

# Behavioural user segmentation of app users based on functionality interaction patterns

Li-Yoong Ooi<sup>a</sup>, Choo-Yee Ting<sup>a</sup> , Helmi Zakariah<sup>b</sup> and Eashvaren Chandar<sup>b</sup>

<sup>a</sup>Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia; <sup>b</sup>Hayat Technologies Sdn Bhd, Kuala Lumpur, Malaysia

## ABSTRACT

User segmentation categorises a large and complex user base into manageable similar groups of users. Existing works encounter challenges when dealing with a sparse dataset and finding insights from the generated clusters. This study has two objectives: (1) to identify an optimal clustering model that can handle a sparse dataset and (2) to extract post-clustering insights via a descriptive persona for each cluster. This study deployed clustering models to handle a behavioural user-interaction dataset with a sparsity rate of 85%. The findings revealed that Density-Based Spatial Clustering of Applications with Noise that leveraged on One-hot Encoding and data representation learning via an autoencoder performed best, with a Silhouette score of 0.36. Subsequently, this study enacted techniques and tools such as classification, SHapley Additive exPlanation value, and manual analysis. Classification and SHAP values were used to identify important features that can differentiate clusters created by different clustering models. Specifically, a linear SHAP explainer object was applied to Logistic Regression had been identified to outperformed Random Forest and Light Gradient Boosting Machine, with an accuracy of 97%. A manual analysis of the central tendencies of these relatively more important features within each cluster was performed to create a descriptive persona. The findings revealed four distinctive personas, namely the "Active User," "COVID-19 Preventer," "Inactive User," and "Average Joe."

## ARTICLE HISTORY

Received 22 July 2024  
Revised 13 October 2024  
Accepted 8 November 2024

## KEYWORDS

User segmentation;  
clustering; cluster analysis;  
sparse data; SHAP

## SUBJECTS



Artificial Intelligence;  
Computer Science (General);  
CAD CAE CAM – Computing  
& Information Technology

## 1. Introduction

App developers desire precise marketing because it is beneficial in boosting the user experience of a smartphone application user by providing relevant and tailored information (Paweloszek, 2021). However, due to the large user base, targeting and implementing customised business strategies for each user is relatively impractical. Therefore, current studies focus on analysing user-related data and grouping users exhibiting similar properties into a single group. Subsequently, persona generation is performed for each group of users. A persona will typically highlight the unique traits and characteristics of a specific group of users as compared to others. For instance, a "gamer" type of persona might represent a group of users who spend more screen time on gaming apps. In contrast, a

"photographer" type of persona comprises a group of users who interact more with camera apps. The combination of steps, including "grouping" and "persona generation," is well-known as the "user segmentation" process, which helps to simplify a large user base into a few conquerable user cases (Egorova et al., 2022). Consequently, business or service providers, such as app developers, can concentrate on a few groups or clusters at a time (Nguyen, 2021).

A good segmentation solution should construct a segment of members with highly similar characteristics within that particular segment but as dissimilar as possible between different segments. Concerning such a principle, a similarity measure between users is required. Such measures should be carefully constructed because they can directly affect user segmentation. To achieve high accuracy, a typical user

**CONTACT** Choo-Yee Ting  [cying@mmu.edu.my](mailto:cying@mmu.edu.my)  Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia

© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

segmentation algorithm requires the expertise of a domain expert to formulate a set of rules that distinguishes segments of users. However, such a discrimination method is largely dictated by human common sense rather than a data-driven approach (Guidotti & Gabrielli, 2018).

Recent studies employed machine learning approaches to generate a data-driven user segmentation solution. Since the app users are fundamentally not associated with any ground-truth labels, an unsupervised machine-learning approach is needed (Pai et al., 2022). As such, the clustering method has been used for automatic segmentation tasks. Researchers experimented with several clustering algorithms, such as k-Means, Hierarchical Agglomerative Clustering (HAC), Fuzzy C-Means (FCM), Gaussian Mixture Model (GMM) clustering, DBSCAN, and many more (Chopra et al., 2023).

However, the state-of-the-art clustering algorithms have several limitations, including but not limited to poor performance when handling a sparse dataset. Data sparsity issues arise when the proportion of non-zero values is significantly less than the proportion of zero values in a dataset. Such an issue is prevalent in a user-interaction dataset. A sparse dataset negatively impacts the clustering outcome. When applying clustering algorithms, such as k-Means, to a sparse user-interaction dataset, the clustering solution often consists of a dominant cluster comprised predominantly of inactive users. Moreover, it tends to either disperse or poorly group the active users, who are generally more relevant and informative from an analytical perspective. Consequently, the process of persona generation becomes challenging, as it is difficult to characterise the needs and preferences of most users due to their low interaction levels. Similarly, the needs and preferences of the more active users become obscured due to their poor representations in the clustering solution. Hence, such a clustering solution is undesired and problematic.

Furthermore, most clustering algorithms are black-box in nature (Nikolij et al., 2023). The specific data-driven decisions used to cluster users are unknown, which poses difficulty for researchers attempting to interpret or explain the clustering solution (Zhou et al., 2020). Without clear data-driven rules provided, analysing the value differences across various clusters as a critical step of persona generation becomes a complex task, especially when there are numerous features to consider. Consequently, researchers find it challenging to

extract and comprehend tangible insights from the clustering solution that can be instrumental for business applications.

In essence, this study will discuss and address two research problems, which are the low performance of clustering algorithms when dealing with a sparse dataset and the difficulty of extracting post-clustering insights.

Motivated by these problems, this study aims to achieve the following objectives, as follows:

1. To identify an optimal clustering model that can handle a sparse dataset
2. To extract post-clustering insights via a descriptive persona for each cluster

The rest of this paper is organised as follows: [Section 2](#) presents a literature review of related studies. [Section 3](#) describes the methods used to achieve the objectives. [Section 4](#) explains the findings derived. [Section 5](#) summarises the achievements of this study and future directions.

## 2. Literature review

This section provides a thorough literature review, encompassing different segmentation bases, techniques for handling a sparse dataset, clustering algorithms, clustering model evaluation approaches, and methods for extracting insights from the generated clustering solutions.

### 2.1. Segmentation base

A segmentation base can be defined as a set of features that contributes to grouping users into several heterogeneous groups (An, 2020). The segmentation base changes according to the nature of the involved features. In general, four common segmentation bases are reported, namely, demographic, behavioural, geographic, and psychographic (An, 2020; Nandapala et al., 2020).

[Table 1](#) shows a summary of different segmentation bases utilised in the reviewed literature.

Demographic segmentation involves the use of demographic features such as age, race, gender, culture, religion, marital status, and others (Nandapala et al., 2020; Vajjhala & Strang, 2019; Mehta et al., 2021). Demographic features are relatively more straightforward to understand and analyse. However, such data is complex and difficult to collect for several reasons. Such data is typically obtained from users rather than automatically derived from the

**Table 1.** User segmentation bases.

References	Demographic segmentation	Behavioural segmentation	Geographic segmentation	Psychographic segmentation
Paweloszek (2021)	✓			
Pai et al. (2022)		✓		
Zhou et al. (2020)	✓			
Nandapala et al. (2020)	✓	✓		
Chang et al. (2020)		✓		
Baumgarte et al. (2021)	✓			
Natilli et al. (2019)	✓	✓		
Vajjhala and Strang (2019)	✓			
Karaliopoulos et al. (2022)	✓			
Ben-Gal et al. (2019)	✓			
Shen (2021)		✓		
He and Chen (2021)	✓	✓		
Mehta et al. (2021)	✓	✓	✓	✓
Nishimi et al. (2022)	✓			
Li et al. (2023)		✓	✓	

user activities. However, not all users are willing to share demographic data. Additionally, users can browse anonymously, thus choosing not to provide any demographic data at all (Alves Gomes & Meisen, 2023).

Behavioural segmentation utilises features related to browsing habits, loyalty to service, and many more (Mehta et al., 2021). Behavioural segmentation is relatively more complex than demographic segmentation. Yet, the collection of data is more automated and implicit (Alves Gomes & Meisen, 2023). In addition, since data such as screen time is internally recorded rather than explicitly inputted by users, the resulting data becomes more reliable.

Geographical segmentation divides users into different groups by considering their geographical location. The geographical data includes but is not limited to country, region, city, and even postal code. It saves resources by selecting a particular target area (Mehta et al., 2021).

Psychographic segmentation requires data on users' hobbies, personality traits, values, goals, and others (Mehta et al., 2021). These data can be collected either explicitly or implicitly. Surveys can be conducted to gather such information explicitly, but users' willingness to share their private information is uncontrollable. On the other hand, implicit data collection requires the aid of domain experts who can extract users' psychographic data from their online activities.

Despite the differences, there is no exact way to conduct micro-segmentation precisely (Nandapala et al., 2020). The features used in different segmentation bases sometimes overlap or are related. For example, the behavioural traits of users are often associated with their demographic attributes, such as age. The study in (Vajjhala & Strang, 2019) showed that younger individuals preferred to spend

more time online than older people, illustrating how age could influence online behaviours.

Moreover, the influence of different features varies in the literature review. For example, the study in concluded that demographic features were less important. Meanwhile, the study in (Nishimi et al., 2022) such as gender, were less influential. Similar to demographic segmentation, those studies that performed behavioural segmentation also present a mixture of findings. The study in (Jansen et al., 2017) concluded that there was no difference in behaviour features across clusters. However, the study heavily relied on behaviour features.

Therefore, this study summarises that the influence of certain features on segmentation results should be discussed in a case-specific manner rather than a generalisation. Additionally, the concept of behavioural segmentation should not be highly rigid to an extent such that disposing of all demographic features that might be intrinsically linked to the users' behaviour.

### 2.1. Handling a sparse dataset

When clustering a sparse dataset, there are different approaches such as using a suitable similarity measure like cosine similarity or normalised squared Euclidean distance, and performing pre-clustering data transformation via techniques like Principal Component Analysis (PCA), autoencoding, and sentence embedding.

