

Web Usage Mining Using Clustering Algorithms (Case Study of LAUTECH Students)

Adeniran Afeez A., Baale A A., Ganiyu R. A., Abdulsalami B. A.,

Ladoke Akintola University of Technology, Ogbomoso,

DOI: <https://doi.org/10.51584/IJRIAS.2025.100800073>

Received: 06 August 2025; Accepted: 14 August 2025; Published: 12 September 2025

ABSTRACT

This study explores the application of Web Usage Mining through clustering algorithms to analyze student web interactions at Ladoke Akintola University of Technology (LAUTECH), Ogbomoso, Nigeria. Leveraging server log data collected over four weeks, the research implements K-means, DBSCAN, Agglomerative Clustering, and Self-Organizing Maps (SOM) to identify user behavior patterns, including login frequencies, session durations, and platform preferences. Preprocessing steps such as log parsing, noise removal, and session identification were critical to structuring raw data for analysis. Results revealed three distinct user clusters (frequent, moderate, and infrequent users) via K-means and Agglomerative methods, while DBSCAN highlighted noise (16 outliers) and SOM provided granular spatial insights. The Key findings include Android dominance (70–80% of users) across clusters, K-means achieved the most actionable segmentation (Silhouette Score: 0.43); and DBSCAN excelled in noise detection (Silhouette Score: 0.86) but failed to form clusters. The study demonstrates web usage mining's potential to optimize institutional web services and personalize user experiences. Challenges like data sparsity and computational complexity are noted, with recommendations for future work on real-time analytics and cross-institutional comparisons.

Keywords: Web Mining, Clustering, Algorithm, Server Log Analysis, Education.

INTRODUCTION

The World Wide Web (WWW) has evolved into a central hub for information dissemination, communication, education, and business intelligence. It hosts a vast repertoire of data generated through user interactions, responses, and content contributions from website owners. As a result, the web has become a powerful means of learning and engagement, particularly within the education sector where service providers increasingly seek analytical insights to understand student interests across regions and disciplines (Aithal et al., 2024).

In response to this data-rich environment, Data Mining (DM) has emerged as a cornerstone of modern software development and academic research (Xu et al., 2019). A vital offshoot of DM is web mining, which applies intelligent algorithms and techniques to extract meaningful patterns from web data that exist in structured, semi-structured, and unstructured formats (Kumar et al., 2022). By transforming human-readable content into machine-understandable semantics, web mining enables organizations to derive actionable knowledge from otherwise raw and disparate web resources (Diop et al., 2025).

Among the various branches of web mining, Web Usage Mining (WUM) has gained prominence for its capacity to analyze user behavior captured in web server logs. These logs record user activities, requests, and navigation paths across websites, serving as a rich data source for uncovering trends, predicting user interests, and improving user experience (Mehrotra and Kohli, 2016; Bahareh et al., 2022). Unlike traditional hit counters, web usage mining offers deep insights into visitor origins, frequency, preferences, and engagement, thereby supporting business decision-making and strategic planning (Kumar et al., 2022).

Web Usage Mining operates through a three-phase process. These are data preprocessing, data mining, and pattern analysis. During preprocessing, tasks such as data cleaning, user identification, and session segmentation are carried out to prepare the log data for mining (Alasali et al., 2024). These sessions, defined by sequences of

user requests, can be identified through time-based or navigation-based methods and are particularly useful when implemented on distributed systems using MapReduce, a framework known for its scalability and fault tolerance (Patidar et al., 2024).

Server log files especially those maintained by educational institutions provide invaluable insights into user behavior. These files record every access and activity on a website, enabling the extraction of useful behavioral patterns and trends. Such analysis supports a range of applications from personalized learning experiences to system optimization and strategic improvements in content delivery (Pacifico and Ludermit, 2019; Wolhuter, 2021).

Modern technologies, including the Hadoop Distributed File System (HDFS), Pig Latin, and tools like Sematext Logs, further enhance the ability to analyze large-scale web logs efficiently (Behera et al., 2022). These systems facilitate the transformation of unstructured web log data into structured formats that are easier to mine for useful patterns.

In academic environments, such as Ladoke Akintola University of Technology (LAUTECH), WUM holds immense potential to improve user experience and service delivery. By applying clustering algorithms to weblog records, it becomes possible to understand user engagement patterns, identify areas of high interest, and optimize web content accordingly. This study, therefore, explores the use of multiple clustering algorithms including K-Means, DBSCAN, Agglomerative Clustering, and Self-Organizing Maps (SOM) to mine and analyze LAUTECH's web log data. The goal is to enhance the accuracy and depth of insights derived from user behavior, contributing to informed decision-making and improved digital services.

