Enhancing CURE Algorithm with Stochastic Neighbor Embedding (CURE-SNE) for Improved Clustering and Outlier Detection

Dewi Sartika Br Ginting*, Syahril Efendi, Amalia, Poltak Sihombing
Department of Computer Science and Technology Information, Universitas Sumatera Utara, Medan, Indonesia

***Abstract*—This study focuses on analyzing stunting data using the CURE and CURE-SNE algorithms for clustering and outlier detection. The primary challenge is identifying patterns in stunting data, which includes variables such as age, gender, height, weight, and nutritional status. Both algorithms were employed to group the data and detect outliers that may affect the results of the analysis. The evaluation methods included determining the optimal number of clusters using the silhouette score and assessing cluster quality using the Davies-Bouldin Index (DBI). The results showed that both algorithms formed four clusters, with CURE-SNE detecting 6,050 outliers, while CURE detected 5,047 outliers. Silhouette score analysis revealed that both algorithms formed four optimal clusters. However, when validated using DBI, CURE achieved a score of 0.523, while CURE-SNE produced a lower score of 0.388, indicating that CURE-SNE outperformed CURE in terms of cluster quality. This suggests that CURE-SNE not only detects more outliers but also produces clusters with better separation and compactness. The findings highlight that both algorithms are effective for clustering stunting data, but CURE-SNE excels in terms of outlier detection and overall cluster quality. Thus, CURE-SNE is more suitable for handling complex datasets with potential outliers, providing more accurate insights into the structure of the data. In conclusion, CURE-SNE demonstrates superior performance compared to CURE, offering a more reliable and detailed clustering solution for stunting data analysis.**

***Keywords*—Stunting; clustering algorithm; CURE; CURE-SNE; outliers**

I. INTRODUCTION

Dataset readiness is a crucial step in the clustering process, because clean and structured data will produce more accurate and reliable clusters. Optimal dataset readiness is also an important foundation for producing accurate and representative clusters. In the clustering process, detecting and handling outliers is a critical step to prevent biased and unreliable results. In its application, outlier detection is very important, because outliers often cause distortion in cluster formation, especially when the distribution of the dataset is not uniform, resulting in the formation of less accurate clusters and even potentially influencing business decisions or misguided analysis. Outliers can come from recording errors, irrelevant data, or extreme variations that do not reflect the general trend of the dataset. Without proper detection and handling, outliers can attract cluster centers or blur the boundaries between clusters, thus making clustering results less than optimal and even misleading. A recent study published in IEEE Transactions on Knowledge and Data Engineering (2021) [1] emphasizes that the presence of outliers in a retail company's customer dataset interferes with the interpretation of customer segments and leads to less representative clustering results. Another in the Journal of Cleaner Production (2021) [2] shows that clustering algorithms density-based ones, such as DBSCAN, have better performance in automatically detecting outliers than conventional algorithms, thereby increasing the accuracy of segmentation results in user behavior analysis.

The CURE (Clustering Using Representatives) algorithm is a clustering approach that is superior in handling large datasets and has the ability to detect non-spherical cluster shapes, and is more resistant to outliers than classical algorithms such as K-Means. CURE works by selecting multiple point representations from each cluster, then compressing (shrinking) these points to the center of the cluster to increase robustness against outliers. However, despite these advantages, CURE still has limitations in handling large datasets optimally, especially if there are many outliers. The use of sampling in CURE to reduce the scale of large datasets can result in important information being missed or even failing to identify relevant outliers. Additionally, point compression methods sometimes sacrifice certain details that are actually important in complex datasets, so clustering results may not always be optimal.

The development of the CURE algorithm to overcome the challenges of large datasets and outliers is increasingly becoming the main focus in various research due to the need for more accurate and efficient clustering. One of the newest approaches that is being developed is a combination of CURE with machine learning models, such as using autoencoders to identify complex features and reduce data dimensions before clustering. In this way, the algorithm can remove noise and outliers more effectively, while preserving important information in large datasets. Additionally, research in ACM Transactions on Knowledge Discovery from Data (2022) [3] shows that integration between CURE and density-based models, such as DBSCAN, produces better clustering results on high-density data, where outliers can cause significant distortion. This approach helps CURE group data more precisely in dense areas, while isolating outliers. Furthermore, experiments on large-scale datasets show that implementing CURE in distributed computing environments, such as Apache Hadoop and Spark, allows these algorithms to handle very large datasets more quickly without sacrificing accuracy. The use of a platform like this also allows the development of more adaptive CURE algorithms, for example by automating the selection

parameters of point representations and compression measures, which are instrumental in avoiding bias from outliers.

Several recent studies have made significant contributions to the development of the CURE algorithm to improve its performance in handling large datasets and outliers problems. Research in IEEE Transactions on Big Data (2021) [26] [4] developed CURE by leveraging Apache Spark, which speeds up processing of large datasets and reduces the impact of outliers on clustering results. On the other hand, a study in ACM Transactions on Knowledge Discovery from Data (2022) [3] combined CURE with density-based DBSCAN to pre-separate outliers, which proved effective for anomaly detection in network data. Additionally, research in the Journal of Cleaner Production (2021) [2] introduces a hybrid CURE and Isolation Forest approach for handling high-dimensional data, thereby increasing precision in customer segmentation and fraud detection. Dimensionality reduction techniques using PCA before applying CURE, as proposed in Data Mining and Knowledge Discovery (2021) [5], also help simplify the data and reduce the effects of outliers, increasing clustering efficiency. Meanwhile, research in Information Sciences (2019) [6] developed an adaptive version of CURE for distributed computing environments, such as Hadoop and Spark, with optimized parameters to be more robust to outliers in large datasets. These studies emphasize the importance of developing CURE to be more efficient and accurate in real applications on large and complex data.

To address these limitations, this research introduces CURE-SNE, an enhanced version of CURE that integrates Stochastic Neighbor Embedding (SNE) for dimensionality reduction. SNE optimizes the mapping of high-dimensional data into a lower-dimensional space, effectively preserving local and global structures. This integration allows CURE-SNE to detect complex patterns and improve the identification of outliers, resulting in more accurate clustering results. By leveraging SNE's ability to emphasize neighborhood relationships, CURE-SNE enhances CURE's robustness and accuracy in clustering and outlier detection.

This paper is organized as follows:

- Section II discusses related work and advancements in CURE-based clustering methods.
- Section III describes the proposed CURE-SNE methodology and its implementation.
- Section IV presents experimental results, evaluated using *Silhouette Score* and *Davies-Bouldin Index* to assess clustering quality.
- Section V provides a discussion on the comparative analysis between CURE and CURE-SNE.
- Section VI concludes the study, emphasizing CURE-SNE's contributions to clustering accuracy and outlier detection.

The integration of SNE into the CURE algorithm represents a significant step toward improving clustering performance, particularly for datasets with complex structures and a high number of outliers.