Table 2 summarises different approaches to handling a sparse dataset from the reviewed literature.

One of the popular approaches to adopt when dealing with a sparse dataset is using cosine similarity, rather than the commonly used Euclidean distance, as a similarity measure between users. Cosine similarity is operated by mapping vectors within a

**Table 2.** Approaches to handle a sparse dataset.

References	Cosine similarity	Normalised squared Euclidean distance	PCA	Autoencoder	Sentence embedding
Egorova et al. (2022)				✓	
Umuhoza et al. (2020)		✓	✓		
McConville et al. (2020)				✓	
Fard et al. (2020)				✓	
Tissera et al. (2024)					✓
Sembiring Brahmana et al. (2020)	✓	✓			
Munusamy and Murugesan (2020)		✓			
Alghamdi (2023)			✓		

graphical context and then computing the cosine of the angle between the corresponding vectors. On the other hand, Euclidean distance is also known as the Pythagorean Distance when the vector space is 2-dimensional, and the calculation is simply the length of the hypotenuse that connects the two vectors in this case. Euclidean distance can be extended to a higher dimensional space similarly. The key difference between a cosine similarity and Euclidean distance is that the former emphasises vectors' orientation, while the latter emphasises both the magnitude and orientation of vectors. When the dataset is sparse, and especially if the dimensionality is high, the non-zero data points are far apart. This causes difficulty in forming meaningful clusters constituting non-zero data points because such data points are dispersed and scattered around. Therefore, the similarity measure should not be too strict. Hence, cosine similarity is preferred due to its greater flexibility. Since the magnitude is ignored in the case of cosine similarity, the vectors can still be close to each other if the angle between them is relatively small, regardless of their difference in magnitude.

Unfortunately, cosine similarity is not available when initialising some clustering models, including but not limited to k-Means. Therefore, an alternative emerged, which is to leverage a normalised squared Euclidean distance. Normalisation reduces the scales of features to a smaller range, thus reducing the magnitude difference between data points. Consequently, the magnitude effect in calculating Euclidean distance is minimised. This produces a similar effect to a cosine similarity, emphasising orientation or direction rather than magnitude. Indeed, the study in (Korenius et al., 2007) mathematically derived that cosine similarity and Euclidean distance were non-linearly related, primarily due to the square root in the Euclidean distance formula. Therefore, a squared Euclidean distance and cosine similarity are linearly related because the effect of the square root is eliminated.

In short, cosine similarity should be used if possible, such as when a DBSCAN clustering algorithm is implemented. On the other hand, the normalised

squared Euclidean distance leads to a similar effect and advantage of cosine similarity. Hence, it should be considered as an alternative if necessary. For instance, the study in (Umuhoza et al., 2020) implemented min-max scaling as a data normalisation step when handling sparse data.

Additionally, another approach is converting sparse data into dense format. Dense data indicates that the proportion of zero values is significantly lower than the proportion of non-zero values, which is the exact opposite of the sparse data scenario. Hence, the data sparsity problem might be solved if the sparse data can be transformed into a dense format. In general, such a data transformation requires discovering the intrinsic structure of data and transforming it into a more readily usable format. In our case, the sparse data is expected to be discovered and transformed into a dense format. The process encompassing these steps is popularly known as data representation learning or feature learning (Zhong et al., 2016). Examples of data representation learning techniques or tools include PCA, autoencoder, and sentence embedding.

PCA learns a linear transformation of data into a new space, typically via the use of eigendecomposition of the covariance matrix or by computing the Singular Value Decomposition of data (McConville et al., 2020).

Autoencoder is an instance of deep neural networks that are trained to transform the inputted data into a dense and low-dimensional vector at the network bottleneck and then attempt to reconstruct the input based on this created vector (Fard et al., 2020). If such an attempt yields acceptable performance, then the data transformation is of high quality and the encoded dense and low-dimensional vector can be used for clustering (Egorova et al., 2022).

The sentence embedding method transforms inputted sentences into vector representations that are able to capture their semantic meaning (Tissera et al., 2024). Numerical data must be integrated into sentences before using the sentence embedding method.

### 2.3. Clustering algorithm

Numerous clustering algorithms were attempted in the reviewed literature, such as k-Means, HAC, FCM, probabilistic clustering with GMM, and DBSCAN. There is no best clustering algorithm among all. In fact, the choice of the clustering algorithm to be used is case-specific because the data on hand contributes heavily to the performance of these clustering algorithms (Alves Gomes & Meisen, 2023).

Table 3 presents a summary of the clustering algorithms implemented in the reviewed literature.

The study in (Pawelozsek, 2021) used the k-Means algorithm to group wearable activity users into several clusters based on their running habits. The author of (Pawelozsek, 2021) also computed several new features, such as average training time per week, average race distance, and average speed from the raw features, including but not limited to heart rate, distance, and speed, to describe the running behaviours of the users best. Subsequently, the study in (Nandapala et al., 2020) defined a set of running motivations for each cluster of users. For instance, the "Novice" group was motivated to reduce weight, while the "High-caliber" group aimed to increase their running performance for competition purposes. On the other hand, the study in (Nandapala et al., 2020) segmented Health Insurance company customers into a few groups based on their claiming patterns. With the aid of 'recency, frequency, and monetary' (RFM) analysis and the k-Means algorithm, the author of (Nandapala et al., 2020) successfully identified five distinct clusters. Among them, one of the clusters represented a group of customers with the most recent claims and the highest amount. Subsequently, the author of (Nandapala et al., 2020) claimed that this group of customers was most dangerous for the company's profit because they caused the most loss. Besides that, the study in (Natilli et al., 2019) utilised the k-

Means algorithm to group students from the University of Pisa based on their eating habits. The incorporated dataset stored the meal records of each student, and the details included but were not limited to the food type and timestamp. Eventually, four clusters of students were identified based on their eating habits, which were "balanced," "foodie," "health fanatic," and "voracious." In this study, the author of (Natilli et al., 2019) claimed that demographic data also played a vital role. For instance, the "voracious" group was mainly consisted of female students.

The study in (Hung et al., 2019) used a HAC algorithm to segment credit card users into three groups based on their credit card usage data, such as credit limits and minimum payment. Besides that, the study in (Irawan et al., 2020) implemented a HAC algorithm to group Twitter users into a few clusters based on their tweets on political news. The common words used by each user were extracted and used as their respective behavioural reactions toward the political issues. Moreover, the study in (Baumgarte et al., 2021) utilised a HAC algorithm to segment carsharing users from Augsburg, Germany. The dataset contained details of each appointment record, such as the car type chosen, appointment timestamp, distance travelled, and many more. Subsequently, the author of (Baumgarte et al., 2021) identified clusters such as frequent, long-term, and long-distance users.

The study in (Munusamy & Murugesan, 2020) implemented a variant of the FCM algorithm to segment customers of a retail supermarket in India. They used a transactional dataset that contained pre-transformed RFM values. As a result, six clusters of customers were identified, namely "best," "shoppers," "first-time," "churn," "frequent," and "uncertain." In addition, they suggested providing a customer reach-out campaign to the "uncertain" group of customers currently considered to be of "no value" to the organisation to

**Table 3.** Clustering algorithms for user segmentation.

References	K-Means	HAC	FCM or its variation	Probabilistic clustering with GMM	DBSCAN
Pawelozsek (2021)	✓				
Egorova et al. (2022)	✓				
Nguyen (2021)				✓	
Pai et al. (2022)	✓				
Nandapala et al. (2020)	✓				
Chang et al. (2020)	✓				
Baumgarte et al. (2021)	✓	✓			
Mehta et al. (2021)	✓	✓		✓	
Sembiring Brahmana et al. (2020)	✓				✓
Munusamy and Murugesan (2020)			✓		
Hung et al. (2019)		✓			
Chen et al. (2023)			✓		
Irawan et al. (2020)	✓	✓			
Lee and Cho (2021)				✓	



revive the interest of such customers towards the stores.

The study in (Lee et al., 2018) attempted a GMM algorithm to group smartphone users into a few clusters based on their app usage sequence. The dataset was provided by LG Electronics, and all the users investigated were using LG smartphones. Unlike most behavioural segmentation studies, this dataset offered additional information, which was the "sequence." The dataset described the app used and the sequence in which these apps are being used. As a result, the model generated ten groups of smartphone users. Subsequently, the author of (Lee et al., 2018) derived a suitable persona for each user group. For example, the "photographers" were regarded as users with the highest engagement on camera-related apps. On the other hand, the "beginners" represented users who used the fewest apps compared to users from all other clusters.

The study in (Ji et al., 2022) implemented a DBSCAN algorithm to segment in-vehicle On Board Diagnostic (OBD) II device users based on travel behaviours. OBD II was a diagnostic system implanted inside each car engine investigated. Such systems provided trip details and were referred to as mobility data in this study. The data was collected from volunteers who installed the devices and agreed to data usage in Chicago, United States, for analytic purposes. The leveraged features incorporated behavioural and geographical natures, such as the frequency of visits to the most frequently visited location, the number of unique places visited, and many more. As a result, the DBSCAN model produced six user clusters based on their travel behaviours.

## 2.4. Clustering model evaluation

Commonly used cluster validation indices include Silhouette, Calinski-Harabasz, Davies-Bouldin, and Dunn. These indices describe the cluster compactness and separation. Typically, a good clustering solution should exhibit high intra-cluster and low

inter-cluster similarities. In other words, the compactness within a cluster should be high, and the separation between distinct clusters should be significant. Table 4 presents a summary of leveraged cluster validation indices from the reviewed literature.

Silhouette's values range in  $[-1, 1]$ . In an ideal case, a value of 1 indicates that the intra-cluster similarity is exceptionally high and the inter-cluster similarity is extremely low. A value of 0 indicates that the clusters are overlapping. A value of  $-1$  expresses that the clusters are poorly formed and irregular in shape.