Related Works

Van et al. (2024) investigated how Web Usage Mining (WUM) can track engagement and technology adoption trends on university websites. They analyzed server logs and key stream data, and applied clustering algorithms (K-means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN)) and sequential pattern mining to identify key user groups, which are prospective students (45%), researchers (30%), administrators (15%), and casual visitors (10%), and their navigation behaviours. The result revealed strong correlations between online application portal usage and enrollment rates ($R^2=0.72$), while low engagement with virtual tours highlighted UI/UX gaps. Geospatial analysis showed that 60% of international traffic came from India, China, and Nigeria, aiding targeted recruitment. Challenges such as bot traffic and privacy regulations (FERPA/GDPR) were noted. This work bridges web analytics and institutional strategy, offering actionable insights for optimizing digital presence in academia.

Sujeet and Sonam (2024) explored the application of the K-Means clustering algorithm using R language to analyze social networking data, with the aim to segment users based on interaction patterns like post engagements and login frequency. Their methodology involved preprocessing data using `tidyverse`, reducing dimensions via principal component analysis (`prcomp()`), and optimizing cluster count with silhouette analysis. The enhanced K-Means implementation achieved 88% silhouette accuracy and identified five distinct user segments such as passive scrollers, content creators. While the study highlighted R's efficiency (processing 100K+ points in <2 minutes) and visualization strengths (`ggplot2`). Limitations with sparse data and manual centroid initialization were noted.

Rawira and Esichaikul (2023) addresses the inefficiencies in traditional web usage mining by proposing a hybrid framework that combines server logs with client-side JavaScript tracking to improve data quality and reduce preprocessing burdens. The authors recognized and addressed the limitations of log-based systems such as noise, incomplete data, and resource-intensive cleaning by developing an approach integrating automated filtering and machine learning for sessionization. The result demonstrates a 30% reduction in processing time while yielding more accurate user behaviour insights, such as navigation paths and conversion drop-offs. The study concludes that the hybrid model significantly enhances web usage mining efficiency and output reliability. However, it notes challenges like privacy compliance and opportunities for real-time analytics expansion.

Another work by Özkan and Ümit (2023) addresses the inefficiencies of traditional web usage mining that relies heavily on preprocessing noisy server logs. The authors proposed a client-side JavaScript-based data collection method for user tracking, session management, and web usage data collection with the aim of eliminating preprocessing phase and enabling a real-time storage and analysis without log parsing method. The research methodology involves embedding tracking scripts in web pages to directly record user behaviour, bypassing traditional log files. The results revealed that the homogeneous data collected and stored with this method is more convenient to browse, filter, and process than web server logs. The study concludes that this approach significantly optimizes web usage mining workflows but notes limitations such as dependency on client-side scripting and privacy considerations.

Taşgetiren and Aktas (2022) conducted a systematic mapping study on mining web user behaviour. They proposed a real-time WUM framework to address the limitations of traditional batch-processing methods such as scalability and dynamic user behaviour analysis. Their work leverages stream processing technologies such as Apache Kafka for instantaneous data collection and Apache Flink for real-time cleaning and sessionization, combined with incremental k-means for adaptive pattern discovery. The framework was tested on a high-traffic news portal and the result demonstrated sub-second latency in user behaviour analysis, handled over 10,000 events per second, and improved recommendation accuracy by 25% compared to static methods. Limitations such as computational costs and privacy concerns were acknowledged.

Ying (2021) developed a novel Graph Kernel-Based Clustering algorithm to address the challenge of accurately reconstructing user sessions from web server logs. The study proposed a hybrid approach combining traditional time-based and navigation-based methods with machine learning techniques to better handle complex user browsing patterns. Key innovations included a dynamic timeout threshold adjustment mechanism and a feature-enhanced classification model for session boundary detection. Experimental results demonstrated a 15-20% improvement in session identification accuracy compared to conventional methods when tested on e-commerce and educational website datasets. The research also introduced a new evaluation metric that considers both temporal and behavioral characteristics of sessions. While effective, the approach showed increased computational complexity for very large-scale datasets. The work contributed to more reliable preprocessing in WUM pipelines, particularly benefiting applications like personalized recommendation systems and user behavior analysis. Future research directions included optimizing the algorithm for real-time processing and adapting it for mobile app usage patterns.

Ying (2021) proposed a Graph Kernel-Based Clustering Algorithm (GKCA) for WUM to better capture complex user navigation patterns. The method models user sessions as graphs and employs a Weisfeiler-Lehman graph kernel to measure session similarities, followed by kernel-based clustering to group behaviors. Their approach eliminates manual feature extraction and preserves topological relationships in navigation paths. When tested on e-commerce data, GKCA achieved 20% higher cluster purity compared to traditional methods like k-means and DBSCAN, proving particularly effective at detecting rare patterns (e.g., fraudulent behaviors). However, the algorithm's computational cost scales with graph size, making it less efficient for very large datasets (>10,000 nodes).