II. Related Work

In recent years, clustering research has made significant strides in addressing critical challenges, particularly in dealing with the increasing complexity of modern datasets. Key issues such as handling large-scale datasets, managing uneven data distributions, and mitigating the disruptive impact of outliers have been at the forefront of this research. Among the most robust and widely recognized clustering algorithms designed to tackle these challenges is the CURE algorithm, short for Clustering Using Representatives. CURE stands out for its ability to effectively handle non-spherical cluster shapes and exhibit strong resistance to outliers. However, despite its robustness, CURE is not without its limitations. One of its primary challenges lies in detecting subtle, complex, and high-dimensional outlier patterns, which can significantly distort clustering outcomes if not adequately addressed.

To overcome these limitations, researchers have proposed and implemented numerous enhancements and hybrid adaptations of the CURE algorithm. For instance, the integration of CURE with Gaussian Mixture Models (GMM) has been shown to greatly improve the detection and representation of outlier patterns in high-dimensional datasets [7]. This hybrid approach has proven particularly effective in domains where data complexity is a significant factor. Similarly, the combination of CURE with Support Vector Data Description (SVDD) has resulted in a highly effective anomaly detection framework specifically tailored for network data analysis [8]. This framework capitalizes on the strengths of both density-based clustering and boundary-based detection techniques to improve the reliability of clustering results in such specialized domains.

Another major challenge associated with clustering large datasets is their high dimensionality, which can complicate data representation and analysis. To address this issue, dimensionality reduction techniques have been successfully employed. For example, research has demonstrated that combining CURE with Principal Component Analysis (PCA) not only reduces the dimensionality of the data but also preserves critical information essential for accurate clustering [9]. This combination enhances the clustering algorithm's performance and ensures better scalability for handling extensive datasets. Additionally, recent advancements have showcased the implementation of CURE on the Apache Flink framework, which is a powerful distributed computing system. This adaptation has resulted in a 45% improvement in processing speed compared to traditional methods, while maintaining sensitivity to outliers, making it an ideal solution for real-time and large-scale data processing needs [10].

The integration of deep learning with clustering algorithms has also opened up new opportunities for innovation. Hybrid models that combine CURE with deep learning techniques, such as autoencoders, have significantly enhanced the algorithm's ability to detect complex, nonlinear relationships in high-dimensional data. For example, in the context of e-commerce applications, this combination has led to remarkable improvements in clustering precision and the detection of subtle outlier patterns [13]. Similarly, hybrid approaches like CURE-DBSCAN [11] and CURE-SNE [12] leverage density-based

clustering and dimensionality reduction techniques, respectively, to handle datasets with intricate structures. These advancements underscore the adaptability, precision, and efficiency of modern clustering algorithms in addressing the challenges posed by diverse and complex datasets.

These developments are not only theoretical but also have practical implications across various fields, including health data analysis and decision-making systems. For instance, Ginting et al. (2023) [28] explored the application of fuzzy logic methods to predict neurotic disorder types. This study emphasized the critical role of precision in computational methods when dealing with sensitive medical data. Similarly, Ginting et al. (2024) [29] developed a perceptron neural network model for predicting postpartum depression. This research demonstrated the significant potential of hybrid and advanced computational techniques in addressing public health challenges and improving the accuracy of predictive analytics.

Moreover, in the domain of decision-making systems, Ginting et al. (2021) [30] introduced an innovative integration of the AHP and TOPSIS methods. This approach optimized the performance of decision support systems for identifying recipients of the Family Hope Program. This combination of multi-criteria decision-making techniques aligns closely with the broader theme of leveraging diverse computational methods to achieve optimal outcomes in complex and multi-dimensional datasets. Such methodologies not only enhance the accuracy of decision-making systems but also ensure scalability and reliability across various application areas.

In summary, the continuous evolution of the CURE algorithm and its hybrid adaptations reflects the growing demand for advanced clustering techniques capable of addressing the ever-increasing complexity of real-world datasets. These innovations have proven to be invaluable tools across multiple domains, providing practical solutions for challenges ranging from network security and anomaly detection to large-scale public health analysis and market segmentation.

III. METHODOLOGY

CURE (Clustering Using Representatives) algorithm is a clustering algorithm designed to handle large datasets and is able to work with data that has a non-spherical cluster shape, while reducing the impact of outliers. [14] One of the main advantages of CURE is its unique approach of representing each cluster with several representative points carefully selected from the data, rather than just one central point as in K-Means. [15] CURE has the advantage of generating clusters of various shapes and sizes, which makes it very effective in the analysis of complex data, such as spatial data or data that is not symmetrically distributed. By using these representative points, CURE maintains flexibility in grouping data that does not conform to simple distribution assumptions, thereby providing more robust clustering results [16].

The main steps in the CURE algorithm consist of several key stages, namely sampling, initial cluster formation, selection of representative points, and compression of representative points towards the cluster center. First, the algorithm samples the data to reduce the number of points that need to be processed, thereby speeding up computing. After that, the sampled data is grouped into initial clusters using a hierarchical clustering approach. [25] Next, for each cluster formed, the algorithm selects a number of representative points (usually several points in the area around the cluster) which will be used to describe the characteristics of the cluster. These representative points were chosen to define the shape and boundaries of the clusters more clearly, including clusters that have asymmetrical shapes [17].

To be more resistant to outliers, CURE applies a compression technique (shrinkage) to each representative point, namely shifting these points towards the cluster center by a certain factor [19]. Suppose C is a cluster with a center of mass μ and a representative point Ri (with i referring to the ith representative point in cluster C), then each point Ri is compressed towards the center μ using a shrinkage factor α which satisfies $0< \alpha <1$. The formula for moving the representative point Ri to Ri' is as given in Eq. (1):

$$Ri' = \mu + \alpha (Ri - \mu) \quad (1)$$

Ri' : New representative point after compression.

μ : The cluster center point (centroid) of cluster C.

Ri : The initial representative point selected for the cluster C.

α : Compression factor, where $0<\alpha<1$.

The parameter α controls how far the representative points will be compressed towards the cluster center. For example, a value of $\alpha = 0.5$ means that each representative point is shifted towards the center by 50% of its distance to the center. Here, the value of α plays an important role in determining how far the representative point moves towards the center. The larger the α value, the greater the influence of the cluster center on the representative point, which reduces the effect of outliers on the cluster. By performing shrinkage, the CURE algorithm reduces the influence of outliers located far from the cluster center, thereby increasing cluster stability and producing more accurate results [27].

Distance Measurement between Clusters (Hierarchical Clustering): In the CURE algorithm, clusters are initially generated through a hierarchical clustering approach. To combine two clusters, CURE measures the distance between two clusters Ci and Cj based on the closest representative point in each cluster. For example, if the representative points of cluster Ci are $\{ri1, ri2, \ldots, rim\}$ and of cluster Cj are $\{rj1, rj2, \ldots, rjn\}$, then the distance between clusters is calculated as in Eq. (2):

$$d(Ci, Cj) = \min_{p \in Ci, q \in Cj} \|rp - rq\| \quad (2)$$

$d(Ci, Cj)$: Distance between clusters Ci and Cj.

rp : Representative point in cluster Ci.

rq : Representative point in cluster Cj.

