engineering proceedings

MDPI

Proceeding Paper

CNAIS: Performance Analysis of the Clustering of Non-Associated Items Set Techniques †

Vinaya Babu Maddala * and Mooramreddy Sreedevi

Department of Computer Science, Sri Venkateswara University, Tirupati 517502, India; msreedevi_svu2007@yahoo.com

* Correspondence: mvinayababu@gmail.com

† Presented at the International Conference on Recent Advances on Science and Engineering, Dubai, United Arab Emirates, 4–5 October 2023.

Abstract: Mining technologies depend upon their outcomes, focusing only on certain data features within the database. They select only certain features related to the process from diverse integrated data resources and transform them into a form suitable for mining tasks. Different implementations of mining techniques run on data sources, which may be of considerable volume, to extract different knowledge outcomes suitable for various analyses and decision-making processes. The proposed study provides the design and development of the Clustering of Non-Associated Items set (CNAIS) within a transactional database. The development of the algorithm and its application to the data set are described and the results are noted. Comparisons with state-of-the-art methods show that CNAIS exhibits better performance.

Keywords: performance; association mining; accuracy; precision; recall

Citation: Maddala, V.B.; Sreedevi, M. CNAIS: Performance Analysis of the Clustering of Non-Associated Items Set Techniques. *Eng. Proc.* **2023**, *59*, 14. https://doi.org/10.3390/engproc2023059014

Academic Editors: Nithesh Naik, Rajiv Selvam, Pavan Hiremath, Suhas Kowshik CS and Ritesh Ramakrishna Bhat

Published: 11 December 2023

1. Introduction

Depending on their results, mining technologies concentrate primarily on specific aspects of a database. They take only specific process-related aspects from integrated data resources that contain a variety of data and convert them into a format that is appropriate for mining operations. Different mining techniques are applied to data sources with potentially enormous volumes in order to extract diverse knowledge results suitable for diverse analyses and decision-making processes. These knowledge results are assessed and represented visually using a variety of methods appropriate to the domain, such as tabular forms, decision tree forms, graphs, rules, charts, data cubes, and multi-dimensional graphics. These are categorized into descriptive and prescriptive viewpoints on data mining. According to the general qualities found in a data repository, descriptive mining summarizes or characterizes massive amounts of data. Prescriptive mining is the process of predicting and inferring information from past data. Both forms of data mining include a variety of methods, such as association, clustering, the categorization of data items, exploring outliers, regression and trending analytics, and machine learning techniques. Finding missing values, choosing the right features, and concentrating on outlier detection are just a few of the difficulties involved in mining massive amounts of data. Other challenges involve clustering techniques to find patterns from complex/distributed data, analyzing high dimensional data, identifying imbalanced classes in the classification, protecting data privacy, and leaving data sets with useless data due to algorithm logic.

Mining huge volumes of data is not a simple task, but throws many challenges across broad categories, such as finding missing values, apt feature selection, and focusing on outlier detection. Also, other challenges include methods for clustering high-dimensional data, identifying imbalanced classes in the classification, ensuring data privacy, extracting patterns from complex/distributed data, leaving data unused in data sets due to algorithm logic, etc. Association rule mining (ARM) is one of the widely utilized data mining

methodologies introduced by this study [1]. ARM finds useful correlations in data items, frequent patterns in datasets, and associations among items or casual structures involved in the transaction databases or data repositories [2]. Association rule mining involves the extraction of interesting associations or correlation relationships among various features within a given large set of data items. Day-to-day activities that generate massive amounts of data are continuously collected and stored. Many industries are becoming interested in mining association rules from their databases. Companies rely on decision making processes, such as cross-marketing, market basket analyses, and loss leader analyses, etc., to develop their business strategies. They use association mining to identify correlations in data items among the huge amounts of business transaction records generated daily [3,4]. In the process of mining, interesting relationships among items in a given data set may be found. Rule support and confidence are two measures of rule interestingness that reflect the usefulness and certainty of discovered rules, respectively. Rules are said to be strong when they satisfy both a minimum support threshold (min_support) and a minimum confidence threshold (min_confidence). With experience, such thresholds can be set by users or domain experts.

2. Related Works

The Apriori algorithm proposed, which is used for frequent itemset extraction from transaction data, is a commonly used technique in Market Basket Analysis. It is used to analyze sales in super markets or any sales related business to find associations between the items purchased by customers so that mangers can make better decisions related to product forecasting in terms of sales and profits [5].