METHODOLOGY

This research was carried out using the university's weblog data. Specifically, the weblog data was extracted from the weblog server of the university ICT center, which is usually part of the web usage information of their internet users; staff and students. This information was captured over a period of four weeks. The approach to this weblog collection was server-level collection, which gives access to multiple users' information over the same site. Data cleaning and pre-processing was carried out on the data set to make it suitable for analysis.

Web Server Log Collection

Logs are first collected and processed while fields such as IP address, timestamp, URL requested, HTTP status, referrer and user agents are established. Privacy and compliance with data protection regulations were ensured. The weblog extraction tool used was Logstash. The screenshot of some of the samples of extracted weblog is shown in Figures 1 and 2.

	A	B	C	D	E	F	G	H	I	J
1	ID	operation	created_at	ipaddress	email	macaddress	username	platform	browser	
2	4104582	Atolagbe I	10/30/2023 12:10	41.190.2.2	atolagben	6474747	1.9E+13	Linux	Chrome 90.0.44	
3	4104583	Adepoju P	10/30/2023 12:10	102.88.36	adepojupe	6474747	4.1E+12	AndroidOS	Chrome 99.0.48	
4	4104584	Salaudeen	10/30/2023 12:10	102.88.37	salaudeen	6474747	1.1E+13	AndroidOS	Chrome 108.0.0	
5	4104585	SALAMI Y	10/30/2023 12:10	197.210.7	salamiyinu	6474747	2E+13	AndroidOS	Chrome 64.0.32	
6	4104586	SALAMI Y	10/30/2023 12:10	197.210.7	salamiyinu	6474747	2E+13	AndroidOS	Chrome 64.0.32	
7	4104587	Olawore Y	10/30/2023 12:10	129.205.1	olaworeyu	6474747	2E+13	iOS 17_0_	Safari 17.0	
8	4104588	Olawore Y	10/30/2023 12:10	129.205.1	olaworeyu	6474747	2E+13	iOS 17_0_	Safari 17.0	
9	4104589	Olawore Y	10/30/2023 12:10	129.205.1	olaworeyu	6474747	2E+13	iOS 17_0_	Safari 17.0	
10	4104590	Olugbemi	10/30/2023 12:10	102.89.41	olugbemic	6474747	2.2E+13	AndroidOS	Chrome 105.0.0	
11	4104591	Quadri M	10/30/2023 12:10	82.145.22	omololaqu	6474747	2.1E+13	AndroidOS	Opera 69.0.225	
12	4104592	Quadri M	10/30/2023 12:10	82.145.22	omololaqu	6474747	2.1E+13	AndroidOS	Opera 69.0.225	

Figure 1: Extracted weblog dataset

	A	B	C	D	E	F	G	H	I	J
12317	4116897	Oyewale I	10/30/2023 11:10	197.211.5	oyewalepe	6474747	3E+13	AndroidOS	Chrome 104.0.5112.9	
12318	4116898	Abdulham	10/30/2023 11:10	102.88.36	abdulham	6474747	3E+13	AndroidOS	Chrome 105.0.0.0	
12319	4116899	Samuel E	10/30/2023 11:10	105.112.7	eleescosa	6474747	2.5E+13	AndroidOS	Chrome 114.0.0.0	
12320	4116900	Samuel E	10/30/2023 11:10	105.112.7	eleescosa	6474747	2.5E+13	AndroidOS	Chrome 114.0.0.0	
12321	4116901	Abdulham	10/30/2023 11:10	102.88.36	abdulham	6474747	3E+13	AndroidOS	Chrome 105.0.0.0	
12322	4116902	Titiloye A	10/30/2023 11:10	102.88.37	titiloyeayo	6474747	1.4E+13	iOS 16_1_	Safari 16.1	
12323	4116903	Garuba M	10/30/2023 11:10	41.190.31	fawwazga	6474747	3E+13	AndroidOS	Chrome 117.0.0.0	
12324	4116904	Hammed A	10/30/2023 11:10	102.91.70	hammeda	6474747	2.1E+12	AndroidOS	Chrome 97.0.4692.98	
12325	4116905	Titiloye A	10/30/2023 11:10	102.88.36	titiloyeayo	6474747	1.4E+13	iOS 16_1_	Safari 16.1	
12326	4116906	BALOGUN	10/30/2023 11:10	102.88.36	balod526	6474747	1.3E+13	AndroidOS	Chrome 114.0.0.0	
12327	4116907	Mojeed A	10/30/2023 11:10	185.25.23	Angelarac	6474747	2.6E+13	OS X 10_1	Safari 16.1	
12328	4116908	Abdulatee	10/30/2023 11:10	102.88.36	afeezo819	6474747	1.5E+21	AndroidOS	Chrome 108.0.0.0	
12329	4116909	Abdulatee	10/30/2023 11:10	102.88.36	afeezo819	6474747	1.5E+21	AndroidOS	Chrome 108.0.0.0	
12330	4116910	Abdulatee	10/30/2023 11:10	102.88.36	afeezo819	6474747	1.5E+21	AndroidOS	Chrome 108.0.0.0	
12331	4116911	Odeyale J	10/30/2023 11:10	102.88.36	odeyalenil	6474747	2.7E+13	AndroidOS	Chrome 114.0.0.0	
12332	4116912	Odeyale J	10/30/2023 11:10	102.88.36	odeyalenil	6474747	2.7E+13	AndroidOS	Chrome 114.0.0.0	
12333	4116913	Akinpelu F	10/30/2023 11:10	105.113.2	olusegun7	6474747	1.9E+13	iOS 16_0_	Safari 16.0	
12334	4116914	Akinpelu F	10/30/2023 11:10	105.113.2	olusegun7	6474747	1.9E+13	iOS 16_0_	Safari 16.0	
12335	4116915	Akinpelu F	10/30/2023 11:10	105.113.2	olusegun7	6474747	1.9E+13	iOS 16_0_	Safari 16.0	
12336	4116916	Odeyale J	10/30/2023 11:10	102.88.36	odeyalenil	6474747	2.7E+13	AndroidOS	Chrome 114.0.0.0	
12337	4116917	Akinpelu F	10/30/2023 11:10	105.113.2	olusegun7	6474747	1.9E+13	iOS 16_0_	Safari 16.0	
12338	4116918	Agboola C	10/30/2023 11:10	105.113.1	oluwapelu	6474747	2.3E+13	iOS 15_2_	Safari 604.1	