$\|rp - rq\|$: Euclidean distance between rp and rq.

This distance measure determines how close two clusters are to each other, so CURE can decide whether two clusters should be combined.

Centroid or cluster center for representative point compression, CURE calculates cluster centers using centroids. If cluster C has data points {x1,x2,…,xn}, then the cluster center μ can be calculated using the Eq. (3):

$$\mu = \frac{1}{n}\sum_{k=1}^{n} Xk \quad (3)$$

μ : Centroid or cluster center *C.*

Xk : *K* data point in the cluster *C.*

N : Number of data points in the cluster *C.*

These cluster centers are used to determine the direction and compression level of representative points.

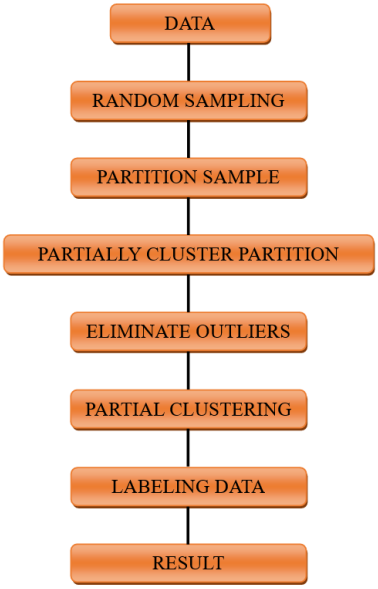


Fig. 1. CURE algorithm.

Fig.1 is the pseudocode for the CURE algorithm.

TABLE I. CURE ALGORITHM FOR CLUSTERING

Step	Description
1	Choose representative points for each cluster
2	For each point, calculate the distance to the representative points.
3	Cluster points based on the minimum distance to the representatives.
4	Repeat steps 2 and 3 until convergence or a stopping criterion is met.
5	Remove outliers: points that are far from any cluster representatives.
6	Output the final cluster with their representative points.

In Table I, summarizes the key steps involved in the CURE algorithm for clustering, focusing on the selection of representative points and the process of refining the clusters while detecting outliers.

A. Cure-SNE

CURE-SNE (Clustering Using Representative Objects with Stochastic Neighbor Embedding) is an advanced clustering algorithm that combines the strengths of CURE with the dimensionality reduction technique of Stochastic Neighbor Embedding (SNE) [18]. While CURE focuses on selecting representative points to form clusters, CURE-SNE enhances this process by first mapping the data into a lower-dimensional space using SNE. This transformation helps reveal complex patterns and relationships that may not be apparent in higher-dimensional spaces [20]. In CURE-SNE, the clustering is performed by calculating the distance between data points and the representative points in this reduced space, making the algorithm more sensitive to underlying structures. One of the key advantages of CURE-SNE is its ability to detect and handle outliers more effectively. By identifying points that are far from any cluster representatives in the low-dimensional space, CURE-SNE ensures that these outliers are excluded from the final clusters, resulting in more accurate and refined clustering outcomes. This hybrid approach makes CURE-SNE particularly useful for datasets with complex structures or a high number of outliers.

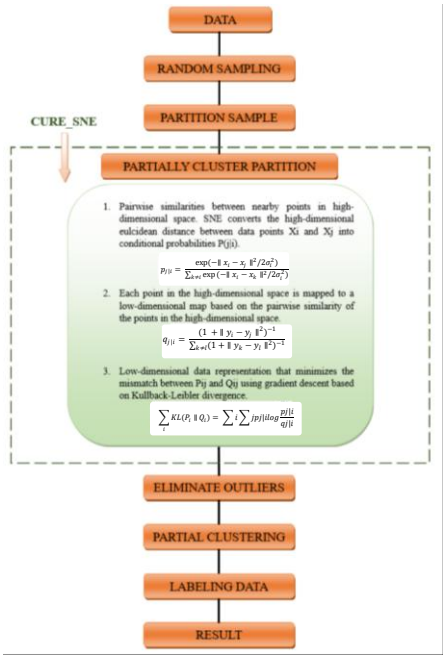


Fig. 2. CURE-SNE algorithm.

In Fig. 2, outlines the key steps involved in the CURE-SNE algorithm, emphasizing the integration of SNE to map the data to a lower-dimensional space and refining the clustering process by detecting outliers effectively.

B. Clustering Evaluation

The clustering evaluation methodology in this research uses two main metrics: Silhouette Score and Davies-Bouldin Index (DBI). The Silhouette Score is used to measure the extent to which each data point is separated from other clusters, with values ranging from -1 to 1. A positive value close to 1 indicates that the data point is well located in the right cluster, while a value close to -1 indicates that the closer to the wrong cluster. This process is carried out by calculating the average distance between each data point to other points in the same cluster, as well as the average distance to the closest point in another cluster. Meanwhile, DBI evaluates the quality of clustering by comparing the distance between clusters with the size of the cluster itself. Lower DBI indicates better separation between

clusters and higher compactness. These two metrics provide a comprehensive picture of the effectiveness of the applied clustering algorithm, thereby allowing the selection of the optimal clustering model based on the structure of the analyzed data.

1) Silhouette score: Silhouette Score is used to measure the extent to which each data point is separated from other clusters. [21] The formula for calculating the Silhouette Score $S(i)$ for data point i is as given in Eq. (4):

$$S(i) = \frac{b(i)-a(i)}{max(a(i),b(i))} \quad (4)$$

where, $a(i)$ is the average distance between point i and all other points in the same cluster, and $b(i)$ is the average distance between point i and the nearest point in another cluster. The Silhouette Score value ranges from -1 to 1; a positive value close to 1 indicates that the data point is well located in the correct cluster, while a value close to -1 indicates that the point is closer to the incorrect cluster. [22]

2) Davies-Bouldin Index (DBI): The Davies-Bouldin Index (DBI) evaluates the quality of clustering by comparing the distance between clusters with the size of the cluster itself. [23] The DBI formula for C cluster is given in Eq. (5):

$$DBI = \frac{1}{C}\sum_{i-1}^{C} \max(\frac{Si+Sj)}{Dij}) \quad (5)$$

where, Si is the size (in terms of distance) of cluster i, Sj is the size of cluster j, and Dij is the distance between the center of cluster i and the center of cluster j [24]. Lower DBI indicates better separation between clusters and higher compactness. These two metrics provide a comprehensive picture of the effectiveness of the applied clustering algorithm, thereby allowing the selection of the optimal clustering model based on the structure of the analyzed data.