There are several steps involved in this process, as mentioned below:

a. Market Basket Analysis to make decisions regarding product stock and purchasing;
b. Recommender Systems to recommend items that are frequently used by customers;
c. Fraud Detection for identifying abnormal transactions;
d. Network Intrusion Detection to identify any abnormal behaviors in access patterns;
e. Medical Analysis to identify diseases by observing patterns;
f. Text Mining to mine texts with frequent phrases or words;
g. Web usage mining to identify frequent visitors to web sites and how are they used.

As an important data mining technique similar to classification, clustering involves grouping a set of data objects when the groupings are unknown. In clustering, grouping produces classes or clusters of objects, where [6] objects within a cluster share many traits but differ greatly from those in other clusters. Dissimilarities are taken into consideration when describing objects based on their attribute values. Distance measurements, centroids, etc., are frequently used. Clustering has applications across many disciplines, including data mining, statistics, biology, and machine learning [7–10].

The grouping of data points based on similar characteristics or nearness among data values is termed clustering. It involves extracting data points that exhibit similar features within a dataset, resulting in the placement of these points into the same cluster.

2.1. K-Means Clustering Technique

K-Means is a well-known clustering approach that is used in machine learning and data analysis to divide a dataset into discrete groups or clusters based on their similarities. K-Means clustering aims to group data points that are close to each other while keeping data points from other clusters reasonably separated. The K-Means clustering algorithm works as follows: K initial cluster centroids are picked at random or on purpose from the dataset. These centroids serve as the focal points for each cluster [11–14].

For huge datasets or a large number of clusters, K-Means can be computationally expensive [15–20]. Outliers can have a major impact on K-Means outcomes. A single outlier may cause the centroid to be pulled away from the main cluster, resulting in poor clustering performance. Techniques such as outlier detection and the use of robust versions of K-Means can help to reduce this problem. K-Means produces predictable results, which means

that running the method numerous times yields the same clusters each time. However, if you are seeking various clusterings to examine the structure of the data, this can be a drawback. You can mitigate this by utilizing K-Means variants, such as K-Means++ or Mini-Batch K-Means. K-Means is primarily intended for numerical data, and it may not perform well with categorical or mixed data types unless adequate preprocessing or distance metric selection is used.

To overcome these constraints, it is critical to evaluate the individual properties of your data, as well as the aims of your clustering task. Other clustering techniques, such as hierarchical clustering, DBSCAN, Gaussian Mixture Models (GMMs), or spectral clustering, may be more suited and successful depending on the nature of the data. Furthermore, in some circumstances, preprocessing steps, including careful analysis of the distance metric and initialization techniques, may increase the performance of K-Means.

2.2. Cross Industry Standard Process in Data Mining (CRISP-DM)

The phases of Cross Industry Standard Process in Data Mining (CRISP-DM) are shown in Figure 1.

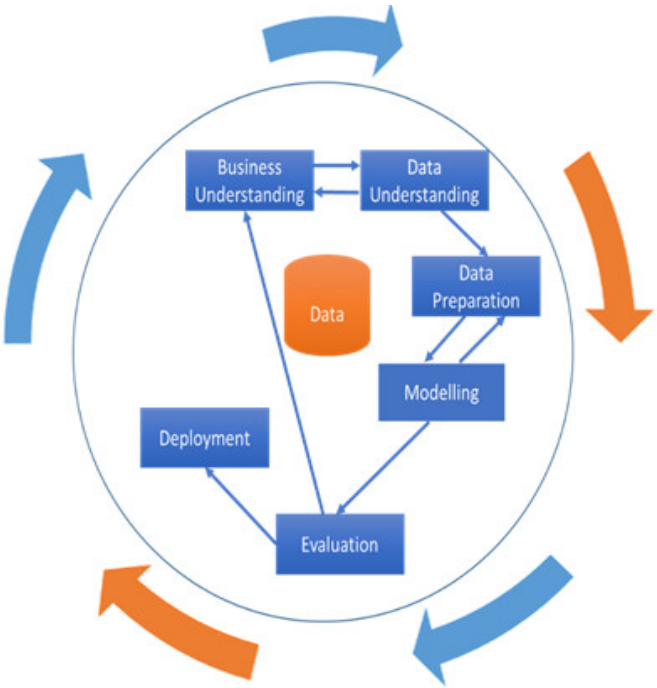

Figure 1. Phases in CRISP-DM.

3. Methodology

In studying the above association and clustering techniques, we propose the Clustering Technique of Non-Association rule Items Set (CNAIS) method over the transaction data sets method, which considers non-association rule item sets to cluster them along with associated rule-based sets or independently to prevent the loss of data items or rows in the database, which might occur as unused data after undergoing filtering in either ARM or clustering processes after long computations.

