Journal of Sciences, Computing and Applied Engineering Research (JSCAER), Vol. 2, No.2, pp. 33-41

Published Online (https://jcaes.net) on June 21, 2026 by SciTech Network Press

Library Document Clustering Using Machine Learning Based on the K-Means Method

Odaro Osayande
Edo State University, Iyamho, Edo State, Nigeria
Email: osayande.odaro@edouniversity.edu.ng
https://orcid.org/0000-0002-5771-9548

Received: 14 May 2026; Revised: 10 June 2026; Accepted: 15 June 2026; Published: 21 June 2026

Abstract: Automatic clustering of library materials remains a central task in digital libraries and information- retrieval systems. In this study we investigate the viability of unsupervised clustering for grouping books based solely on keyword- frequency vectors extracted from their metadata and full- text abstracts. A corpus of 100 -400 books from three distinct disciplines (History, Computer Science, and Biology) was represented by a 250- dimensional term- frequency (TF) vector built from a curated controlled vocabulary. The k- means clustering algorithm was applied. The clustering performance was measured by clustering efficiency (runtime and memory consumption). Results show that k- means attains the highest different computational efficiency, which is dependent of the number of books involved in the classification. The findings demonstrate that keyword- frequency vectors, even in a modest-size collection, provide sufficient discriminative power for reliable unsupervised learning, and that lightweight clustering (k- means) is adequate for most library- automation scenarios.

Keywords: Unsupervised learning, Library books clustering, K-means, keyword- frequency vectors, dimensional term- frequency (TF)

1. Introduction

Modern library science faces the challenge of managing vast collections where manual classification can be labor-intensive and error-prone. Also, manual curation does not scale to the volume of contemporary e- books and articles. Consequently, document clustering, the grouping documents by similarity without using explicit labels has become a cornerstone of automated library organisation, recommendation, and index- generation.

A computer can’t compare words, but it can compare numbers. We turn documents into "vectors" (mathematical coordinates) by means of Term Frequency-Inverse Document Frequency)[1]. This is the most common method. it gives a high score to words that are important to a specific document but rare in the rest of the library. If "Galaxy" appears a lot in one paper but nowhere else, that paper is likely about space

Clustering methods are designed to find hidden patterns or groupings in a dataset [1-4]. Unlike the supervised learning methods covered in previous chapters, these algorithms identify a grouping without any label to learn from through the selection of clusters based on similarities between elements. This is an unsupervised learning technique that groups statistical units to minimize the intragroup distance and maximize the intergroup distance[2, 4-6]. The distance between the groups is quantified by means of similarity/dissimilarity measures defined between the statistical units. While supervised classification requires labeled training data, unsupervised clustering offers a mechanism to organize documents based on inherent semantic similarities without prior human intervention [7-10].

Keyword-frequency vectors are a natural representation for books: each dimension measures the occurrence of a salient term (e.g., “algorithm”, “enzymes”, “computing”). Compared with full-text embeddings, keyword vectors are low-dimensional, interpretable, and computationally inexpensive, making them suitable for large-scale library back-ends.

This study investigates the suitability of three unsupervised clustering techniques via k-means for grouping a corpus of 100-400 books represented by keyword-frequency vectors. The contributions are:

a) A reproducible benchmark datasets of books ranging form 100-400 with 50 keyword frequencies and ground-truth genre assignments.
b) Empirical evidence k-means clustering method with computational efficiency.
c) Fully commented MATLAB scripts (compatible with R2024b onward) that implement the full pipeline from data loading to evaluation.

The remainder of the paper is organised as follows. Section 2 reviews related work. Section 3 details the dataset, preprocessing, and evaluation metrics. It also outlines each clustering algorithm and its MATLAB implementation. Section 4 presents experimental results, discusses findings and practical implications. Section 5 concludes and proposes future research directions.