Calinski-Harabasz has no clear range of values. However, a high Calinski-Harabasz value is preferred to indicate a significant separation between clusters and an excellent tightness within the cluster.

Davies-Bouldin's values range in  $[0, \infty)$ . Unlike Calinski-Harabasz, a lower value of Davies-Bouldin should be obtained for well-separated clusters.

The value range for a Dunn score is  $[0, \infty)$ . A high Dunn score is preferred as it indicates the minimum inter-cluster distance is large, and the maximum intra-cluster distance is small. Dunn index is considered to evaluate the "worst case" because its computation includes extreme values of intra-cluster and inter-cluster distances.

All four of these aforementioned cluster validation indices are known as internal cluster validation indices as well. Specifically, internal cluster validation indices do not require the presence of ground truth. Therefore, they are popularly used when evaluating a clustering model because ground truth is mostly absent for an unsupervised learning scenario like clustering.

Several studies have used the idea of examining if clusters can be distinguished easily and accurately to describe the quality of a clustering solution, based on the rationale that a good clustering solution will yield a good classification model. Cluster labels predicted by a clustering model are treated as the target variable, and the features being inputted into that particular clustering model previously are

**Table 4.** Commonly-used cluster validation indices.

References	Silhouette	Calinski-Harabasz	Davies-Bouldin	Dunn	Classification
Nguyen (2021)	✓				
Chang et al. (2020)	✓				
Baumgarte et al. (2021)	✓				
Ben-Gal et al. (2019)	✓				
Umuhoza et al. (2020)	✓	✓			
Hung et al. (2019)	✓				
Rodríguez et al. (2018)					✓
Peker et al. (2017)	✓	✓	✓		
Zhu et al. (2019)		✓	✓	✓	
Zhu and Ma (2018)		✓	✓	✓	
Rezaei and Franti (2020)					✓

considered the independent variables. Then, the independent variables and target variable are inputted into one or more classifiers to attempt to distinguish between the cluster labels, which are treated as the target variable. The trained classifier should be capable of correctly predicting the belonging cluster of an unseen object, provided that the inputted clustering solution is good because the patterns that govern clustering are easily identifiable and learnable (Abul et al., 2003; Rodríguez et al., 2018). In other words, the better the performance of the trained classifier, the better the quality of the clustering solution. Metrics such as accuracy, precision, recall, and others are useful for evaluating the performance of the trained classifier.

Unsupervised clustering has no natural cluster labels or ground truth. Therefore, internal cluster validation indices that evaluate the goodness of clustering structure without respect to ground-truth cluster labels are more commonly used for clustering model evaluation purposes. Aside from that, using a classification approach to evaluate the performance of a clustering model is an intuitive and logical approach as well.

## 2.5. Extracting post-clustering insights

The desired insights from a user segmentation process are several descriptive personas that effectively describe an extensive user system in simple terms. However, the specific data-driven rules that contributed to the formation of clusters are generally untold by the black-box clustering models. Therefore, researchers often implemented additional methods post-clustering to analyse and transform the clustering solution into actionable insights.

Some studies computed and compared the central tendencies of each feature across different clusters, aiming to identify a significant value difference. If a feature within a cluster exhibits a significantly different central tendency compared to the same feature in other clusters, then this implies that the users in this cluster have unique behaviours related to that feature. Subsequently, such information aids in persona generation. This study classifies such a method as “manual analysis” because it revolves around logical inspection and deduction. Unfortunately, manual analysis of many features is challenging due to the extensive resources required for thorough analysis.

A smaller scope of features to analyse has to be decided to overcome the challenge above. Some studies attempted a “classification” approach before

manual analysis (Li et al., 2023). Such an approach treats the cluster labels predicted by a clustering model as the target variable and the original features as independent variables. Subsequently, an easily interpretable classifier, such as a decision tree, is instructed to predict cluster labels based on the set of independent variables. Then, the trained classifier is evaluated using metrics such as accuracy, precision, recall, and others. If satisfactory performance is obtained, this indicates that the classifier learned data-driven rules to distinguish between clusters rather than memorising the training data. Unlike a clustering model, the data-driven rules used by an easily interpretable classifier are explained in terms of the feature importance of independent variables towards predictions during the training phase. As a result, a set of “relatively more important features” can be decided after performing a feature importance analysis. This set of “relatively more important features” contributed more to distinguishing between clusters. In other words, these feature values have more significant variations across different clusters; thus, they are more worth analysing. Hence, the scope of features to analyse manually for persona creation purposes can be narrowed down to only these “relatively more important features.”

However, not all classifiers are easily interpretable. Some studies, such as those in (Li et al., 2023), incorporated SHAP with the trained classifier to extend the feature importance analysis to less interpretable classifiers. SHAP provides a unified measure of feature importance, regardless of the classifier’s inherent interpretability. Due to its model-agnostic property, SHAP can be applied to various classifiers, such as LR, RF, LGBM, and many more.

Table 5 summarises approaches to extract post-clustering insights from the reviewed literature.

## 3. Method

This section explains the methods employed in this study. This study started with data pre-processing and proceeded with clustering model construction and evaluation. Finally, it extracted insights from the generated clustering solution.

### 3.1. Data pre-processing

A Malaysian Company provided eight quarterly datasets for this study. These raw datasets were publicly unavailable. This study classified these raw datasets as  $D_{2021-Q4}$ ,  $D_{2022-Q1}$ ,  $D_{2022-Q2}$ ,  $D_{2022-Q3}$ ,  $D_{2022-Q4}$ ,  $D_{2023-}$

**Table 5.** Approaches for extracting post-clustering insights.

References	Manual analysis	Classification + manual analysis	Classification + SHAP + manual analysis
Paweloszek (2021)		✓	
Egorova et al. (2022)	✓		
Nguyen (2021)	✓		
Zhou et al. (2020)	✓		
Chang et al. (2020)	✓		
Baumgarte et al. (2021)	✓		
He and Chen (2021)			✓
Li et al. (2023)		✓	
Munusamy and Murugesan (2020)	✓		
Hung et al. (2019)	✓		
Irawan et al. (2020)	✓		
Stormi et al. (2020)	✓		
Myburg and Berman (2022)		✓	

**Table 6.** Description of raw datasets.

Dataset	Description
$D_{2021-Q4}$	Dataset of 2021 fourth quarter
$D_{2022-Q1}$	Dataset of 2022 first quarter
$D_{2022-Q2}$	Dataset of 2022 second quarter
$D_{2022-Q3}$	Dataset of 2022 third quarter
$D_{2022-Q4}$	Dataset of 2022 fourth quarter
$D_{2023-Q1}$	Dataset of 2023 first quarter
$D_{2023-Q2}$	Dataset of 2023 second quarter
$D_{2023-Q3}$	Dataset of 2023 third quarter

$Q1$ ,  $D_{2023-Q2}$ , and  $D_{2023-Q3}$  respectively. Table 6 shows a description of each raw dataset.

Each raw dataset was a mixture of user-demographic and user-interaction datasets. All users were using the integrated app by the specific Malaysian Company. Each raw dataset contained several demographic features and behavioural features. Examples of demographic features were app users' gender, race, and date of birth (DOB). On the other hand, the behavioural features were represented by the screen time of app users on different app pages. A feature in the dataset represented each screen, and that corresponding feature shared the same name with that screen. Table 7 shows the information of some features.

From Table 7, the top three rows displayed the demographic features in the raw datasets. Meanwhile, only the descriptions of three behavioural features were shown in the bottom three rows. Other behavioural features worked similarly.

However, a notable fact was the difference in the number of features across raw datasets. Due to newly added app pages, later raw datasets contained more features than the earlier raw datasets.

This study aimed to prepare a single analytical dataset from the raw datasets to avoid duplicating efforts afterwards. Older raw datasets such as  $D_{2021-Q4}$ ,  $D_{2022-Q1}$ ,  $D_{2022-Q2}$ ,  $D_{2022-Q3}$ ,  $D_{2022-Q4}$ ,  $D_{2023-Q1}$ , and  $D_{2023-Q2}$  were aligned to the latest dataset, which was  $D_{2023-Q3}$  in terms of the dimensionality. Features were created and filled with null values,

if necessary, during the feature integration step. Then, this study aggregated data from all raw datasets. Value summation strategy was implemented for numerical features, while the first value encountered in the chronological order of quarters was chosen for categorical features. Furthermore, the 'DOB' feature was transformed into the 'Age' feature for more straightforward interpretation. Data cleaning was always performed throughout the process through tasks such as removing missing and duplicated values and maintaining categorical value consistency. In addition, the index key was removed. Finally, the processed analytical dataset was hereinafter referred to as  $D_{analytical}$ .  $D_{analytical}$  contained 53 features and achieved a sparsity rate of 85%. Therefore, this study classified  $D_{analytical}$  as a high-dimensional sparse dataset, necessitating a tailored clustering approach.

In addition, a dataset containing 1000 records randomly sampled from  $D_{analytical}$  is generated and hereinafter referred to as  $D_{sample}$ .  $D_{sample}$  also achieved a sparsity rate of 85% and is classified as a high-dimensional sparse dataset.  $D_{sample}$  is for later use during the construction of some clustering models.

Table 8 displays the available features contained in  $D_{analytical}$  and  $D_{sample}$ . The detailed information of features is omitted in this table.

Later, exploratory data analysis (EDA) was carried out to gather more information, such as data distribution, feature correlations, and others.

### 3.2. Clustering model construction

Concerning the first study objective, this study deployed clustering models to handle our sparse dataset.