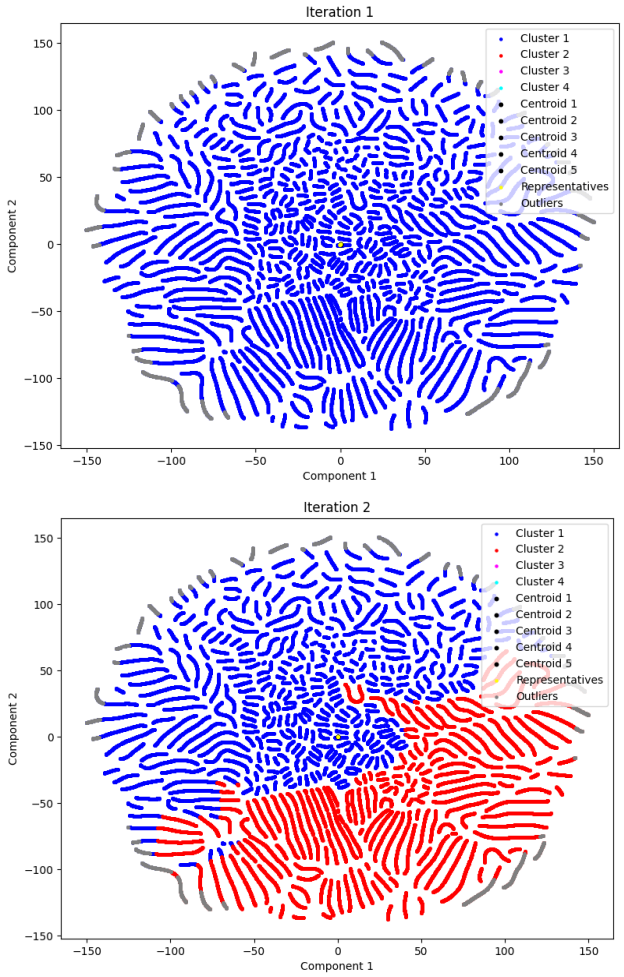
IV. Experimental Results and Analysis

The dataset under analysis contains data related to stunting cases, comprising 121,000 rows, with several key variables essential for health analysis, including age (in months or years), gender (male or female), height (in centimeters), weight (in kilograms), and nutritional status (categories indicating nutritional conditions such as well-nourished, undernourished, or stunted). The aim of analyzing this dataset is to understand the patterns and factors associated with the occurrence of stunting, which can assist in formulating more effective interventions or policies to address this public health issue. To ensure accurate and reliable analysis, outlier detection and handling will be performed on the data. Outliers, which may appear as extreme or anomalous values in certain variables, can significantly impact the results of the analysis. By identifying and addressing outliers, we can ensure that the dataset maintains high quality, allowing for more valid insights into the patterns and factors influencing stunting.

In Table II, a visualization of the clustering results using the CURE (Clustering Using Representatives) model. This model group's data based on representative points that capture the characteristics of each cluster, employing an approach that identifies patterns in the data and handles variations in distribution. The image shows several iterations, resulting in clusters with different characteristics, with each cluster represented by a different color, indicating groups of data with similarities in the analyzed variables. This visualization helps in understanding the distribution and patterns within the cluster space generated by the CURE algorithm.

TABLE II. Table Datasets

No	Age (month)	Gender	Height (cm)	Weight (kg)	Nutritional Status
1	18	Boy	80.5	10.1	Stunted
2	23	Girl	101.2	16	Over
3	18	Boy	74.1	7.2	Severely Stunted
4	30	Boy	102	16.4	Over
5	8	Boy	76.1	8.0	Normal
6	2	Boy	52.0	3.3	Normal
7	32	Boy	101.9	17.6	Over
8	24	Girl	100.0	15.1	Over
9	50	Boy	112.2	21.0	Over
10	18	Boy	75.7	8.2	Severely Stunted
...	...	...	...	...	...
120.999	5	Boy	50.0	4.1	Severely Stunted



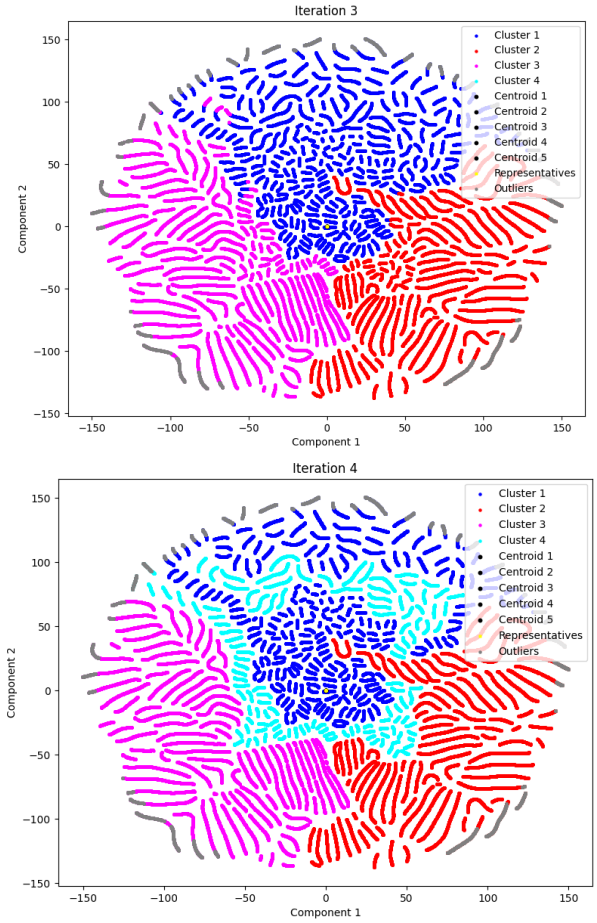


Fig. 3. Visualization of clustering iteration with CURE.

Fig. 3 is a visualization of the clustering results using the CURE-SNE (Clustering Using Representatives and Stochastic Neighbor Embedding) model. This model groups data based on the proximity between points, employing an approach that effectively handles outliers. The image shows four iterations, resulting in four distinct clusters, with each cluster represented by a different color, indicating groups of data with similar characteristics based on the analyzed variables. This visualization aids in understanding the distribution and patterns within the cluster space generated by the algorithm.

In Fig. 4:

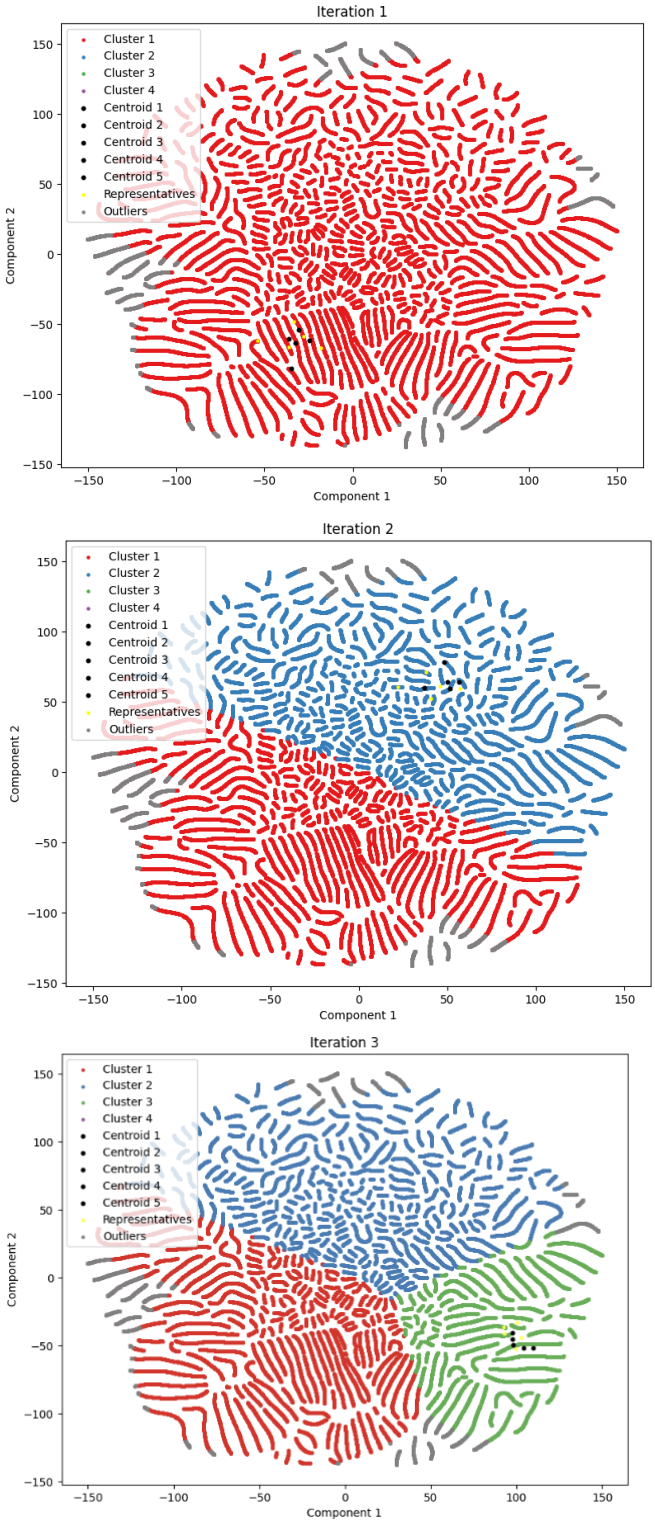
Iteration 1: In the first iteration, the CURE-SNE algorithm initializes 121K clusters based on the data, just as in the original dataset. Each data point represents a separate cluster.

Iteration 2: In the second iteration, the CURE-SNE algorithm begins merging nearby clusters. Clusters that are close to each other in the SNE visualization tend to share similar characteristics in the original data, leading them to merge into larger clusters.

Iteration 3: The merging of clusters continues in the next iteration. Clusters that exhibit similar patterns in the SNE space, as indicated by their proximity in the visualization, continue to merge. The size of the dominant clusters increases, while smaller clusters or outliers remain separate.

Iteration 4: In the final iteration, the CURE-SNE algorithm reaches the desired number of clusters, which is 4 clusters.

The following images show the clustering results from both CURE and CURE-SNE, highlighting the differences in the distribution of points for each cluster.



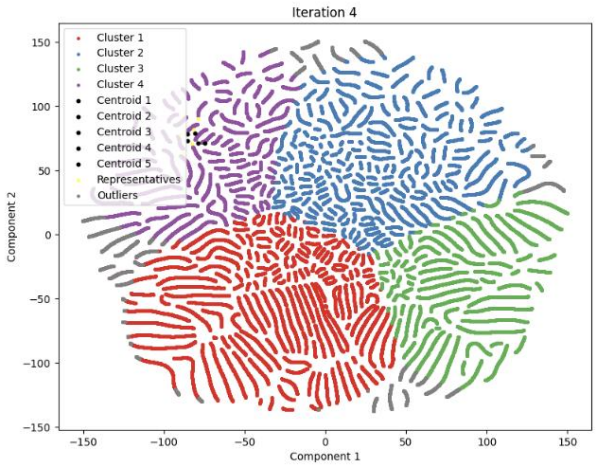


Fig. 4. Visualization of clustering iteration with CURE-SNE.

From Fig. 5 and Fig. 6, these four clusters represent groups of toddlers with distinct characteristics, as described below:

Cluster 1: Male toddlers with a very stunted nutritional status.

Cluster 2: Female toddlers with a normal nutritional status.

Cluster 3: Male toddlers with a normal nutritional status.

Cluster 4: Male toddlers with a high nutritional status.

Table III presents the composition of the clustering results for nutritional status using two different approaches: CURE and CURE-SNE. In the CURE method, the data is grouped based on representative points to capture the cluster structure, while CURE-SNE combines the outlier-handling capabilities of CURE with dimensionality reduction through Stochastic Neighbor Embedding (SNE). Each table displays the number of individuals in each cluster, along with their distribution across nutritional status categories such as well-nourished, undernourished, and stunted, providing valuable insights into the patterns and characteristics within the data.

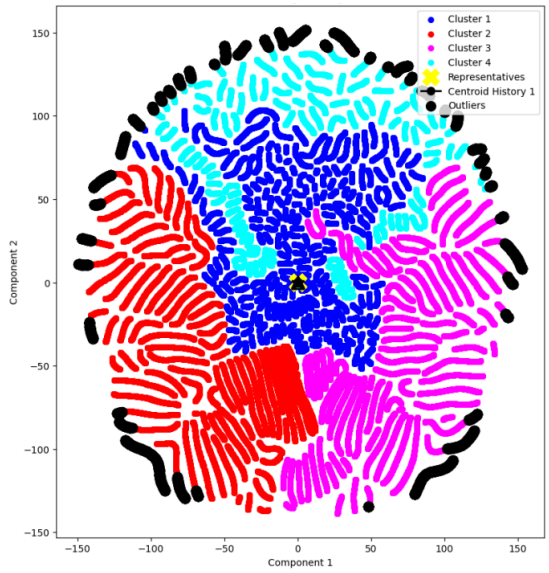


Fig. 5. Clustering results using the CURE.

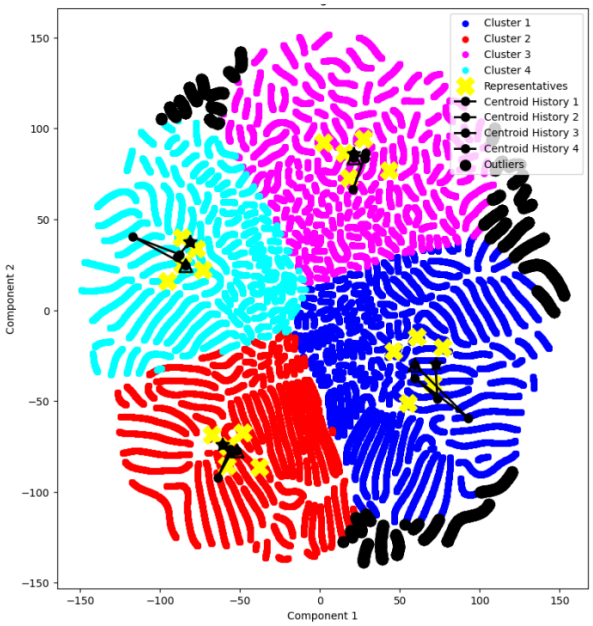


Fig. 6. Clustering results using the CURE-SNE.