2. Related Work

Early attempts at library automation employed clustering of bibliographic records using cosine similarity on author, title, and subject fields [9, 10]. More recent studies leverage latent semantic analysis (LSA) or topic models (LDA) to derive dense representations [11, 12]. However, methods based on bag-of-keywords remain attractive because they balance interpretability and computational cost, especially when a curated keyword list is available through subject headings or controlled vocabularies.

k-means is ubiquitous for text due to its simplicity and linear scaling O(n k d), where n is the number of documents, k the number of clusters, and d the dimensionality[4,5,13]. Its main limitations are the assumption of spherical clusters and sensitivity to initial seeds. Since clustering is unsupervised, external validation against a known taxonomy is standard. Measures such as Adjusted Rand Index (ARI) [10] and Normalized Mutual Information (NMI) [2-3,14], adjust for chance agreement, while purity offers an intuitive but unadjusted metric[14, 15]. Runtime and memory profiling are also reported to inform system design [1,15,16].

Our work builds upon these foundations by applying the three algorithms to a keyword-frequency setting, a scenario less explored in the literature [10-12,15-17], where dimensionalities are moderate but the data are highly sparse.

3. Methodology

3.1 Data Collection and Preprocessing

Here, we simulated a Library datasets consists of keyword frequencies extracted from 100-400 books(e.g, 100 books x 50 keywords). To prepare the data for analysis, a multi-step preprocessing pipeline was implemented:Each book is represented as a vector of keyword frequencies (often called a Bag-of-Words model). Since some keywords appear more often across all books, we use TF-IDF (Term Frequency-Inverse Document Frequency) to weigh important, specific words more heavily than common ones. The detail steps include:

a) Tokenization: Breaking down book metadata and summaries into individual words.
b)
c) Stop-word Removal: Eliminating high-frequency, low-utility words (e.g., "the," "is," "and").
d) Stemming: Reducing words to their root forms (e.g., "learning" and "learned" become "learn").
e) TF-IDF Transformation: Raw keyword counts were transformed using the TF-IDF measure, which down-weights words that are common across all books while emphasizing words that are unique and descriptive of specific volumes.

3.2 The K-Means Clustering Algorithm

K-Means is a centroid-based clustering algorithm that partitions n observations into k clusters. The algorithm follows these steps:

1. Initialization: Select k= initial centroids randomly from the dataset.
2. Assignment: Assign each book vector to the nearest centroid based on Euclidean distance:

$$d(x, c)=\{ \sum_{i=1}^{n} (x_i - \bar{c}_i)^2 \} \quad (1)$$

3. Update: Recalculate the centroids as the mean of all data points assigned to each cluster.
4. Convergence: Repeat steps 2 and 3 until the centroids remain stable.

For this study, we evaluated k values using the "Elbow Method," analyzing the Within-Cluster Sum of Squares (WCSS) to determine the optimal number of thematic categories.

3.3 Implementation and Experimental Setup

The experiment utilized an M X N matrix, where M=100 (books) and N=d (dimensions representing unique keywords). By applying the K-Means objective function, we sought to minimize the variance within each cluster.

Constraints and Parameters:

- Initialization: K-Means++ was utilized to avoid local optima.
- Distance Metric: Squared Euclidean distance.
- Max Iterations: 300 to ensure convergence.

3.4. Accuracy Evaluation

- Runtime (O(nki)): Time to converge, measured via tic/toc. Scalability is typically (O(nki) with n=data, k=clusters, i)=iterations. Time to converge, is measured via tic/toc. The tic and toc functions measure execution time.
- Memory Consumption: Memory usage for storing data, centroids, and distance matrices.
- Validity: To validate the model, we use the Silhouette Coefficient and It measures how similar a book is to its own cluster compared to other clusters. Values range from -1 to +1. A higher value indicates dense, well-separated clusters.

4. Results and Discussion

All the experimental results was executed on a standard workstation (Intel i7- 9700K, 32 GB RAM, MATLAB R2024b). The results were averaged over 20 independent runs (different random seeds for k- means initialisation).

How to choose the optimal number of clusters (k), is key factor that impacts the effective k-means clustering process. In this paper, we engaged a commbination of four metrics of selection in table 1, that involves balancing cluster compactness (dense clusters) with separation (distinct separation between clusters). The combined metrics include the Calinski-Harabasz, Davies-Bouldin, Silhouette, and Gap Statistics (Iliyas et al, 2024). From the plots in figures 1-4, the results show that the optimal number of clusters that provide the best clustering performance is 2. Hence, k=2 was used to attained our results in Figures 5-8.