This study implemented three clustering algorithms, namely k-Means, HAC, and DBSCAN. Varying



**Table 7.** Information of some features.

Feature	Description	Data type
GENDER	Recorded the user's gender	Categorical
RACE	Recorded the user's race	Categorical
DOB	Recorded the user's DOB	Continuous
Mental SEHAT	Recorded the user's screen time on the "Mental SEHAT" app page	Continuous
Screening Booking	Recorded the user's screen time on the "Screening Booking" app page	Continuous
Health Record	Recorded the user's screen time on the Health Record app page	Continuous

**Table 8.** Features in  $D_{analytical}$ .

Feature base	Feature
Demographic	GENDER, RACE, age
Behavioural	APK Download, Bubble page, Frequently Asked Question, Health, Homepage, Inform Status, Map, Mental SEHAT, Notification, Remedi CAC, Remedi HAT, Remedi Screening, Remedi SelVax, Screening Booking, SelCare Store, SelVax, User Info, Online Wallet, Settings page, Vaccine Booking, Vaccine Cert, WhatsDoc Line 1, WhatsDoc Line 2, ASAS, Adworks, Bingkas (CARD), Lifetime Health Record, Medical Record (Saring Result), Biz, Selangor Saring, Zakat, Sign in, Sign up, Verify OTP, Clinic/Dental Appointment (Selcare), Clinic Locator, Home Nursing & Physio (Selcare), ISS, ISSPA, Kanser Selangor, Perak, Pharmacy (Selcare), Rawatan Tibi Selangor, SelVax (Vaccine), Selangor, Selcare Corporate/TPA (Selcare), Selgate Foundation, Skim Rawatan Jantung, Telemedicine (Selcare), Terengganu

models of these algorithms were instantiated with different combinations of pre-processing techniques and hyperparameter values.

Examples of pre-processing techniques attempted were robust scaling, min-max scaling, standard scaling, OHE, PCA, autoencoder, and sentence embedding.

For PCA, the number of principal components to include was determined by the cumulative explained variance. All principal components that collectively account for at least 80% of the total variance were selected.

For autoencoding, this study used a fully-connected Multi-Layer Perceptron autoencoder. The dimensions were  $d_i$ -500-500-2000- $d_e$ , where  $d_i$  were the dimensionality of the inputted data and  $d_e$  referred to the dimensionality of the encoded data. Here,  $d_e$  was always defined as 10. In other words, the autoencoder always encodes the inputted data into 10 dimensions. As typical with autoencoders, the decoder network was a mirror of the encoder. All layers used ReLU activation (Nair & Hinton, 2010). The optimizer was Adam (Kingma & Lei Ba, 2014). In addition, the autoencoder was trained for 50 epochs with a batch size of 256.

To prepare sentence embedding, a sentence was first created for each inputted dataset's record by concatenating its categorical and numerical values. An example of the created sentences is shown in Figure 1. Due to its great length, the middle part of this sentence was truncated. All sentences were then passed to a "paraphrase-MiniLM-L6-v2" sentence transformer for embedding generation.

For the k-Means algorithm, the number of clusters,  $k$ , must be determined a priori. The elbow criterion method was utilised for this purpose.

```
GENDER: female,
RACE: indian,
Apk Download: 0.0,
Bubble page: 8.33333333333332,
Frequently Asked Question: 140.0,
Health: 0.0,
Homepage: 4.823809523809524,
Inform Status: 0.0,
Map: 0.0,
.
.
.
Verify OTP: 0.0,
Zakat: 0.0,
Age: 37
```

**Figure 1.** Example sentence.

For both HAC and DBSCAN algorithms, the models were using  $D_{sample}$  due to hardware limitations.

For the HAC algorithm,  $k$  was determined visually from the dendrograms.

For all DBSCAN models, the 'eps' and 'min\_samples' values were set to 0.3 and 58, respectively, which were optimal after a grid search.

In addition, OHE was performed to encode categorical data into numerical representations whenever necessary due to clustering algorithm restrictions.

Table 9 presents all clustering models constructed. Each clustering model was assigned a code to be referred to in further discussions.

### 3.3. Clustering model evaluation

This study evaluated the performance of all constructed clustering models using cluster validation indices such as Silhouette and Calinski-Harabasz. Additionally, for each clustering model, a LR was trained to distinguish between clusters formed by it, then this trained LR was evaluated using metrics like

**Table 9.** Clustering models constructed.

Model code	Dataset	Experimental settings		
		Data pre-processing	Clustering algorithm	Hyperparameter
K1	$D_{analytical}$	OHE	K-Means	N/A
K2	$D_{analytical}$	Robust scaling + OHE	K-Means	N/A
K3	$D_{analytical}$	Min-max scaling + OHE	K-Means	N/A
K4	$D_{analytical}$	Standard scaling + OHE	K-Means	N/A
K5	$D_{analytical}$	OHE + PCA	K-Means	N/A
K6	$D_{analytical}$	OHE + Autoencoder	K-Means	N/A
K7	$D_{analytical}$	Sentence embedding	K-Means	N/A
H1	$D_{sample}$	OHE	HAC	Linkage = 'complete'
H2	$D_{sample}$	OHE	HAC	Linkage = 'single'
H3	$D_{sample}$	OHE	HAC	Linkage = 'average'
H4	$D_{sample}$	OHE	HAC	Linkage = 'ward'
H5	$D_{sample}$	Robust scaling + OHE	HAC	Linkage = 'complete'
H6	$D_{sample}$	Robust scaling + OHE	HAC	Linkage = 'single'
H7	$D_{sample}$	Robust scaling + OHE	HAC	Linkage = 'average'
H8	$D_{sample}$	Robust scaling + OHE	HAC	Linkage = 'ward'
H9	$D_{sample}$	Min-max scaling + OHE	HAC	Linkage = 'complete'
H10	$D_{sample}$	Min-max scaling + OHE	HAC	Linkage = 'single'
H11	$D_{sample}$	Min-max scaling + OHE	HAC	Linkage = 'average'
H12	$D_{sample}$	Min-max scaling + OHE	HAC	Linkage = 'ward'
H13	$D_{sample}$	Standard scaling + OHE	HAC	Linkage = 'complete'
H14	$D_{sample}$	Standard scaling + OHE	HAC	Linkage = 'single'
H15	$D_{sample}$	Standard scaling + OHE	HAC	Linkage = 'average'
H16	$D_{sample}$	Standard scaling + OHE	HAC	Linkage = 'ward'
H17	$D_{sample}$	OHE + PCA	HAC	Linkage = 'complete'
H18	$D_{sample}$	OHE + PCA	HAC	Linkage = 'single'
H19	$D_{sample}$	OHE + PCA	HAC	Linkage = 'average'
H20	$D_{sample}$	OHE + PCA	HAC	Linkage = 'ward'
H21	$D_{sample}$	OHE + Autoencoder	HAC	Linkage = 'complete'
H22	$D_{sample}$	OHE + Autoencoder	HAC	Linkage = 'single'
H23	$D_{sample}$	OHE + Autoencoder	HAC	Linkage = 'average'
H24	$D_{sample}$	OHE + Autoencoder	HAC	Linkage = 'ward'
H25	$D_{sample}$	Sentence embedding	HAC	Linkage = 'complete'
H26	$D_{sample}$	Sentence embedding	HAC	Linkage = 'single'
H27	$D_{sample}$	Sentence embedding	HAC	Linkage = 'average'
H28	$D_{sample}$	Sentence embedding	HAC	Linkage = 'ward'
D1	$D_{sample}$	OHE	DBSCAN	Metric = 'Euclidean'
D2	$D_{sample}$	OHE	DBSCAN	Metric = 'cosine'
D3	$D_{sample}$	Robust scaling + OHE	DBSCAN	Metric = 'Euclidean'
D4	$D_{sample}$	Robust scaling + OHE	DBSCAN	Metric = 'cosine'
D5	$D_{sample}$	Min-max scaling + OHE	DBSCAN	Metric = 'Euclidean'
D6	$D_{sample}$	Min-max scaling + OHE	DBSCAN	Metric = 'cosine'
D7	$D_{sample}$	Standard scaling + OHE	DBSCAN	Metric = 'Euclidean'
D8	$D_{sample}$	Standard scaling + OHE	DBSCAN	Metric = 'cosine'
D9	$D_{sample}$	OHE + PCA	DBSCAN	Metric = 'Euclidean'
D10	$D_{sample}$	OHE + PCA	DBSCAN	Metric = 'cosine'
D11	$D_{sample}$	OHE + Autoencoder	DBSCAN	Metric = 'Euclidean'
D12	$D_{sample}$	OHE + Autoencoder	DBSCAN	Metric = 'cosine'
D13	$D_{sample}$	Sentence embedding	DBSCAN	Metric = 'Euclidean'
D14	$D_{sample}$	Sentence embedding	DBSCAN	Metric = 'cosine'

accuracy, precision, recall, and  $F_1$ -score. To accomplish this, the same dataset inputted during clustering is combined with cluster labels generated, and the resulting dataset was further split into 70% training data and 30% testing data. The training data was inputted to train the LR, while the testing data was utilised to evaluate the performance of the trained LR.

Subsequently, the best-performing clustering model was decided, and its clustering solution was used in further analysis to extract post-clustering insights. This best-performing clustering model was hereinafter referred to as *Clustering<sub>best</sub>*.

Equation (1) showed the formula to calculate the silhouette score of a point  $i$ , where  $a(i)$  was the average distance from  $i$  to other points in the same cluster, and  $b(i)$  was the minimum average distance from  $i$  to points assigned in a different cluster. Silhouette scores of all points would be computed, and their average score would be the Silhouette value.

$$Silhouette(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (1)$$

Equation (2) should be used to compute the corresponding Calinski-Harabasz value for  $k$  clusters

formed and  $n$  data points.  $B(k)$  was the inter-cluster dispersion matrix, whereas  $W(k)$  was the intra-cluster dispersion matrix.

$$\text{Calinski - Harabasz}(k) = \frac{B(k) * (n - k)}{W(k) * (k - 1)} \quad (2)$$

Equations (3–6) showed the equations to compute accuracy, precision, recall, and  $F_1$ -score, respectively. Alongside these metrics, the evaluation incorporates vital measurements such as True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). TP signifies the total number of correctly identified positive cases, and TN describes the total number of correctly predicted negative cases. FP occurs when a positive case is falsely predicted to be negative. FN arises when a negative case is incorrectly identified as positive.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (5)$$

$$F1 = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

### 3.4. Post-clustering insights extraction

Regarding the second study objective, this study leveraged a combination of classification, SHAP, and manual analysis techniques to extract insights from the clustering solution generated by  $\text{Clustering}_{best}$ . The feature importance evaluation before manual analysis was vital to effectively reduce the extensive resources required due to the high dimensionality of  $D_{analytical}$  or  $D_{sample}$ . If  $\text{Clustering}_{best}$  was a k-Means model, then  $D_{analytical}$  was leveraged for this section. Otherwise,  $D_{sample}$  was utilised for this section.