Figure 2: More sample on extracted weblog dataset

Data Preprocessing

The following data preprocessing activities were implemented. This includes:

Log Parsing: Fields extracted from raw log entries,

Data Cleaning: Irrelevant data (e.g., bot traffic, error logs, duplicate entries) were removed.

Session Identification: Session timeout defined (30 minutes of inactivity), Group requests by unique IP addresses or user identifiers and timestamps,

Feature Extraction: Features such as session duration, pages visited, time spent on each page were extracted.

Normalization: Scaling data for clustering (e.g., Min-Max scaling) and Feature Selection. Key features identified for clustering includes platform (OS) distribution, login frequency, and average login hour. Redundant and irrelevant features were then removed. These were executed using python programming.

Figure 3 shows that there are multiple stages involved in data preprocessing which includes: data cleaning (to get rid of repeated and irrelevant information), data integration, data transformation (e.g. normalization to ensure data consistency) and data reduction (convert the data into a suitable dimension).

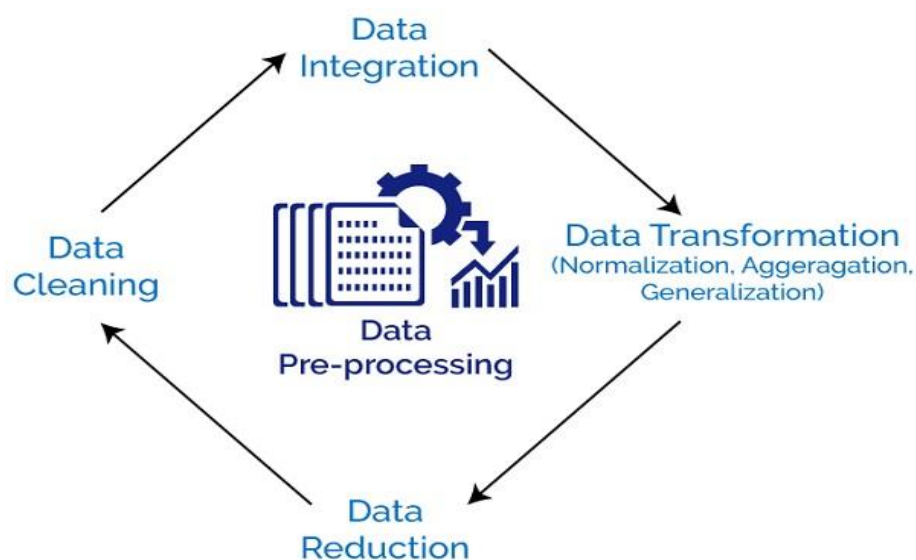


Figure 3: Stages in data preprocessing

Clustering with Multiple Algorithms

The procedure requires choosing particular data subsets for clustering. K-means algorithm was first selected for use. Density-based spatial clustering of applications with noise (DBSCAN) algorithm was applied next on the same data subset. The clusters obtained were analyzed and the results computed. Agglomerative clustering, which is a form of hierarchical clustering was also used to cluster data objects. It started with individual points and merged clusters iteratively using a linkage criteria. Figure 4 shows the clusters formed when agglomerative clustering was implemented on the weblog dataset. Among the implementation steps include applying hierarchical clustering to user sessions; visualizing using a dendrogram to determine the optimal number of clusters; and cutting the dendrogram at a specific level to form clusters.