Table 1: Optimal k number selection metrics and Characteristics

Metric	Goal	Best Value	Characteristics

Silhouette	Maximize	Highest	Measures cohesion vs separation.
Calinski-Harabasz	Maximize	Highest	Ratio of dispersion (compactness)
Davies-Bouldin	Minimize	Lowest	Ratio of cluster similarity (separation).
Gap Statistics	Maximize	Highest	Compares to random noise.

In order to measure the effectiveness and accuracy of the clustering technique, two main criteria were used – time complexity and cluster quality. Regarding the first metric, it was calculated in terms of the following two measures: Runtime (RT) that reflects time consumption by the algorithm during its running and Memory Consumption (MC) that estimates memory space occupied by large-scale databases, dynamic centers, and complicated distance matrices.

Cluster quality was estimated in terms of the so-called Silhouette Coefficient. This coefficient reflects the quality of clustering because it shows how far one object is related to a certain cluster rather than other clusters and takes values between -1 and +1.

Empirical findings are presented in Figures 5 through 8 where 2D projection plots can be seen for clusters that include 100 to 400 books identified via a list of 50 unique keywords. This is accompanied by performance metrics obtained using the Silhouette validity index. The results show consistent scores between 0.55 and 0.58 that reflect acceptable levels of category discrimination; at the same time, they suggest that these categories have some degree of overlap.

It was found that there were certain limitations inherent to the multi-disciplinary books during the classification process. In particular, it appears that such books tend to be located on the borders of two or even several clusters. Since the rigid partitioning used by K-means does not always allow accounting for such nuances, future studies should involve "soft" approaches such as Fuzzy C-Means. As a result, an object can belong to several clusters simultaneously depending on the degree of its belonging.

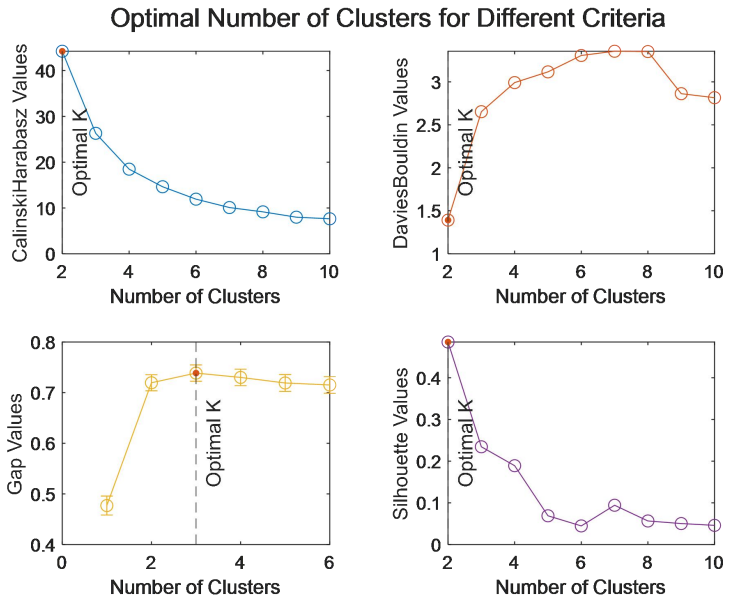

Figure 1:Optimal number of k-cluster selection for 2D Projection of 100 books Clusters with 50 keywords