Either  $D_{analytical}$  or  $D_{sample}$  was combined with cluster labels generated by  $\text{Clustering}_{best}$ , and the resulting dataset was further split into 70% training data and 30% testing data. Three classifiers, namely LR, RF, and LGBM, were trained to predict cluster labels from the training data. The trained classifiers were then evaluated with the testing data, using evaluation metrics, including accuracy, precision, recall, and  $F_1$ -score. Then, the best-performing classifier for distinguishing between clusters was decided and hereinafter referred to as  $\text{Classifier}_{best}$ .

Subsequently, a suitable SHAP explainer object was instantiated based on the type of  $\text{Classifier}_{best}$ . If the  $\text{Classifier}_{best}$  was RF or LGBM, the suitable SHAP

explainer object was `shap.TreeExplainer`. On the other hand, if the  $\text{Classifier}_{best}$  was LR, then the suitable SHAP explainer object was `shap.LinearExplainer`. This study incorporated the SHAP framework for the feature importance evaluation task because LR was not easily interpretable.

Later, this study identified “ $k$  most important features,” where  $k$  is the number of clusters found in the clustering solution of  $\text{Clustering}_{best}$ .

Manual analysis was then performed. For each feature within the “ $k$  most important features,” this study calculated the corresponding central tendency across different clusters. Specifically, the mean value was computed for a numerical feature, while the mode value was obtained for a categorical feature. The mean value is derived from the average level of engagement or interaction exhibited by app users from a cluster with that app page. On the other hand, the mode value implied the most frequent category within each cluster.

Then, this study compared these central tendencies across distinct clusters to identify unique user behaviours. A higher mean screen time might imply a more vital interest or necessity for that app page or relevant services. On the other hand, the distinctions between mode values described demographic differences, such as a predominance of young teenagers against senior citizens, to highlight unique cluster identities.

Finally, this study summarised a series of unique behaviours or characteristics displayed by each cluster of users. Such persona encapsulated the behavioural patterns of cluster members, thus effectively providing a comprehensive profile.

## 4. Findings

This section illustrates findings extracted from EDA, comparisons between clustering models, and insights from the clustering solution of  $\text{Clustering}_{best}$ .

### 4.1. Exploratory data analysis (EDA)

Figure 2 shows boxplots of two behavioural features in  $D_{analytical}$ . All behavioural features had similar patterns in their respective boxplots, where a clear “box” was absent in every case. Almost all non-zeroes were treated as upper outliers relative to the distribution. This supported the fact that  $D_{analytical}$  was highly sparse to an extent such that all non-zeroes were treated as upper outliers. However, these outliers were not faulty values and should not be simply removed.

Figure 3 shows a heatmap visualising the correlation matrix for all pairs of numerical features from *D\_analytical*. Inspired by Tissera et al. (2024), this study considered two features to be highly correlated if their correlation coefficient's absolute value was greater than or equal to 0.75. From the heatmap shown, the highest absolute correlation coefficient was 0.20. Hence, this study concluded that no

strongly correlated features existed. No feature selection was made from this perspective.

## 4.2. Comparisons between clustering models constructed

Table 10 shows the performance of all clustering models constructed. The metrics, number of clusters generated,  $k$ , and the corresponding cluster proportions were displayed. All numerical values except Calinski-Harabasz scores were rounded to two decimal points. Calinski-Harabasz scores were rounded off to the nearest whole number.

Furthermore, some HAC models, including H1, H2, H3, H4, H5, H6, H7, H8, H12, H13, H14, H15, H16, H17, H18, H19, H22, H23, and H26 were unable to evaluate because their  $k$  values were visually not interpretable from their respective dendrograms.

In addition, several DBSCAN models, including D1, D2, D3, D7, D8, D10, D13, and D14, were unable to be evaluated because either zero or only one non-outliers cluster was successfully formed. For the remaining DBSCAN models, the outlier cluster, if any, was ignored when computing Silhouette, Calinski-Harabasz scores, and cluster proportions.

K2, K4, K5, K6, and H20 seemed to produce good clustering solutions because an extremely high Silhouette score of 0.99 was obtained for all of

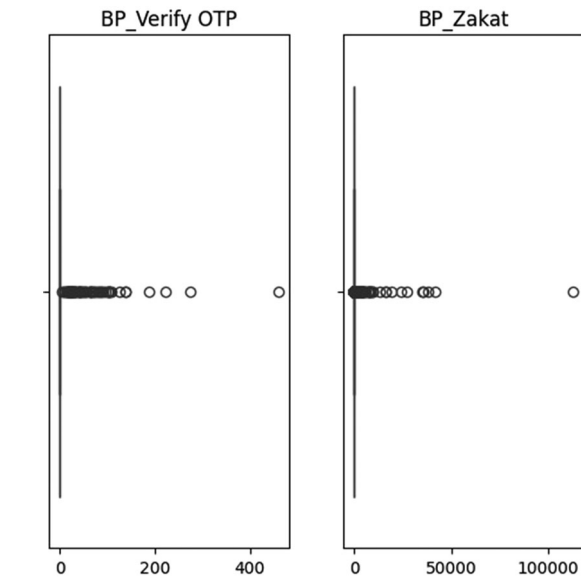


Figure 2. Sample boxplots.

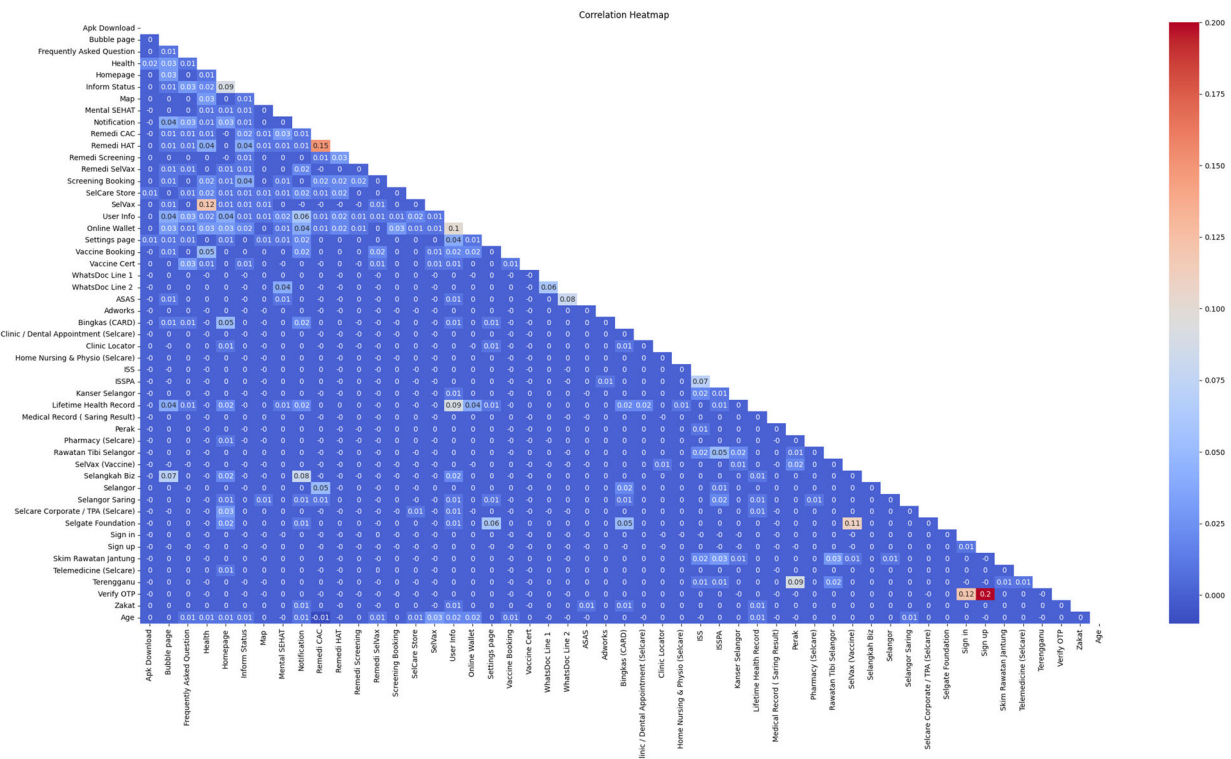


Figure 3. Correlation heatmap.

