks. *IEEE Transactions on Network and Service Management*, *15*(1), 27-38.

[7] Hossain, M. S., Moniruzzaman, M., Muhammad, G., Ghoneim, A., &Alamri, A. (2016). Big data-driven service composition using parallel clustered particle swarm optimization in mobile environment. *IEEE Transactions on Services Computing*, *9*(5), 806-817.

[8] Zhang, C., Hao, L., & Fan, L. (2019). Optimization and improvement of data mining algorithm based on efficient incremental kernel fuzzy clustering for large data. *Cluster Computing*, *22*(2), 3001-3010.

[9] Ilango, S. S., Vimal, S., Kaliappan, M., &Subbulakshmi, P. (2019). Optimization using artificial bee colony based clustering approach for big data. *Cluster Computing*, *22*(5), 12169-12177.

[10] Herrera, V. M., Khoshgoftaar, T. M., Villanustre, F., &Furht, B. (2019). Random forest implementation and optimization for Big Data analytics on LexisNexis's high performance computing cluster platform. *Journal of Big Data*, *6*(1), 1-36.