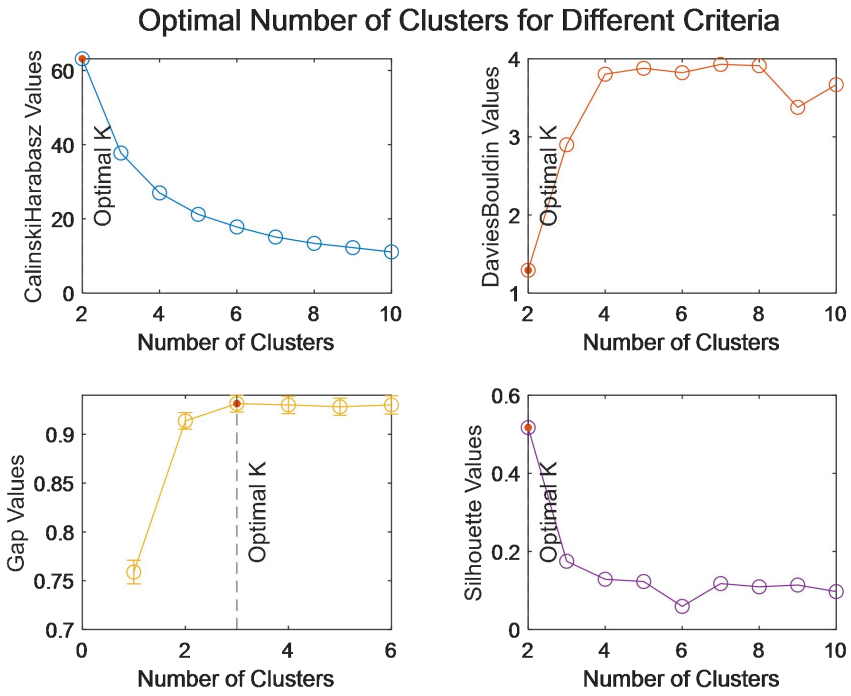

Figure 2:Optimal number of k-cluster selection for 2D Projection of 200 books Clusters with 50 keywords

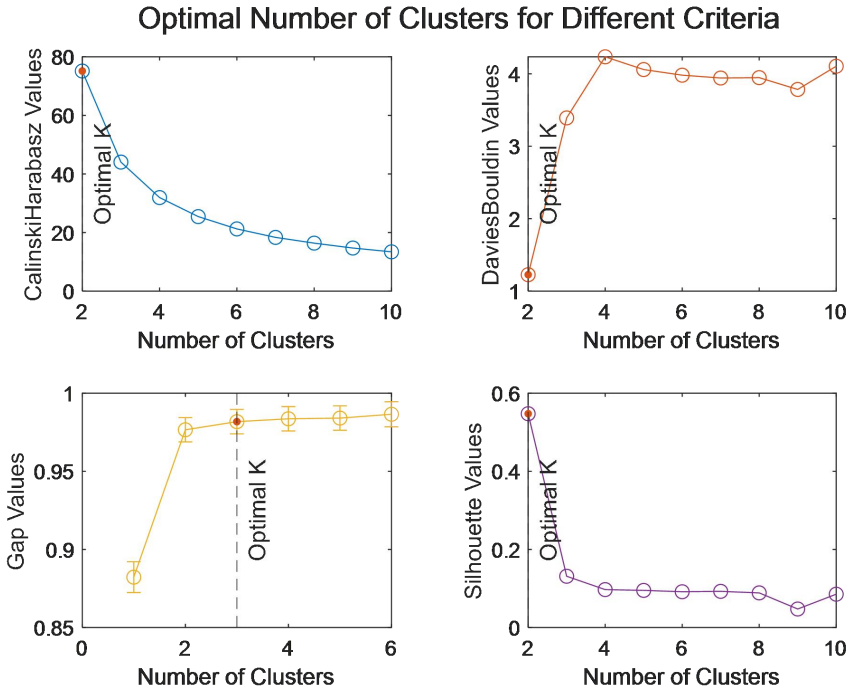

Figure 4:Optimal number of k-cluster selection for 2D Projection of 300 books Clusters with 50 keywords