**Table 10.** Performance comparison between clustering models.

Model code	$k$	Silhouette	Calinski-Harabasz	Accuracy	Precision	Recall	F <sub>1</sub> -score	Cluster proportions
K1	6	0.98	42056	1.00	1.00	1.00	1.00	[0.99, 0.00, 0.00, 0.00, 0.00, 0.00]
K2	6	0.99	69455	1.00	1.00	1.00	1.00	[1.00, 0.00, 0.00, 0.00, 0.00, 0.00]
K3	5	0.80	897625	1.00	1.00	1.00	1.00	[0.30, 0.25, 0.21, 0.16, 0.08]
K4	8	0.99	11207	1.00	1.00	1.00	1.00	[1.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00]
K5	6	0.99	54680	1.00	1.00	1.00	1.00	[1.00, 0.00, 0.00, 0.00, 0.00, 0.00]
K6	4	0.99	341558	0.99	1.00	0.99	1.00	[1.00, 0.00, 0.00, 0.00]
K7	4	0.29	180116	1.00	1.00	1.00	1.00	[0.28, 0.27, 0.23, 0.21]
H9	4	0.47	356	1.00	1.00	1.00	1.00	[0.45, 0.31, 0.24, 0.00]
H10	2	0.49	5	1.00	1.00	1.00	1.00	[1.00, 0.00]
H11	3	0.44	5	1.00	1.00	0.99	1.00	[1.00, 0.00, 0.00]
H12	2	0.45	715	1.00	1.00	1.00	1.00	[0.55, 0.45]
H20	2	0.99	965	1.00	1.00	1.00	1.00	[1.00, 0.00]
H21	2	0.98	428	1.00	1.00	1.00	1.00	[1.00, 0.00]
H24	2	0.98	428	1.00	1.00	1.00	1.00	[1.00, 0.00]
H25	2	0.43	90	0.98	0.97	0.98	0.98	[0.98, 0.02]
H27	2	0.43	93	0.98	0.97	0.98	0.98	[0.98, 0.03]
H28	2	0.26	371	1.00	1.00	1.00	1.00	[0.52, 0.48]
D4	3	-0.29	1	0.98	0.98	0.98	0.98	[0.21, 0.07, 0.07]
D5	3	0.92	17738	1.00	1.00	1.00	1.00	[0.21, 0.20, 0.07]
D6	4	0.84	5327	1.00	1.00	1.00	1.00	[0.30, 0.23, 0.15, 0.14]
D9	2	0.98	88184	1.00	1.00	1.00	1.00	[0.08, 0.03]
D11	2	0.73	376	1.00	1.00	1.00	1.00	[0.08, 0.03]
D12	4	0.36	5	0.97	0.97	0.97	0.97	[0.87, 0.03, 0.02, 0.01]

them. Notably, these clustering models also yield similar cluster proportions, in which the biggest cluster contained almost all data points. In other words, almost all users were grouped within the same cluster, and the remaining users were dispersed within several extremely small clusters. Such a clustering solution was typically standard or common when a sparse dataset was clustered, where the clustering solution often consisted of a dominant cluster predominantly of zero data points, and the non-zero data points were dispersed or poorly grouped around the remaining clusters. In short, K2, K4, K5, K6, and H20 did not generate superior clustering solutions when handling the sparse  $D_{analytical}$  or  $D_{sample}$  despite their high Silhouette scores. Apart from these clustering models, others, such as K1, H10, H11, H21, H24, H25, and H27 generated similar clustering solutions as well. Furthermore, D9 and D11 produced a clustering solution with a dominant outlier cluster and two small clusters. Such a clustering solution should be considered similar to the aforementioned scenario as well. Therefore, K1, K2, K4, K5, K6, H10, H11, H20, H21, H24, H25, H27, D9, and D11 should be excluded in the further analysis.

Besides that, D4's clustering solution achieved a negative Silhouette score of -0.29. Such a negative Silhouette score implied that D4's clustering solution was invalid because higher similarity was observed between users from different clusters than users from the same cluster. Hence, D4 was ignored in the further analysis.

The remaining clustering models included D5, D6, K3, H9, H12, D12, K7, and H28. These models were

arranged descendingly based on their respective Silhouette scores. Here, Silhouette scores were prioritised over Calinski-Harabasz scores to avoid bias introduced when computing a Calinski-Harabasz score due to involving two datasets of different sizes. A bigger dataset size would generally lead to a higher Calinski-Harabasz score (Solorio-Fernández et al., 2016). Scores such as accuracy, precision, recall, and F<sub>1</sub>, which were used for the evaluation of the trained LR to distinguish between clusters, were less emphasised than the Silhouette as well because these scores obtained by the clustering models were relatively similar.

According to the flow of methods, the *Clustering<sub>best</sub>* should be identified so that its clustering solution could be utilised for extracting post-clustering insights. Therefore, D5 was initially appointed as *Clustering<sub>best</sub>* but its clustering solution was found to be uninformative in creating descriptive personas according to users' behaviours. Thus, this study had to step back and re-select an appropriate *Clustering<sub>best</sub>*. Unfortunately, similar results were yielded when leveraging clustering solutions of D6, K3, H9, and H12, in which their clustering solutions were uninformative. Here, this study supported the work of (Hennig, 2015), which claimed that internal cluster validation indices such as Silhouette and Calinski-Harabasz effectively defined how high the quality of clusters formed but not how useful these clusters were in terms of providing practical insights. Conclusively, this study decided *Clustering<sub>best</sub>* to be D12 with the  $k_{best}$  of 4. Subsequently, its clustering solution would be utilised to extract post-clustering insights.



Concerning the first study objective, the optimal clustering model for handling the sparse  $D_{sample}$  was identified as D12, a specialised DBSCAN model instantiated to use cosine similarity in terms of similarity measure and leveraging a combination of several data pre-processing techniques such as feature encoding via OHE and data representation learning with a Multi-Layer Perceptron autoencoder. In this case, the data representation learning approach effectively mitigated the data sparsity issue by transforming the sparse dataset to a dense format pre-clustering. Additionally, cosine similarity's greater flexibility yielded an optimized clustering solution.

#### 4.3. Insights from clustering solution of $Clustering_{best}$

LR, RF, and LGBM classifiers were used separately to predict cluster labels by  $Clustering_{best}$ . Later, the trained classifiers were evaluated using evaluation metrics such as accuracy, precision, recall, and  $F_1$ -score. Each classifier's training and testing performances were checked to detect overfitting issues.

Table 11 shows a summary of comparisons between the constructed classifiers. All values were rounded off to two decimal places. Notably, all classifiers were not overfitting. Overall, this study decided LR as  $Classifier_{best}$ .

Since  $Classifier_{best}$  was LR, the suitable SHAP explainer object would be `shap.LinearExplainer`. After applying the relevant explainer object to LR, the feature importance of independent variables in  $D_{sample}$  was computed in terms of SHAP mean scores. Figure 4 shows a SHAP summary plot to visualise the feature importance analysis's result. Since  $Clustering_{best}$  resulted in four clusters ( $k_{best} = 4$ ), the four most important features were identified from the SHAP summary plot. In other words, central tendencies of "Bubble page," "Homepage," "Online Wallet," and "User Info" would be analysed later for persona generation purposes because these features had the highest SHAP scores, which implied more interesting value variations or deviations to analyse and interpret.

Subsequently, this study computed the required central tendencies and presented them as summary statistics, as in Table 12. All values were rounded off

to two decimal places. Finally, each cluster's descriptive persona was created to describe that group of app users. Each persona was assigned a descriptive name as well.

Cluster 0 was named "Active User." This group of users actively interacted with the app. To support this claim, the mean screen time of "Homepage," "Online Wallet," and "User Info" was the highest compared to mean values of the same features from all other clusters. Since they were active users, they were most probably aware of the latest app-related news or updates. Therefore, advertising strategies for boosting their interests were ineffective and should not be applied to them. Instead, conditional rewards should be offered to maintain their loyalty to the app. For example, the company could integrate with restaurants to offer discounts when using the "Online Wallet" to pay the bills.

Cluster 1 was named "COVID-19 Preventer." This group of users displayed extremely high screen time on the "Bubble page" compared to other features from the same cluster and users from other clusters. "Bubble" is a service introduced during the COVID-19 era. It offered various functionalities, such as checking in when entering shops or restaurants, as a

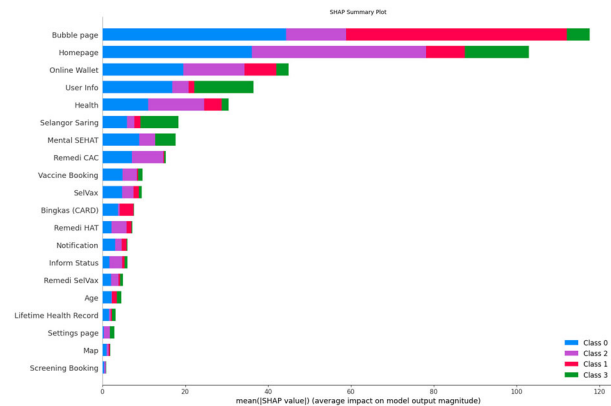


Figure 4. SHAP summary plot of  $Classifier_{best}$ .

Table 12. Summary statistics of  $Clustering_{best}$ .

Cluster	Bubble page	Homepage	Online wallet	User info
0	12.09	<b>63.99</b>	<b>43.37</b>	<b>108.33</b>
1	<b>719.46</b>	17.23	6.25	11.80
2	2.23	1.34	24.15	28.79
3	3.25	32.73	21.10	43.09

Table 11. Performance comparison between classifiers.

Classifier	Training results				Testing results			
	Accuracy	Precision	Recall	$F_1$ -score	Accuracy	Precision	Recall	$F_1$ -score
LR	1.00	1.00	1.00	1.00	0.97	0.97	0.97	0.97
RF	1.00	1.00	1.00	1.00	0.96	0.97	0.96	0.95
LGBM	1.00	1.00	1.00	1.00	0.96	0.96	0.96	0.96

mandatory action enforced by the government during the COVID-19 era. Furthermore, each "Bubble" signified a "social circle." Users could include other users in the same "Bubble." For example, Haziq's family members could participate in the "Bubble" named "The Haziq's Family." As a matter of fact, interactions between individuals should be avoided to reduce the virus spread during the COVID-19 era. Therefore, by adopting the "Bubble" concept and service, users were committed to only interacting with certain individuals. In other words, users had their social connections in moderation. Users from the "COVID-19 Preventer" cluster actively performed COVID-19 precautions, attempting to mitigate this crisis as soon as possible. These users had a high awareness of minimizing medical-related crises. Consequently, more medical-related advertising campaigns should be targeted to expand their awareness in preventing more health issues.

Cluster 2 was named "Inactive User." This group of users showed the lowest screen time among all four clusters. Reach-out campaigns should be carried out to understand what contributed to their low level of interaction with the app.