TABLE III. CLUSTERING COMPOSITION OF NUTRITIONAL STATUS WITH CURE (%)

Nutritional Status/Cluster	Normal	Severely Stunted	Stunted	Over
1	72.855	1.112	13.015	13.018
2	22.683	32.860	17.248	27.209
3	43.932	29.154	9.078	17.934
4	100.000000	0.000000	0.000000	0.000000

TABLE IV. CLUSTERING COMPOSITION OF NUTRITIONAL STATUS WITH CURE-SNE (%)

Nutritional Status/Cluster	Normal	Severely Stunted	Stunted	Over
1	46.811	3.436	35.665	14.018
2	37.017	24.788	23.076	15.119
3	38.390	29.199	10.078	22.766
4	100.000000	0.000000	0.000000	0.000000

From Table IV, the clustering results using the CURE and CURE-SNE algorithms demonstrate the capabilities of both methods in grouping data while detecting outliers, though with differences in the number of outliers identified. The CURE algorithm effectively clusters data based on representative points within each cluster and detects outliers using a distance-based approach. On the other hand, the CURE-SNE algorithm, which integrates Stochastic Neighbor Embedding (SNE) to optimize the mapping of data in a lower-dimensional space, is able to detect a greater number of outliers compared to CURE. This indicates that CURE-SNE is more sensitive in identifying data points that do not conform to general patterns, resulting in a more detailed clustering outcome.

Fig. 7 and Fig. 8 shows the outlier detection process by CURE, where a total of 5,047 outlier data points were identified. In comparison, CURE-SNE detected 6,050 outliers.

Outliers in data_CURE:

	Age (month)	Gender	Height (cm)	Nutritional Status
12024	6	male	60.6	severely stunted
12028	6	male	59.1	severely stunted
12031	6	male	60.4	severely stunted
12038	6	male	60.7	severely stunted
12041	6	male	59.4	severely stunted
...	...	...	...	...
86829	43	female	111.5	normal
86836	43	female	111.4	normal
86889	43	female	111.4	normal
86929	43	female	111.6	normal
86948	43	female	111.2	normal

[5047 rows x 5 columns]

Number of outliers in original data: 5047

Fig. 7. Outliers detection by CURE.

Outliers in data_CURE-SNE:

	TSNE1	TSNE2
12016	10.998404	134.536285
12022	8.577623	138.188461
12024	7.951420	146.315186
12028	7.508828	142.431824
12031	7.873251	145.992142
...	...	...
119898	-5.304644	-136.068207
119909	-5.924666	-135.762711
119931	-5.790865	-135.835434
119953	-5.790865	-135.835434
119965	-6.047623	-135.675613

[6050 rows x 2 columns]

Number of outliers in data: 6050

Fig. 8. Outliers detection by CURE-SNE.

Fig. 9 presents a comparison of the clustering results using the CURE and CURE-SNE algorithms across several iterations. Both graphs illustrate the changes in cluster sizes over iterations and show how the algorithms detect and handle outliers. CURE uses a distance-based approach to group data and detect outliers, while CURE-SNE combines this approach with dimensionality reduction mapping, resulting in more detailed clustering outcomes.

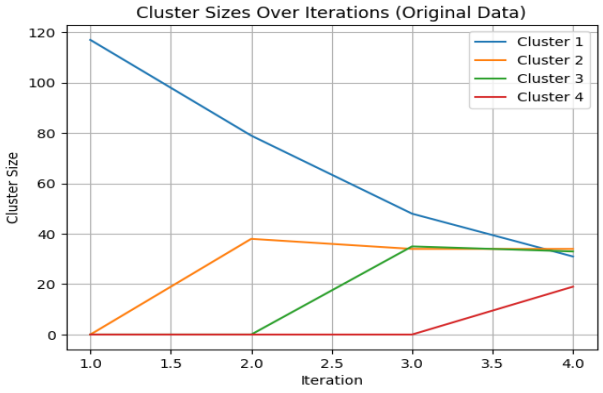


Fig. 9. Graph of CURE.

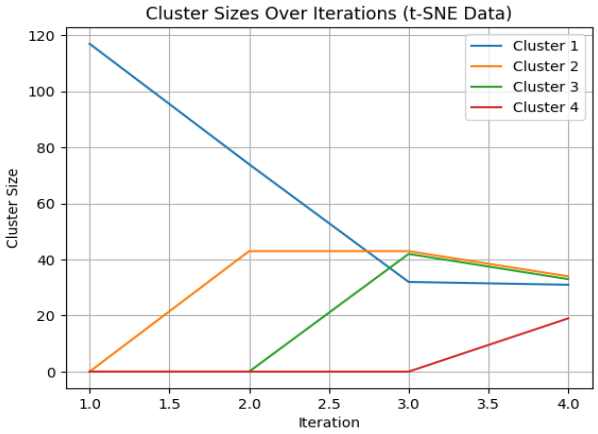


Fig. 10. Graph of CURE-SNE.

Fig. 10 also shows the clustering results using the CURE algorithm on the original data. The initial clusters also decrease in size during the iterations. However, the changes in cluster size are less dramatic, indicating that fewer outliers were detected compared to the CURE-SNE approach.

In the first graph, it can be observed that the CURE-SNE algorithm detects significant changes in cluster sizes over iterations. The initially large clusters gradually break down into smaller clusters, with a clearer data distribution in the final iterations. This indicates CURE-SNE's sensitivity in handling outliers, reflected in the reduction of main cluster sizes and the formation of additional clusters.

The comparison of these two graphs shows that CURE-SNE is generally more effective in detecting outliers and producing clusters with a more segmented data distribution.

It is essential to evaluate the performance of the clustering results generated by both the CURE and CURE-SNE methods to ensure the validity of the cluster structures. In this study, two evaluation metrics are used: the silhouette score, which helps determine the optimal number of clusters by measuring cluster cohesion and separation, and the Davies-Bouldin Index (DBI), which assesses the quality of the clusters by considering their compactness and separation. These evaluations provide valuable insights into the effectiveness of the clustering methods in capturing meaningful patterns in the data.

1) Sillhouette score evaluation: Fig. 11 illustrates the silhouette score analysis for determining the optimal number of clusters generated by both the CURE and CURE-SNE algorithms. The silhouette score, which measures the quality of clustering by assessing the separation and cohesion of clusters, indicates that both methods achieve the highest clustering performance at 4 clusters. As shown, the silhouette score initially increases and peaks at 4 clusters before dropping significantly as the number of clusters increases. This suggests that 4 clusters provide the best balance between intra-cluster cohesion and inter-cluster separation for the given dataset. The similarity in results highlights the effectiveness of both CURE and CURE-SNE in identifying the optimal cluster structure.