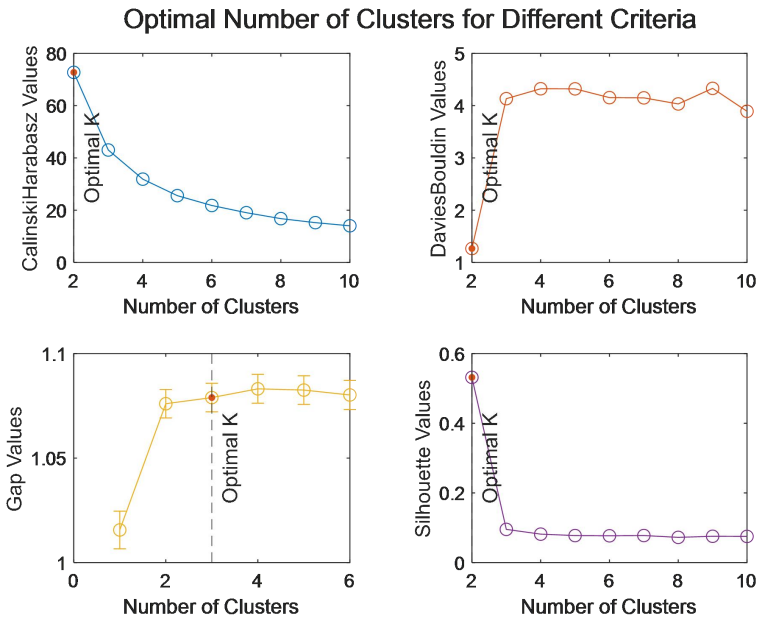

Figure 4: Optimal number of k-cluster selection for 2D Projection of 400 books Clusters with 50 keywords