Cluster 3 was named "Average Joe." This group of users exhibited an average level of interaction with the app based on their screen time. In addition, they mostly browsed app pages such as "Homepage," which were considered general-related. They might be unaware of relatively more domain-specific services such as "Online Wallet" which were finance-related. In fact, the app also offered health-related services like medical screening and COVID-19 vaccination. Hence, related advertising information should be provided to this group of users so that their interests might be expanded to more domains available in the app.

Concerning the second study objective, post-clustering insights were extracted from the *Clustering<sub>best</sub>* using a combination of techniques such as classification, SHAP, and manual analysis. The insights were presented as descriptive personas and relevant marketing strategies were suggested for distinct personas.

#### 4.4. Limitation

A key limitation of this study lies in the process of creating user personas via manual analysis. While this study was able to derive distinct descriptive personas, this approach might not always be useful. In scenarios where the value differences between central tendencies across varying clusters are less

meaningful, persona generation becomes challenging. This limitation points to the potential issue of lack of generalisability. In such cases, domain experts could be included to better analyse the clustering solution.

## 5. Conclusion

This study set forth two objectives: (1) to identify an optimal clustering model that can handle a sparse dataset and (2) to extract post-clustering insights via a descriptive persona for each cluster. The findings of this study showed that a specialised DBSCAN model instantiated to use cosine similarity as the similarity measure and augmented with data pre-processing tasks such as OHE and data representation learning through a Multi-Layer Perceptron autoencoder, was most effective when clustering the utilised sparse dataset. The use of data representation learning techniques and cosine similarity as the suitable similarity measure significantly boosted the clustering performance when dealing with a sparse dataset. Subsequently, this study leveraged a mixed approach encompassing classification, SHAP, and manual analysis for extracting post-clustering insights. As a result, insights were extracted and presented as descriptive personas. The created personas were "Active User," "COVID-19 Preventer," "Inactive User," and "Average Joe." Nevertheless, further development and usage of personas should be proceeded and monitored with the aid of domain experts to optimize the outcome of this study.

## Acknowledgment

The dataset used in this study was provided by Hayat Technologies Sdn Bhd.

## Author contributions

Li-Yoong Ooi performed the experiments and drafted the manuscript. Choo-Yee Ting, Helmi Zakariah, and Eashvaren Chandar reviewed and improved the manuscript. All authors approved the final version of the manuscript.

## Disclosure statement

No potential competing interests were reported by the authors.

## Funding

No funding was provided for this study.

## About the authors



**Li Yoong Ooi** is currently a degree student attached to the Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia. He is mainly interested in the application of machine learning techniques for behaviour recognition. While still working in this area, his other research interests include Information Systems and Applied Statistics.



**Choo-Yee Ting** is currently a professor attached to the Faculty of Computing and Informatics, Multimedia University, Cyberjaya, Malaysia. In year 2002, Choo-Yee Ting was awarded the Fellow of Microsoft Research by Microsoft Research Asia, Beijing, China. He has been active in research projects related to predictive analytics and Big Data funded by MOE, MOSTI, Telekom Malaysia, MDeC, and industries. In year 2014, he and his team members won two national level Big Data Analytics competitions. Prof. Ting has been providing training and consultancy in data science to JPA, MDeC, INTAN, and companies. In year 2020, he was invited to support the Selangor COVID-19 initiative through SELANGKAH and in year 2021, he was appointed by the Malaysia government to support the National COVID-19 Immunisation Programme. In 2022, he was appointed by the government to analyze data about healthcare work culture in Malaysia. Currently, he is active in the Malaysia's National Planetary Health roadmap.



**Dr. Mohd Helmi Zakariah** is the CEO of Hayat Technologies, where he leads innovative projects in AI-powered healthcare systems and digital epidemiology. He also serves on the UN Global Initiative for AI in Health, is a board member of the South Asia Field Epidemiology & Technology Network (SAFETYNET), and is a Commissioner at Chatham House for Universal Health Coverage. Previously, he was the Chief Information Officer at Selangkah Ventures, driving the development of digital health platforms for disease surveillance and precision screening. Dr. Helmi has been a key contributor to the Selangor Public Health Advisory Council and the Care Economy Consultative Council, shaping public health policy and strategies. As a prolific researcher, he has authored impactful studies on COVID-19 data stability using machine learning, geospatial analytics for active case detection, and risk profiling for non-communicable diseases. Dr. Helmi holds a Doctor of Medicine from Volgograd State Medical University and a Master of Public Health from the University of Liverpool.

**Eashvaren Chandar** holds a First Class Bachelor's degree in Computer Science with a specialization in Data Science from Multimedia University, Cyberjaya. Currently, he works as a Data Scientist at Hayat Technologies Sdn. Bhd., where he focuses on applying data-driven solutions to complex problems. His primary research interests include natural language processing (NLP) and geospatial analysis, with a focus on providing data-driven solutions through analytical dashboards to support decision-making processes. Eashvaren recently submitted a paper that enhances sentiment analysis by incorporating objectivity and subjectivity text classification for BERT embedding models. In addition to his research work, he developed an interactive dashboard for the Selangor state government to monitor and support data-driven decision-making related to the supply and demand of care economy policies. Eashvaren is also an active member of the research group that strategizes and plans policy writing for the care economy, helping shape important frameworks for sustainable economic development.

## ORCID

Choo-Yee Ting  <http://orcid.org/0000-0001-5667-2816>

## Data availability statement

The data that support the findings of this study are available from one of the authors, Dr. Helmi, upon reasonable request.

## References

- Abul, O., Lo, A., Alhajj, R., Polat, F., & Barker, K. 2003. Cluster validity analysis using subsampling. *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483)*, Washington, DC, USA, 2003, pp. 1435–1440 vol.2, doi: [10.1109/ICSMC.2003.1244614](https://doi.org/10.1109/ICSMC.2003.1244614).
- Alghamdi, A. (2023). A hybrid method for big data analysis using fuzzy clustering, feature selection and adaptive neuro-fuzzy inferences system techniques: Case of mecca and medina hotels in Saudi Arabia. *Arabian Journal for Science and Engineering*, 48(2), 1693–1714. <https://doi.org/10.1007/s13369-022-06978-0>
- Alves Gomes, M., & Meisen, T. (2023). A review on customer segmentation methods for personalized customer targeting in e-commerce use cases. *Information Systems and E-Business Management*, 21(3), 527–570. <https://doi.org/10.1007/s10257-023-00640-4>
- An, N. (2020). Analysis on market segmentation in advertising companies. In *International Conference Proceeding Series, Association for Computing Machinery* (pp. 126–129). University of California. <https://doi.org/10.1145/3436209.3436887>
- Baumgarte, F., Brandt, T., Keller, R., Röhrich, F., & Schmidt, L. (2021). You'll never share alone: Analyzing carsharing user group behavior. *Transportation Research Part D*:

- Transport and Environment*, 93, 102754. <https://doi.org/10.1016/j.trd.2021.102754>
- Ben-Gal, I., Weinstock, S., Singer, G., & Bambos, N. (2019). Clustering users by their mobility behavioral patterns. *ACM Transactions on Knowledge Discovery from Data*, 13(4), 1–28. <https://doi.org/10.1145/3322126>
- Chang, D., Zhao, J., Zou, F., & Xu, G. (2020). A user segmentation approach for UGC platform based on a new lead user identification index system and K-means clustering. In *IEEE International Conference on Industrial Engineering and Engineering Management IEEE Computer Society* (pp. 954–958). IEEE. <https://doi.org/10.1109/IEEM45057.2020.9309940>
- Chen, J., Zhu, J., Jiang, H., Yang, H., & Nie, F. (2023). Sparsity fuzzy C-means clustering with principal component analysis embedding. *IEEE Transactions on Fuzzy Systems*, 31(7), 2099–2111. <https://doi.org/10.1109/TFUZZ.2022.3217343>
- Chopra, H., Sinha, A. R., Choudhary, S., Rossi, R. A., Indela, P. K., Parwatala, V. P., Paul, S., and Maiti, A. (2023). Delivery optimized discovery in behavioral user segmentation under budget constraint. In *International Conference on Information and Knowledge Management, Proceedings, Association for Computing Machinery* (pp. 359–368). Association for Computing Machinery. <https://doi.org/10.1145/3583780.3614839>
- Egorova, E., Glukhov, G., & Shikov, E. (2022). Customer transactional behaviour analysis through embedding interpretation. *Procedia Computer Science*, 212, 284–294. <https://doi.org/10.1016/j.procs.2022.11.012>
- Fard, M. M., Thonet, T., & Gaussier, E. (2020). Pattern recognition letters deep k-means: Jointly clustering with k-Means and learning representations. <https://www.elsevier.com/open-access/userlicense/1.0/>
- Guidotti, R., & Gabrielli, L. (2018). Recognizing residents and tourists with retail data using shopping profiles. In *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICTS*. Springer Verlag. pp. 353–363. [https://doi.org/10.1007/978-3-319-76111-4\\_35](https://doi.org/10.1007/978-3-319-76111-4_35)
- He, X., & Chen, Y. (2021). Understanding structural hole spanners in location-based social networks: A data-driven study. In *UbiComp/ISWC 2021 – Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers* (pp. 619–624). Association for Computing Machinery Inc. <https://doi.org/10.1145/3460418.3480398>
- Hennig, C. (2015). What are the true clusters? *Pattern Recognition Letters*, 64, 53–62. <https://doi.org/10.1016/j.patrec.2015.04.009>
- Hung, P. D., Thuy Lien, N. T., & Ngoc, N. D. (2019). Customer segmentation using hierarchical agglomerative clustering. In *ACM International Conference Proceeding Series, Association for Computing Machinery* (pp. 33–37). ICIS. <https://doi.org/10.1145/3322645.3322677>
- Irawan, E., Mantoro, T., Ayu, M. A., Catur Bhakti, M. A., & Permana, I. K. Y. T. (2020). Analyzing reactions on political issues in social media using hierarchical and K-means clustering methods. In *6th International Conference on Computing, Engineering, and Design, ICCED 2020*. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ICCED51276.2020.9415839>
- Jansen, B. J., Jung, S. G., Salminen, J., An, J., & Kwa, H. (2017). Leveraging social analytics data for identifying customer segments for online news media. In *Proceedings of IEEE/ACS International Conference on Computer Systems and Applications, AICCSA* (pp. 463–468). IEEE Computer Society. <https://doi.org/10.1109/AICCSA.2017.64>
- Ji, Y., Gao, S., Kruse, J., Huynh, T., Triveri, J., Scheele, C., Bennett, C., and Wen, Y. (2022). Exploring multilevel regularity in human mobility patterns using a feature engineering approach: A case study in Chicago. In *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*. Association for Computing Machinery. <https://doi.org/10.1145/3557915.3561007>
- Karaliopoulos, M., Tsolas, L., Koutsopoulos, I., Haldiki, M., Van Hove, S., and Conradie, P. (2022). Beyond clustering: Rethinking the segmentation of energy consumers when nudging them towards energy-saving behavior. *Energy Informatics Review*, 2(4), 28–43.
- Kingma, D. P., & Lei Ba, J. 2014. Adam: a method for stochastic optimization. *CoRR*, abs/1412.6980.
- Korenus, T., Laurikkala, J., & Juhola, M. (2007). On principal component analysis, cosine and Euclidean measures in information retrieval. *Information Sciences*, 177(22), 4893–4905. <https://doi.org/10.1016/j.ins.2007.05.027>
- Lee, Y., & Cho, S. (2021). User segmentation via interpretable user representation and relative similarity-based segmentation method. *Multimedia System*, 27(1), 61–72. <https://doi.org/10.1007/s00530-020-00702-4>
- Lee, Y., Park, I., Cho, S., & Choi, J. (2018). Smartphone user segmentation based on app usage sequence with neural networks. *Telematics and Informatics*, 35(2), 329–339. <https://doi.org/10.1016/j.tele.2017.12.007>
- Li, T., Li, Y., Zhang, M., Tarkoma, S., & Hui, P. (2023). You are how you use apps: User profiling based on spatio-temporal app usage behavior. *ACM Transactions on Intelligent Systems and Technology*, 14(4), 1–21. <https://doi.org/10.1145/3597212>
- McConville, R., Santos-Rodríguez, R., Piechocki, R. J., & Craddock, I. (2020). N2D: (not too) deep clustering via clustering the local manifold of an autoencoded embedding. In *Proceedings International Conference on Pattern Recognition* (pp. 5145–5152). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ICPR48806.2021.9413131>
- Mehta, V., Mehra, R., & Verma, S. S. (2021). A survey on customer segmentation using machine learning algorithms to find prospective clients. In *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions), ICRITO 2021*. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ICRITO51393.2021.9596118>
- Munusamy, S., & Murugesan, P. (2020). Modified dynamic fuzzy c-means clustering algorithm – application in dynamic customer segmentation. *Applied Intelligence*,