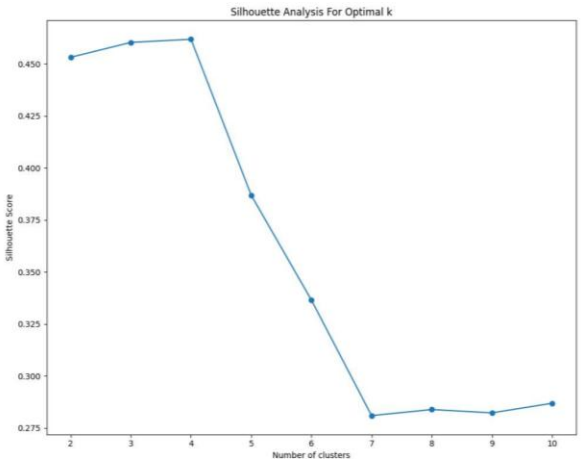


Fig. 11. Graph of sillhouette score evaluation.

2) Davies-Bouldin Index (DBI) Evaluation: From Table V, clustering evaluation can be performed using the Davies-Bouldin Index (DBI), which measures the quality of clusters based on the separation between clusters and the compactness within clusters. The smaller the DBI value, the better the clustering quality, as it indicates well-separated and tightly-knit clusters. In this study, the CURE-SNE method yielded a DBI value of 0.388, which is smaller than the DBI value of 0.523 obtained by the CURE method. This demonstrates that the CURE-SNE method performs better in generating clusters with higher quality, featuring clearer separation between clusters and greater compactness within them.

Evaluation of CURE Clustering:

$R_{12} = 0.314$, $R_{13} = 0.611$, $R_{14} = 0.366$, $R_{23} = 0.431$, $R_{24} = 0.363$, $R_{34} = 0.438$

$D_1 = \max(R_{12}, R_{13}, R_{14}) = \max(0.314, 0.611, 0.366) = 0.611$

$D_2 = \max(R_{21}, R_{23}, R_{24}) = \max(0.314, 0.431, 0.363) = 0.431$

$D_3 = \max(R_{31}, R_{32}, R_{34}) = \max(0.611, 0.431, 0.438) = 0.611$

$D_4 = \max(R_{41}, R_{42}, R_{43}) = \max(0.366, 0.363, 0.438) = 0.438$

$DBI = ¼ (0.611+0.431+0.611+0.438) = 0.523$

Evaluation of CURE-SNE Clustering:

$R_{12} = 0.4$, $R_{13} = 0.203$, $R_{14} = 0.164$, $R_{23} = 0.422$, $R_{24} = 0.272$, $R_{34} = 0.309$

$D_1 = \max(R_{12}, R_{13}, R_{14}) = \max(0.4, 0.203, 0.164) = 0.4$

$D_2 = \max(R_{21}, R_{23}, R_{24}) = \max(0.4, 0.422, 0.272) = 0.422$

$D_3 = \max(R_{31}, R_{32}, R_{34}) = \max(0.203, 0.422, 0.309) = 0.422$

$D_4 = \max(R_{41}, R_{42}, R_{43}) = \max(0.164, 0.272, 0.309) = 0.309$

$DBI = ¼ (0.4+0.422+0.422+0.309) = 0.388$

TABLE V. COMPARISON OF CLUSTER EVALUATION

Algorithm	Sillhouette Score	Davies-Bouldin Index
CURE	4	0.523
CURE-SNE	4	0.388

V. DISCUSSION

The clustering results using the CURE and CURE-SNE algorithms show significant differences in how the two approaches process data and detect outliers. In the CURE algorithm, the clustering process is based on data representation through representative points that reflect the characteristics of each cluster. The results show that this algorithm is effective in grouping data but less sensitive to detecting outliers, with 5047 outliers detected. This results in clusters with more stable data distributions and less drastic changes in size at each iteration.

In contrast, the CURE-SNE algorithm, which combines the CURE representation approach with dimensionality reduction through Stochastic Neighbor Embedding (SNE), exhibits a higher ability to detect outliers. This is evident from the dramatic reduction in the size of large clusters in the early iterations and the formation of smaller, more separated clusters in subsequent iterations. CURE-SNE detected 6050 outliers, which is more than the CURE algorithm. The sensitivity of CURE-SNE in handling outliers makes it more effective at identifying deviating data points, resulting in more segmented clusters.

Overall, both algorithms have their respective advantages. CURE is better suited for clustering data with a more regular distribution and is less influenced by outliers, while CURE-SNE excels in detecting complex patterns and handling data with many outliers. Therefore, the choice of algorithm should be tailored to the characteristics of the dataset and the analysis goals. In this case, the results from CURE-SNE provide deeper insights into the data structure, particularly in the context of identifying significant outliers for further analysis.

VI. CONCLUSION

The clustering analysis results using the CURE and CURE-SNE algorithms provide valuable insights into the capabilities of both methods in grouping data and detecting outliers. Both algorithms resulted in four clusters, but with differences in outlier detection, where CURE-SNE was able to detect 6050 outliers, while CURE detected 5047 outliers. Cluster validation using silhouette score showed that both CURE and CURE-SNE formed four optimal clusters. However, when validated with the Davies-Bouldin Index (DBI), CURE achieved a value of 0.523, while CURE-SNE achieved a value of 0.388, indicating that CURE-SNE outperformed CURE in terms of the quality of the clusters formed. The CURE algorithm demonstrated strong performance in generating stable clusters with well-organized data distributions but had limitations in sensitively detecting outliers. On the other hand, CURE-SNE, with the integration of the Stochastic Neighbor Embedding (SNE) technique, was able to detect more outliers and generate more segmented clusters, reflecting complex patterns within the data. This difference indicates that CURE-SNE is more effective for datasets with irregular distributions or many outliers, while CURE is better suited for data with a more homogeneous structure. Therefore, the choice of algorithm should consider the characteristics of the dataset and the analysis objectives. These findings can serve as a reference for selecting the appropriate clustering method for analyzing complex data, such as public health cases, including identifying factors contributing to stunting.

ACKNOWLEDGMENT

We would like to express our deepest gratitude to the Directorate of Research, Technology, and Community Service (DRTPM) for funding this research under the Doctoral Dissertation Research (PDD) scheme. We also extend our sincere thanks to Universitas Sumatera Utara for providing invaluable support and resources that greatly contributed to the success of this study.

REFERENCES

[1] Anonymous "2021 Index IEEE Transactions on Knowledge and Data Engineering Vol. 33," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, *(1)*, pp. 1-37, 2022.

[2] P. V. Balachandran, "Data-driven design of B20 alloys with targeted magnetic properties guided by machine learning and density functional theory," *Journal of Materials Research*, vol. 35, *(8)*, pp. 890-897, 2020.