In addition, Euclidean distance metrics were adopted to achieve a reliable outcome. Self-Organizing Maps (SOM) algorithm was the final algorithm used to cluster the dataset.

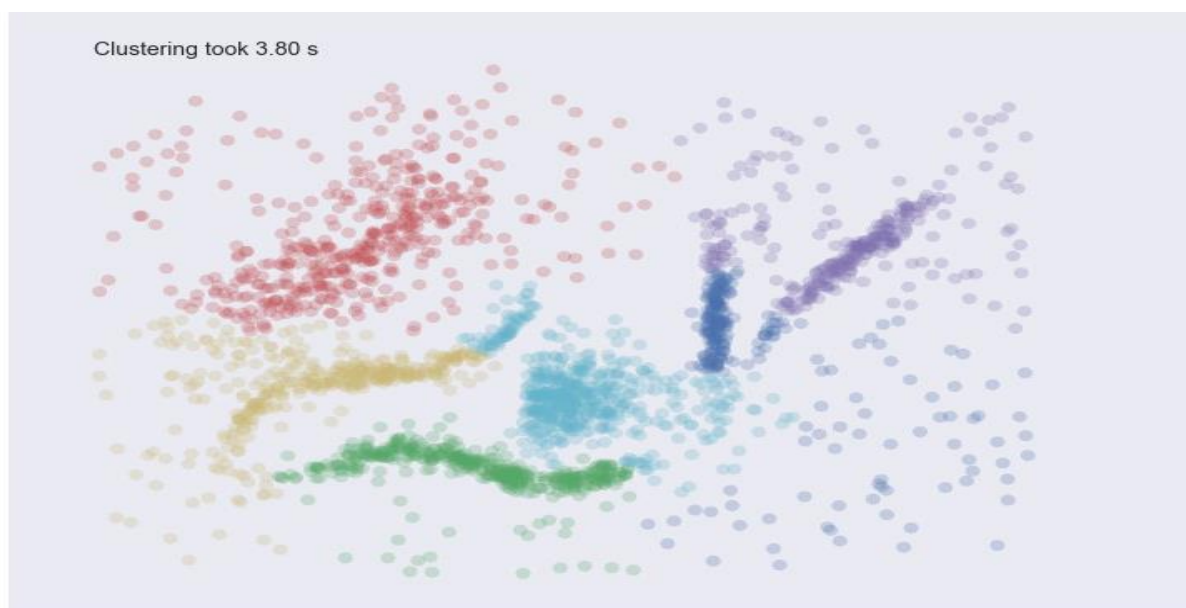


Figure 4: Clusters found by Agglomerative Clustering

Pattern visualization

This was carried out in order to evaluate the impact of each of the afore-mentioned algorithms using silhouette score, number of clusters and noise point. Deducing actionable insights from the analytical testing was made easy through this approach.

RESULTS AND DISCUSSION

Bar charts were used to put forth a pictorial look of cluster distribution obtained through the login frequencies of the various users. These bar charts were used to compare the results of the implementation when the datasets were subjected to algorithmic analysis. Login frequency, average login hour, platform distribution are the metrics used to achieve these.

Kmeans was applied first. It was discovered that 3 distinct clusters were formed as shown in figure 5. The average login count was 2.03 while its silhouette score was 0.4. DBSCAN was applied after Kmeans. There was an overlap of clusters obtained through this algorithm. This shows a high level of noise as shown in figure 6. With DBSCAN, Average login count of 1.50 and silhouette score was 0.86.

Agglomerative clustering was executed next to DBSCAN. Figure 7 shows the bar chart that indicates platform distribution when dataset was clustered by Agglomerative clustering. Three balanced clusters were displayed on the bar chart. The value of Average login count obtained was 1.71 while silhouette score of 0.41 was recorded. The indicated platforms were Android, iOS, Linux, Windows/Java OS with percentage distributions as 70%, 15%, 10% and 5% respectively.