- 50(6), 1922–1942. <https://doi.org/10.1007/s10489-019-01626-x>
- Myburg, M., & Berman, S. (2022). Customer lifetime value prediction with K-means clustering and XGBoost. In *Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2022* (pp. 298–302). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ASONAM55673.2022.10068602>
- Nair, V., & Hinton, G. E. 2010. Rectified linear units improve restricted Boltzmann machines. *Proceedings of the 27th International Conference on International Conference on Machine Learning*. 807–814.
- Nandapala, E. Y. L., Jayasena, K. P. N., & Rathnayaka, R. M. K. T. (2020). Behavior segmentation based micro-segmentation approach for health insurance industry. In *Proceedings in ICAC 2020 – 2nd International Conference on Advancements in Computing* (pp. 333–338). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ICAC51239.2020.9357282>
- Natilli, M., Monreale, A., Guidotti, R., & Pappalardo, L. 2019. Exploring students eating habits through individual profiling and clustering analysis. In: Alzate, C., (eds.), *ECML PKDD 2018 Workshops. MIDAS PAP 2018*. Lecture Notes in Computer Science(), vol 11054. Springer, Cham. [https://doi.org/10.1007/978-3-030-13463-1\\_12](https://doi.org/10.1007/978-3-030-13463-1_12)
- Nguyen, S. P. (2021). Deep customer segmentation with applications to a Vietnamese supermarkets' data. *Soft Computing*, 25(12), 7785–7793. <https://doi.org/10.1007/s00500-021-05796-0>
- Nikolij, A., Dzeroski, S., Munoz, M. A., Doerr, C., Korosec, P., & Eftimov, T. (2023). Algorithm instance footprint: Separating easily solvable and challenging problem instances. In *GECCO 2023 – Proceedings of the 2023 Genetic and Evolutionary Computation Conference* (pp. 529–537). Association for Computing Machinery Inc. <https://doi.org/10.1145/3583131.3590424>
- Nishimi, K., Borsari, B., Marx, B. P., Rosen, R. C., Cohen, B. E., Woodward, E., Maven, D., Tripp, P., Jiha, A., Woolley, J. D., Neylan, T. C., & O'Donovan, A. (2022). Clusters of COVID-19 protective and risky behaviors and their associations with pandemic, socio-demographic, and mental health factors in the United States. *Preventive Medicine Reports*, 25, 101671. <https://doi.org/10.1016/j.pmedr.2021.101671>
- Pai, S., Brennan, F., Janik, A., Correia, T., & Costabello, L. (2022). Unsupervised customer segmentation with knowledge graph embeddings. In *WWW 2022 – Companion Proceedings of the Web Conference* (pp. 157–161). Association for Computing Machinery Inc. <https://doi.org/10.1145/3487553.3524224>
- Paweloszczek, I. (2021). Customer segmentation based on activity monitoring applications for the recommendation system. In *Procedia Computer Science* (pp. 4751–4761). Elsevier B.V. <https://doi.org/10.1016/j.procs.2021.09.253>
- Peker, S., Kocuyigit, A., & Eren, P. E. (2017). LRFMP model for customer segmentation in the grocery retail industry: A case study. *Marketing Intelligence & Planning*, 35(4), 544–559. <https://doi.org/10.1108/MIP-11-2016-0210>
- Rezaei, M., & Franti, P. (2020). Can the number of clusters be determined by external indices? *IEEE Access*, 8, 89239–89257. <https://doi.org/10.1109/ACCESS.2020.2993295>
- Rodríguez, J., Medina-Pérez, M. A., Gutierrez-Rodríguez, A. E., Monroy, R., & Terashima-Marín, H. (2018). Cluster validation using an ensemble of supervised classifiers. *Knowledge Based System*, 145, 134–144. <https://doi.org/10.1016/j.knosys.2018.01.010>
- Sembiring Brahmana, R. W., Mohammed, F. A., Chairuang, K., & Komputer, L. (2020). Customer segmentation based on RFM model using K-means, K-medoids, and DBSCAN methods. *Jurnal Ilmiah Teknologi Informasi*, 11(1), 32. <https://doi.org/10.24843/ikjiti.2020.v11.i01.p04>
- Shen, B. (2021) E-commerce customer segmentation via unsupervised machine learning. In *ACM International Conference Proceeding Series*. Association for Computing Machinery. <https://doi.org/10.1145/3448734.3450775>
- Solorio-Fernández, S., Carrasco-Ochoa, J. A., & Martínez-Trinidad, J. F. (2016). A new hybrid filter-wrapper feature selection method for clustering based on ranking. *Neurocomputing*, 214, 866–880. <https://doi.org/10.1016/j.neucom.2016.07.026>
- Stormi, K., Lindholm, A., Laine, T., & Korhonen, T. (2020). RFM customer analysis for product-oriented services and service business development: An interventionist case study of two machinery manufacturers. *Journal of Management and Governance*, 24(3), 623–653. <https://doi.org/10.1007/s10997-018-9447-3>
- Tissera, M. N. S., Asanka, P. P. G. D., & Rajapakse, R. A. C. P. (2024). Enhancing customer segmentation using large language models (LLMs) and deterministic, independent-of-corpus embeddings (DICE). In *ICARC 2024 – 4th International Conference on Advanced Research in Computing: Smart and Innovative Trends in Next Generation Computing Technologies* (pp. 73–78). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ICARC61713.2024.10499784>
- Umhuza, E., Ntirushwamaboko, D., Awuah, J., & Birir, B. (2020). Using unsupervised machine learning techniques for behavioral-based credit card users segmentation in Africa. *SAIEE Africa Research Journal*, 111(3), 95–101. <https://doi.org/10.23919/SAIEE.2020.9142602>
- Vajjhala, N. R., & Strang, K. D. (2019). Impact of psychodemographic factors on smartphone purchase decisions. In *ACM International Conference Proceeding Series* (pp. 5–10). Association for Computing Machinery. <https://doi.org/10.1145/3394788.3394790>
- Zhong, G., Wang, L. N., Ling, X., & Dong, J. (2016). An overview on data representation learning: From traditional feature learning to recent deep learning. *The Journal of Finance and Data Science*, 2, 265–278. <https://doi.org/10.1016/j.jfds.2017.05.001>
- Zhou, J., Zhai, L., & Pantelous, A. A. (2020). Market segmentation using high-dimensional sparse consumers data. *Expert Systems with Applications*, 145, 113136. <https://doi.org/10.1016/j.eswa.2019.113136>
- Zhu, E., & Ma, R. (2018). An effective partitional clustering algorithm based on new clustering validity index. *Applied Soft Computing Journal*, 71, 608–621. <https://doi.org/10.1016/j.asoc.2018.07.026>
- Zhu, E., Zhang, Y., Wen, P., & Liu, F. (2019). Fast and stable clustering analysis based on Grid-mapping K-means algorithm and new clustering validity index. *Neurocomputing*, 363, 149–170. <https://doi.org/10.1016/j.neucom.2019.07.048>