The proposed algorithm, Clustering Technique of Non-Association rule Items Sets over transaction data sets (CNAIS), works in three phases as shown in Algorithm 1:

(a) Extracting Non-Associative Rule Sets (S_{Na}) from the transaction dataset (T_d);
(b) Computation of the threshold value and the application of the clustering technique over items of both S_A and S_{NA} to form clusters, which will enable us to group transactions (T_i) in the T_d according to the T-value;
(c) Based on different T-Values we can prioritize transactions (T_i) in the T_d.

In Phase 1, we used the function ENAIS (T_d, S_A, and S_{NA}), as shown below.

Algorithm 1: ENAIS (T_d, S_A, and S_{NA})

Inputs: Transaction Database T_d containing set $T\{t_1,t_2,t_3 \ldots t_n\}$
Returns*:* S_A,S_{NA}
1. Begin;
2. For each T_i in T_d perform
   2.1 Count occurrences of I_k
   2.2 Find SupportCount *(Sc)*
   2.3 S_A = *[set of items* $\geq$ *Sc]*
   2.4 S_{NA}= *[set of items* $<$ *Sc]*
   End of step 2;
3. Repeat until *count_itemset*(S_A) = *Prev_Count* for each item in S_A
   (Loop until no more items satisfy)
   3.1 Prev_count = count_itemset(S_A)
   (Item set count before generating the next new item sets)
   3.2 Calculate SupportCount*(Sc)*
   3.3 Obtain Subset S_k that satisfies the Support-count*(*S_c*)*
   3.4 Combine items in $S_A = S_k \cup S_{Ak-1}$
   (For the generation item sets of size k that satisfy Support-count (Sc));
4. S_{NA}=$T_d - S_A$;
5. Return S_A,S_{NA}.

In Phase 2, we used the function GenerateCluster(S_A,S_{NA}) to create clusters which contained fewer outliers as shown in Algorithm 2.

Algorithm 2:*GenerateCluster(,*S_A,S_{NA}*)*

Inputs: SA = { ia_1,ia_2,. . ..ia_n} is the set of associated item sets, which is a subset of I.
SNA = { ina_1,ina_2,. . ..ina_n} is the set of non-associated item sets, which is a subset of I.
Returns: Clusters $K_1..K_m$*(contains* S_A,S_{NA}*)*
1. Begin;
2. Select attribute A_{it}, which will be used as the T-Value(T_v) for the selection of the item from S_A,S_{NA};
3. Arrange in ascending order of A_{it} in S_A and find Count C_n;
4. Find the median of values M_{SA};
5. Repeat steps 3 and 4 for S_{NA} and find M_{SNA};
6. T_v= M_{SA} + M_{SNA}/2;
7. Choose T_V as the threshold value for creating clusters;
8. Initialize the points randomly $k_1 \ldots k_m \ldots$ as the number of clusters required from A_{it} of S_A,S_{NA} such that the value is nearest to that of the support count (+ or - Sc%) and the T_V (threshold value) from S_A and S_{NA} data points as the medoids;
9. For each of the points chosen from step 8, create from n objects in S_A and
10. For all the other non-medoids in each k_m, compute the cost (distance as computed using the Manhattan/Euclidean method) from the initial medoid;
11. In each k_i^{th} cluster, compare each medoid to that of the k_{i+1}^{th} cluster, select the minimum distance (values related to SC% of threshold value Tv), and form clusters with S_A and S_{NA};
12. Compute the total cost of min medoids in k_i^{th} and assign to Dk_{i+1} ie D_{ki} = Σ(min(ki)) and Dk_{i+1} = Σ(min(k_{i+1}));
13: Compute $S_1 = |D_k - D_{k+1}|$;
14. Repeat the steps for the k_i^{th} and k_{i+2}^{th} clusters from step 11 to step 13 and find S_2;
15. Compute Z = S1-S2;
16. If (z < 0) then:
Swap the initial medoids to the next random medoid and repeat steps 8 to 15 until the clusters do not change, or clusters k_i, k_{i+1}, k_{i+2} are perfect;
Return $k_i..k_m$ clusters;
End.

In Phase 3, we used the Algorithm 3 to generate clusters $K_1 \ldots K_m$, where 1...m are priority-based clusters that are created based on the threshold value.

Algorithm 3: PriorityClusterGenerate(k1..Km)

Input: Clusters $K_1 \ldots K_m$ formed using S_A and S_{NA} based on T_V which depends on A_{it}
Output: Cluster graphs with non-associated items selected and priority-based selection.
1. Begin;
2. For each Pair(K_i,K_{i+1}),
   (a) Compute x = $\Sigma \mid k_i med + k_{i+1} med \mid$ and y = $k_{i+2} med$
   (b) Plot scatted graph G_i;
3. G_i gives S=non-associated items selected for S_A and S_{NA} satisfying the threshold value T_V
   Prioritize the cluster points based on T_v;
4. Plot the graph using clusters K_i/K_{i+1} K_i/K_{i+2}. Threshold value Tv clusters are formed with different points with T_v + Sc% of A_{it};
End.