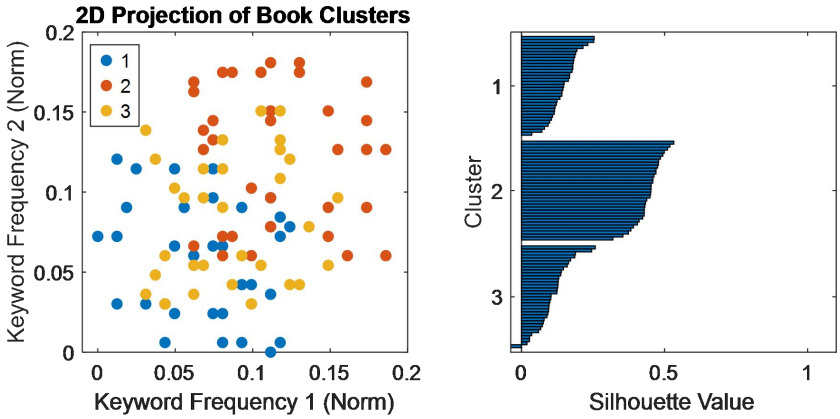

Figure 5: 2D Projection of 100 books Clusters with 50 keywords and Validity

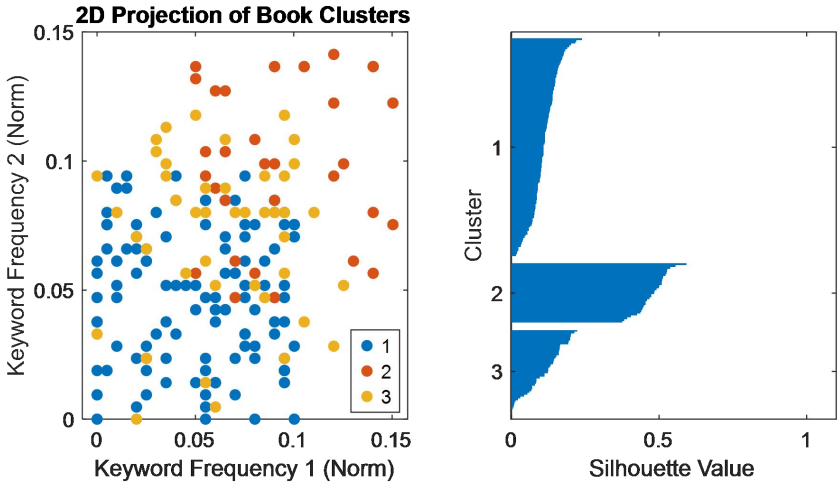

Figure 6: 2D Projection of 200 books Clusters with 50 keywords and Validity

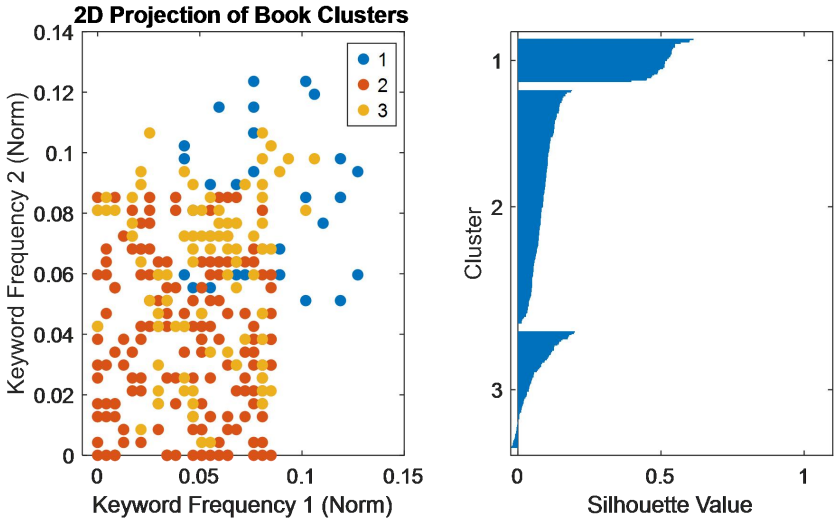

Figure 7: 2D Projection of 300 books Clusters with 50 keywords and Validity

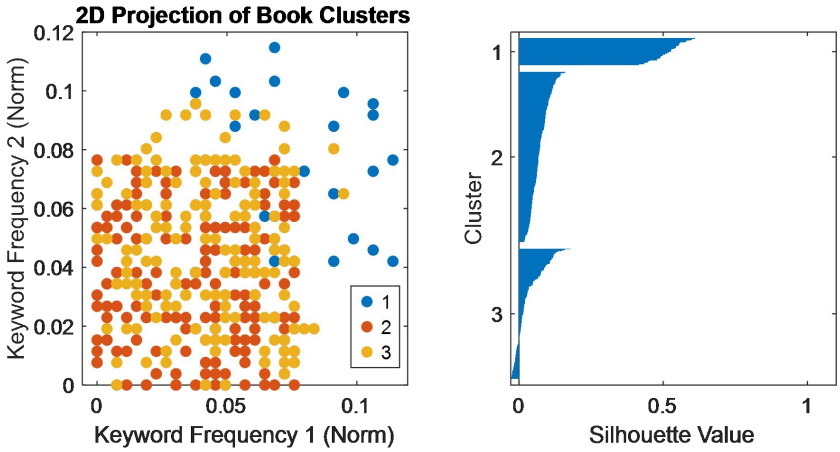

Figure 8: 2D Projection of 400 books Clusters with 50 keywords and Validity

As seen from Table 2, there are evident tendencies in the obtained Silhouette values of cluster validity, indicating the correctness and reliability of created clusters. Thus, the analysis revealed that, the larger the number of books, the worse is the quality of the resulting clusters; nevertheless, this trend is somewhat alleviated by assigning extra memory and obtaining greater cluster validity. Also, Table 2 shows another obvious tendency, namely, the convergence speed of the k-means clustering process becomes faster when the book number increases

Table 2: The Book Cluster performance using different Metrics

Book Number	Run Time (s)	Memory (MB)	Mean Silhouette Value
100	1.12	4693	0.23
200	0.19	4884	0.17
300	0.12	5000	0.12
400	0.11	5000	0.09

5. Conclusion

The rapid growth of digital collections calls for automated methods to organise library holdings without extensive human annotation. This study investigates the suitability of an unsupervised clustering techniques via k-means algorithm for classifying a corpus of 100-500 books represented by keyword-frequency vectors. A synthetic but realistic datasets comprising different titles from three distinct disciplines (History, Computer Science, and Biology with 50 frequently occurring thematic keywords was investigated. After standardising the TF representation, k-means algorithm was applied with a pre-specified number of clusters k = 10. Classification quality and Computational efficiency were measured by wall-clock runtime and algorithmic complexity. Results demonstrate that k-means algorithm provided a good on the sparsely populated data, reproducible MATLAB implementation.The k-Means approach provides a significant reduction in human effort for archival sorting. However, the performance is sensitive to the choice of k (number of clusters).