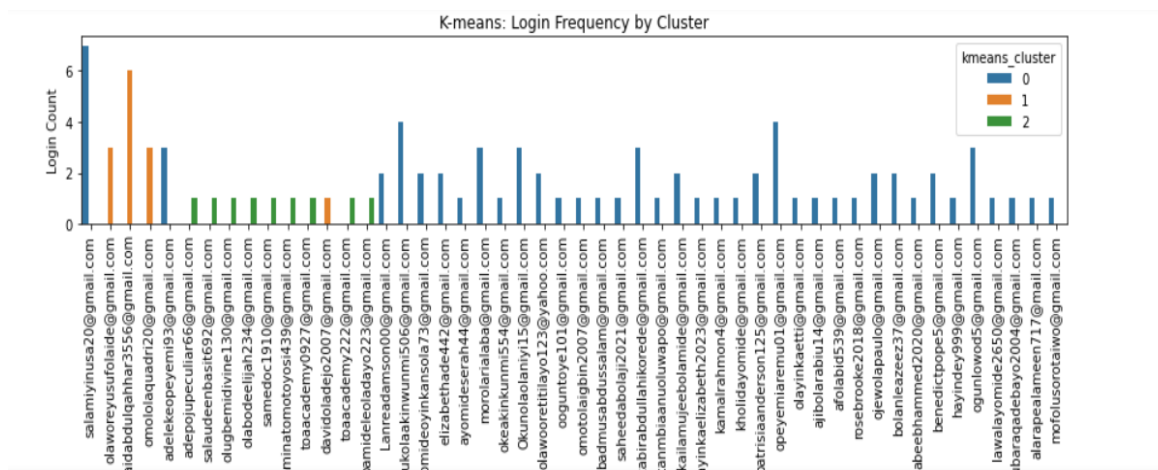


Figure 5: Graph of Login Frequencies with K-means Algorithm

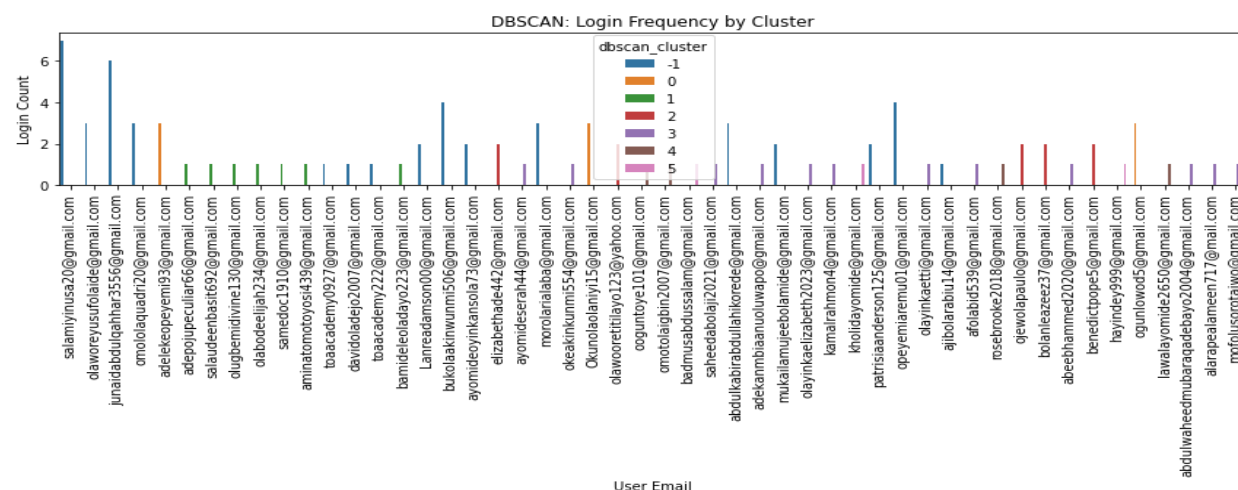


Figure 6: Graph of Login Frequencies with DBSCAN Algorithm

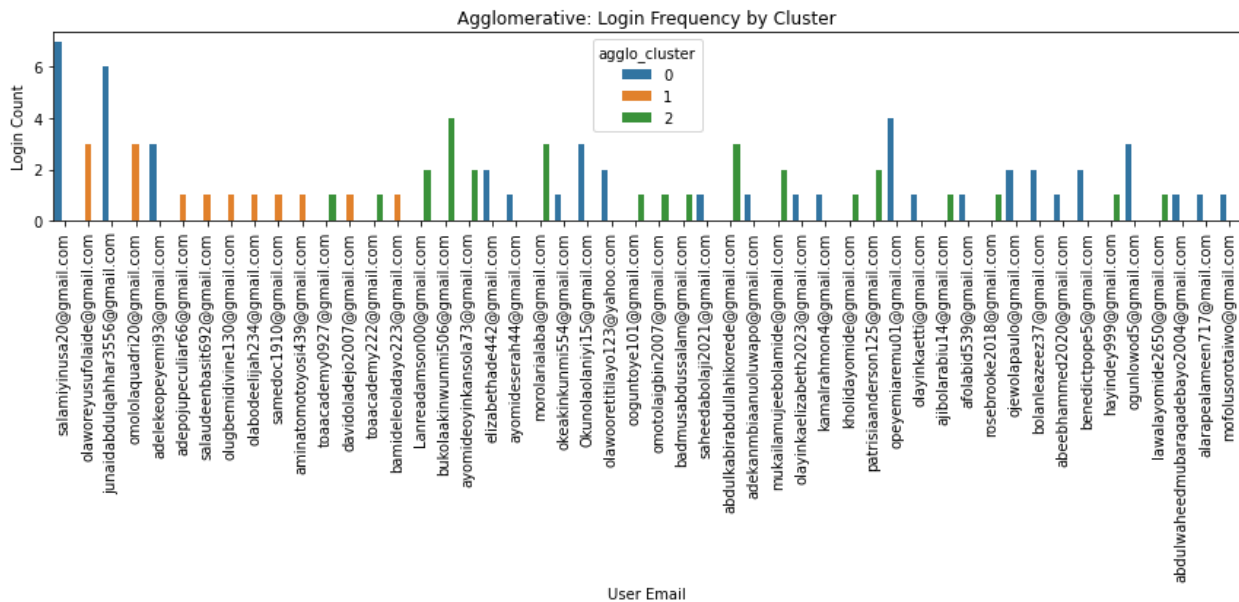


Figure 7: Graph of Login frequencies with Agglomerative clustering

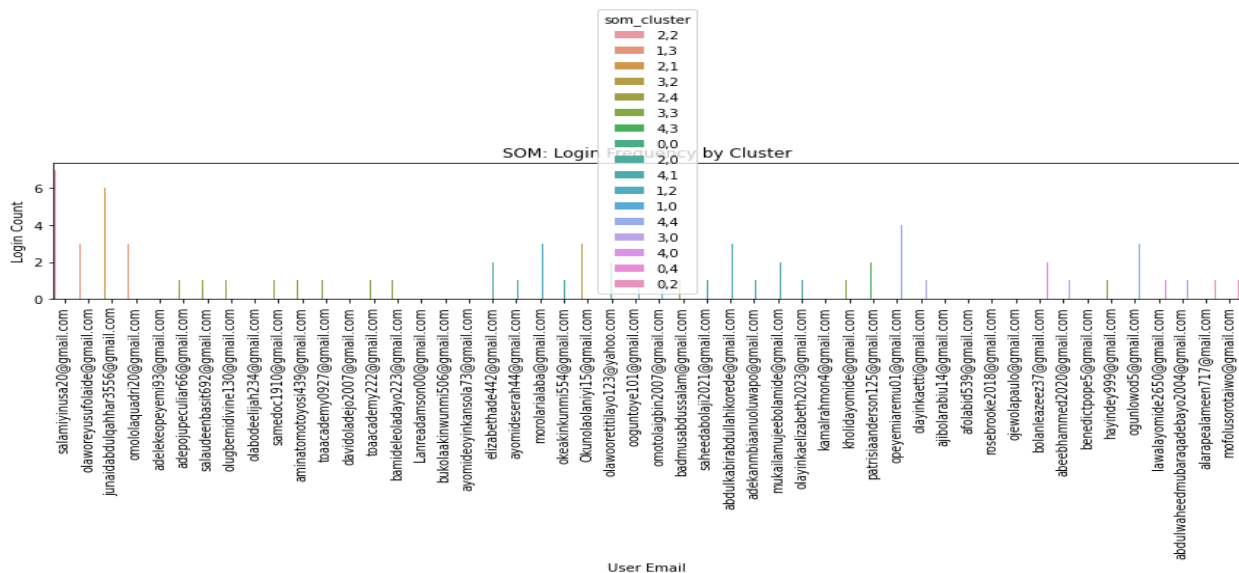


Figure 8: Graph of login frequencies with SOM

Clustering with SOM yields 2.36 Average login count and 0.65 silhouette score. Figure 8 illustrates the performance of SOM through classification of users by their login frequencies into early hours, mid-day and late night users. Each of the clusters in the bar chart was identified by their grid positions in much the same way as points are located on a geographical plane (e.g., “0,0”, “4,4”). The bar chart displays login counts for 50 users in each figure stated above.

Table 1 shows the results of clustering using the stated parameters with the respective algorithms.

Table 1: Summary of Performances of Clustering Algorithms

Algorithm	Silhouette Score	Num of Clusters	Noise Points	Avg. Login Count	Avg. Hour
K-means	0.428116	3	0	2.029279	7.836261
DBSCAN	0.857603	7	16	1.500000	3.688889

Agglomerative	0.409067	3	0	1.711338	5.797084
SOM	0.648579	17	0	2.362745	3.408824

SOM offers a detailed, visual gradient (5-10 grids), with bar charts revealing nuanced frequency and time patterns, but its fragmentation (e.g., 20-100 users per grid) and moderate Silhouette score (~0.30) make it less actionable for large-scale decisions, which is better for exploratory analysis than decisive strategy.