[3] C. C. Aggarwal, "Communication from the Editor-in-Chief: State of the ACM Transactions on Knowledge Discovery from Data," *ACM Transactions on Knowledge Discovery from Data*, vol. 16, *(2)*, pp. 1-2, 2022.

[4] Anonymous "IEEE Transactions on Big Data," *IEEE Software*, vol. 38, *(5)*, pp. 22-22, 2021.

[5] J. Plasse, H. Hoeltgebaum and N. M. Adams, "Correction to: Streaming changepoint detection for transition matrices (Data Mining and Knowledge Discovery, (2021), 10.1007/s10618-021-00747-7)," *Data Mining and Knowledge Discovery*, 2021.

[6] Anonymous "Corrigendum to Spam Profiles Detection on Social Networks Using Computational Intelligence Methods: The Effect of The Lingual Context (Journal of Information Science, (2019), 10.1177/0165551519861599)," *Journal of Information Science*, 2019.

[7] A. N. M. B. Rashid *et al*, "Correction to: Cooperative co-evolution for feature selection in Big Data with random feature grouping (Journal of Big Data, (2020), 7, 1, (107), 10.1186/s40537-020-00381-y)," *Journal of Big Data*, vol. 7, *(1)*, 2020.

[8] D. Zhang, Z. Ye, G. Feng and H. Li, "Intelligent Event-Based Fuzzy Dynamic Positioning Control of Nonlinear Unmanned Marine Vehicles Under DoS Attack," in *IEEE Transactions on Cybernetics*, vol. 52, no. 12, pp. 13486-13499, Dec. 2022, doi: 10.1109/TCYB.2021.3128170.

[9] C. Garibotto, A. Sciarrone, F. Lavagetto, L. Pronzati, A. Baljak and G. Tagliabue, "Performance Analysis of an IoT-Based Personal Vocal Assistant for Cruise Ships Over Satellite Networks," in *IEEE Internet of Things Journal*, vol. 9, no. 16, pp. 14857-14866, 15 Aug.15, 2022, doi: 10.1109/JIOT.2021.3116435.

[10] Y. Zhang *et al*, "Research on resource allocation technology in highly trusted environment of edge computing," *Journal of Parallel and Distributed Computing*, vol. 178, pp. 29-42, 2023.

[11] C. Boya-Lara, O. Rivera-Caballero and J. Alfredo Ardila-Rey, "Clustering by communication with local agents for noise and multiple partial Discharges discrimination," *Expert Systems with Applications*, vol. 225, pp. 120067, 2023.

[12] H. Xu, P. Tong and Y. Li, "Correction to: Different treatments of pixels in unlabeled images for semi-supervised sonar image segmentation (International Journal of Machine Learning and Cybernetics, (2024), 15, 2, (637-646), 10.1007/s13042-023-01930-6)," *International Journal of Machine Learning and Cybernetics*, 2024.

[13] Jamei, M., Alkouh, A.B., Karbasi, M. *et al.* Thermo-physical properties estimation of an oil-based hybrid nanofluid: application of a new hybrid neurocomputing approach. *J Therm Anal Calorim* (2024).

[14] V. Bhadra Pratap Singh *et al*, "Hierarchical cluster analysis implementation using the algorithm of clustering using representatives," in 2022, . DOI: 10.1109/GlobConET53749.2022.9872414.

[15] J. Park and M. Choi, "A K-Means Clustering Algorithm to Determine Representative Operational Profiles of a Ship Using AIS Data," *Journal of Marine Science and Engineering*, vol. 10, *(9)*, pp. 1245, 2022.

[16] A. Ghimire, M. Alkurdi and F. Amsaad, "Enhancing hardware trojan security through reference-free clustering using representatives," in 2024, . DOI: 10.1109/VLSID60093.2024.00084.

[17] T. Gu *et al*, "A robust reconstruction method based on local Bayesian estimation combined with CURE clustering," *Information Sciences*, vol. 680, pp. 121132, 2024.

[18] T. He *et al*, "Rolling Bearing Fault Diagnosis Using a Deep Convolutional Autoencoding Network and Improved Gustafson–Kessel Clustering," *Shock and Vibration*, vol. 2020, *(2020)*, pp. 1-17, 2020.

[19] M. Cebecauer *et al*, "Revealing representative day-types in transport networks using traffic data clustering," *Journal of Intelligent Transportation Systems*, vol. 28, *(5)*, pp. 695-718, 2024.

[20] C. Manyfield-Donald, T. A. Kwembe and J. C. Cheng, "A modified clustering using representatives to enhance and optimize tracking and monitoring of maritime traffic in real-time using automatic identification system data," in 2021, . DOI: 10.1109/CSCI54926.2021.00119.

[21] D. Joshi *et al*, "Prediction of sonic log and correlation of lithology by comparing geophysical well log data using machine learning principles," *Geojournal*, vol. 88, *(Suppl 1)*, pp. 47-68, 2023.

[22] B. Rim *et al*, "Semantic cardiac segmentation in chest CT images using K-means clustering and the mathematical morphology method," *Sensors (Basel, Switzerland)*, vol. 21, *(8)*, pp. 2675, 2021.

[23] F. Ros, R. Riad and S. Guillaume, "PDBI: A partitioning Davies-Bouldin index for clustering evaluation," *Neurocomputing (Amsterdam)*, vol. 528, pp. 178-199, 2023.

[24] A. Idrus *et al*, "Distance Analysis Measuring for Clustering using K-Means and Davies Bouldin Index Algorithm," *TEM Journal*, vol. 11, *(4)*, pp. 1871-1876, 2022.

[25] L. Dahlström *et al*, "Identification of representative building archetypes: A novel approach using multi-parameter cluster analysis applied to the Swedish residential building stock," *Energy and Buildings*, vol. 303, pp. 113823, 2024.

[26] D. Kumar *et al*, "A Hybrid Approach to Clustering in Big Data," *IEEE Transactions on Cybernetics*, vol. 46, *(10)*, pp. 2372-2385, 2016.

[27] L. Ma and S. Fan, "CURE-SMOTE algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests," *BMC Bioinformatics*, vol. 18, *(1)*, pp. 169-169, 2017.

[28] D. S. Br Ginting, F. Y. Manik, R. Arrahmi, M. A. A. Saragih, M. D. A. A. Dalimunthe, and M. I. Aldeena, "Performance of Fuzzy Tsukamoto and Fuzzy Sugeno Methods in Predicting Types of Neurotic Disorder," 2023, pp. 194–199.

[29] D. S. Br Ginting, F. Y. Manik, F. N. Nasution, and M. I. Aldeena, "Perceptron neural network model on predicting postpartum depression in the puerperium," in AIP Conf. Proc., vol. 2987, 2024, p. 020038.

[30] D. S. Br Ginting, R. L. Sipahutar, F. Natalida, C. N. Kudadiri, and D. E. R. Purba, "Combination AHP and TOPSIS methods optimizes performance of decision support system for the recipients family hope program in Huta Limbong Padang Sidempuan," in 2021 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA), 2021, doi: 10.1109/DATABIA53375.2021.9650342.