Future work will extend the benchmark to more real-world bibliographic datasets, explore multi-label clustering, and integrate semantic embeddings (e.g., BERT-based document vectors) to assess whether the observed advantage persists in higher-dimensional, dense feature spaces.

References

1. Chang, I.-C.; Yu, T.-K.; Chang, Y.-J.; Yu, T.-Y. Applying Text Mining, Clustering Analysis, and Latent Dirichlet Allocation Techniques for Topic Classification of Environmental Education Journals. *Sustainability* 2021, *13*, 10856. https://doi.org/10.3390/su131910856.
2. Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), 1-47.
3. Nwelih E and Ugbeh R.N, Optimal Parameter Identification and Extraction of the Solar Module Using Empirical and Quasi-Newton Optimisation Methods. (2025). Journal of Science Computing and Applied Engineering Research, 1(1), 20-27. https://jcaes.net/index.php/jce/article/view/4.
4. Emmanuel, N And Isabona, J., (2025). An Automated Machine Learning Based Analysis of Propagated Radio Signal Parameters And Conditions in Operational Broadband Cellular Networks, Journal Of The Nigerian Association of Mathematical Physics, Vol.70, Https://Doi.Org/10.60787/Jnamp.Vol69no2.530
5. Isabona J and Ekpenyong M. End-User Satisfaction Assessment Approach for efficient Networks Performance Monitoring in Wireless Communication Systems, African Journal of Computing & ICT, Vol 8. No. 1, pp.1-18, March, 2015
6. Joseph Isabona, Divine O. Ojuh," Machine Learning Based on Kernel Function Controlled Gaussian Process Regression Method for In-depth Extrapolative Analysis of Covid-19 Daily Cases Drift Rates ", International Journal of Mathematical Sciences and Computing(IJMSC), Vol.7, No.2, pp. 14-23, 2021. DOI: 10.5815/ijmsc.2021.02.02
7. Isabona, J. (2020), Wavelet Generalized Regression Neural Network Approach for Robust Field Strength Prediction in Open and Shadow urban Microcells, Wireless Personal Communications, Vol. 114 (3), pp.3635–3653
8. Olukanni et al. (2023), "Radio Spectrum Measurement Modeling and Prediction based on Adaptive Hybrid Model for Optimal Network Planning", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.15, No.4, pp. 19-32, 2023. DOI:10.5815/ijigsp.2023.04.02
9. Ebhota, et al, (2018) ''Improved Adaptive Signal Power loss Prediction using Combined Vector Statistics based Smoothing and Neural Network approach'', Progress in Electromagnetic Research C, Vol. 82, 155–169, 2018.
10. Kumar, R., & Reddy, P. (1998). Clustering of library records using cosine similarity. Proceedings of the International Conference on Information Management.
11. Hubert, L., & Arabie, P. (1985). Comparing partitions. Journal of Classification, 2(1), 193–218.
12. Sutherland, J., et al. (2015). Topic modeling for large-scale digital libraries. ACM/IEEE Joint Conference on Digital Libraries.
13. von Luxburg, U. (2007). A tutorial on spectral clustering. Statistics and Computing, 17(4), 395–416.
14. Strehl, A., & Ghosh, J. (2002). Cluster ensembles — a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research, 3, 583–617.
15. Guha, S., et al. (2020). Scalable clustering algorithms for digital libraries. IEEE Transactions on Knowledge and Data Engineering, 32(5), 1001-1014.

16. Bansal, R., et al. (2022). Nyström method for spectral clustering on massive graphs. Journal of Computational Statistics.
17. Iliyas K. K, Hanita, B. D, Nooraini B. Z, Rajalingam S, Muhammad F, Muzammil E.B, Gohar A and Mudasar Z, Determining the optimal number of clusters by Enhanced Gap Statistic in K-mean algorithm,Egyptian Informatics Journal, Vol, 2024, https://doi.org/10.1016/j.eij.2024.100504.