CONCLUSION AND FUTURE WORK

The analysis reveals distinct patterns in students' login behaviours, with K-means and Agglomerative Clustering offering the most coherent and actionable segmentation of the user base. The bar chart for K-means highlights three clear engagement levels: frequent user, infrequent users, and moderate users. This pattern was closely mirrored by Agglomerative clustering, suggesting a robust division of students into high, low, and medium activity groups. For future work, it is recommended that other institutional weblog can be harvested and subjected to analysis and a comparison conducted on its clustering patterns.

REFERENCES

1. Abeer, A. (2024). Optimizing Patient Stratification in Healthcare: A Comparative Analysis of Clustering Algorithms for EHR Data. . International Journal of Computational Intelligence Systems, 17, 173. Retrieved from <https://doi.org/10.1007/s44196-024-00568-8>
2. Ahmad, A., and Hashmi, S. (2016). K-harmonic means type clustering algorithm for mixed datasets. Journal of Applied Soft Computing, 48(2), 39-49. Retrieved 11 27, 2023, from <https://sematext.com/blog/log-analysis/>
3. Aithal, P. S., Prabhu, S., and Aithal, S. (2024). Future of higher education through technology prediction and forecasting. Poornaprajna International Journal of Management, Education, and Social Science (PIJMESS), 1(1), 01-50.
4. Alasali, T., and Ortakci, Y. (2024). Clustering techniques in data mining: a survey of methods, challenges, and applications. Computer Science, 9(1), 32-50.
5. Bahareh, S. A., Neda, A., and ' Saeedeh, R. H. (2022). Predicting customers' behaviour using web content mining and web usage mining. International journal of Information science and Management, 20(3), 141-163. doi:<https://dorl.net/dor/20.1001.1.20088302.2022.20.3.9.6>
6. Behera, A., Panigrahi, C. R., and Pati, B. (2022). Unstructured Log Analysis for System Anomaly Detection. (S. Borah, S. K. Mishra, B. K. Mishra, V. E. Balas, & Z. Polkowski, Eds.) Advances in Data Science and Management, 86(1). doi:https://doi.org/10.1007/978-981-16-5685-9_48
7. Diop, A., El-Malki, N., Chevalier, M., Péninou, A., Roman-Jimenez, G., and Teste, O. (2025). Simrec: a similarity measure recommendation system for mixed data clustering algorithms. Journal of Big Data, 12(1), 43.
8. Kumar, B., Roy, S., Sinha, A., Iwendi, C., and Strážovská, Ľ. (2022). E-commerce website usability analysis using the association rule mining and machine learning algorithm. Mathematics, 11(1), 25.
9. Mehrotra, S., and Kohli, S. (2016, February). Application of clustering for improving search result of a website. In Information Systems Design and Intelligent Applications: Proceedings of Third International Conference INDIA 2016, Volume 2 (pp. 349-356). New Delhi: Springer India.
10. Özkan C. and Ümit K. (2023). An innovative data collection method to eliminate the preprocessing phase in web usage mining, Engineering Science and Technology, an International Journal, Volume 40,101360, ISSN 2215-0986, <https://doi.org/10.1016/j.jestch.2023.101360>.
11. Patidar, P., Posa, S. V., Rao, S., and Bhowmik, B. (2024, December). Enhancing movie recommendation systems with mapreduce genetic algorithms: Addressing scalability and accuracy challenges. In 2024 International Conference on Smart Electronics and Communication Systems (ISENSE) (pp. 1-6). IEEE.
12. Rawira, P., and Esichaikul, V. (2023). Web Usage Mining for Determining a Website's Usage Pattern: A Case Study of Government Website. In C. Anutariya, & M. (. Bonsangue, Data Science and Artificial Intelligence (Vol. 1942). Singapore: Springer, Singapore. doi:https://doi.org/10.1007/978-981-99-7969-1_7

13. Sow, H., and Anandhi, R. (2022). An Efficient and Scalable Dynamic Session Identification framework for web Usage Mining. *International Journal of Information Technology*.
14. Taşgetiren, N., and Aktas, M. S. (2022). Mining web user behavior: a systematic mapping study. In *International Conference on Computational Science and Its Applications* (pp.667 - 83). Springer, Cham.
15. Van Aartsen, B., Noteboom, C., Talley, D., and Tech, D. (2024). Technology adoption in higher education: an analysis of web usage mining in public-facing websites. *Issues in Information Systems*, 25(4).
16. Wolhuter, C. C. (2021). Comparative and International Education: a field of scholarship exploring critical issues in contemporary education. In H. J. Steyn, & C. C. Wolhuter, *Critical Issues in Education Systems: Comparative International Perspectives*. Noordbrug : Keurkopie.