4. Results and Discussion

Data Preparation

The sample Ttansaction dataset included features such as TransId, considering purchases of up to six items on different dates. The items dataset with prices was considered from Table 1.

Table 1. Items available in the hotel which were present in the transaction file.

Item No.	Item Name	Price
1	BIRIYANI	300
2	PIZZA	270
3	COOL-DRINK	85
4	FISH-FRY	200
5	GULAB-JAM	70
6	HALWA	60
7	TANDOORI	250
8	VEG-RICE	200
9	MUTTON-FRY	230
10	SANDWICH	120
11	LEMON-SODA	70
12	ICE-CREAM	150
13	SAMOSA	50
14	CHICKEN-FINGERS	150

Some of the frequently purchased items were selected from the above transactions and are noted in Table 2.

Table 2. Frequently purchased items in the transactions given.

Transaction ID	Item Sets
10016	Biryani, Pizza, Cool drink, Fish fry, and Ice-Cream
31001	Gulab jam, Biryani, Tandoori, Halwa, Sandwich, Mutton fry, and Lemon soda
90121	Pizza, Cool drink, Biryani, Veg rice, Samosa, and Ice-Cream
50091	Biryani, Gulab jam, and Halwa
69091	Veg Rice, Lemon soda, Cool drink, Ice-Cream, Pizza, and Biryani
10909	Sandwich, Chicken fingers, Mutton fry, and Cool drink

CNAIS was implemented using Python. Figure 2 shows the clusters formed with non-associated items added with their threshold values. Figure 3 shows the cluster of points showing transaction IDs with the corresponding purchase amounts. Figure 4 shows

the cluster of points showing transaction IDs with the corresponding purchase amounts and their threshold values.

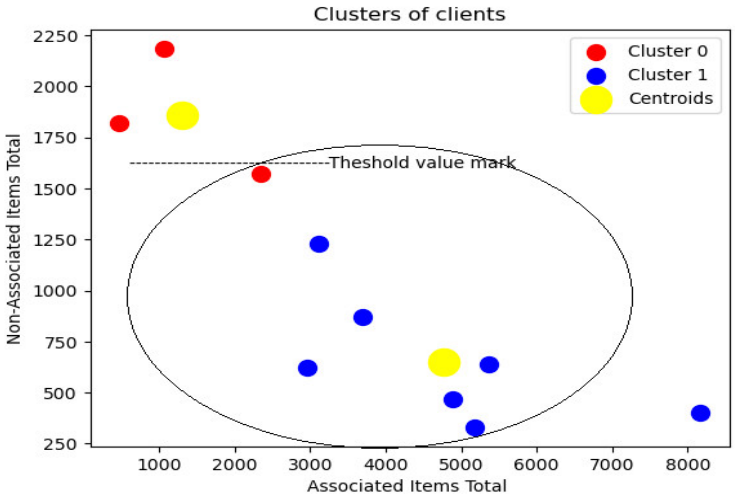

Figure 2. Clusters formed with non-associated items added with their threshold values.

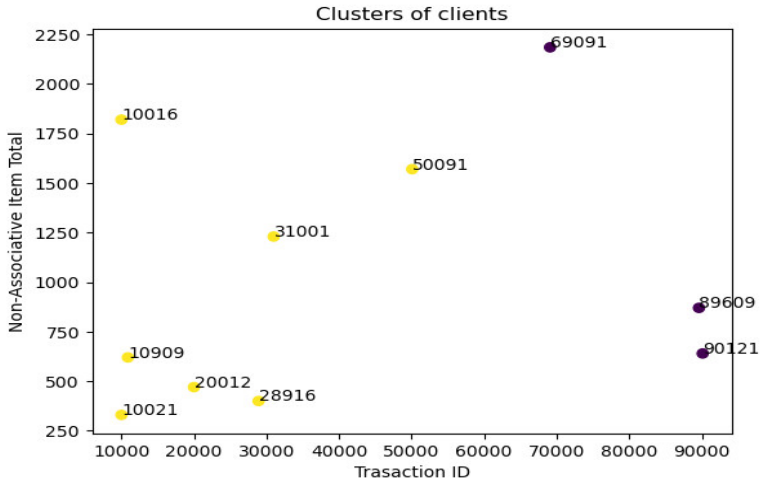

Figure 3. Cluster of points showing the transaction ID with the purchase amounts.

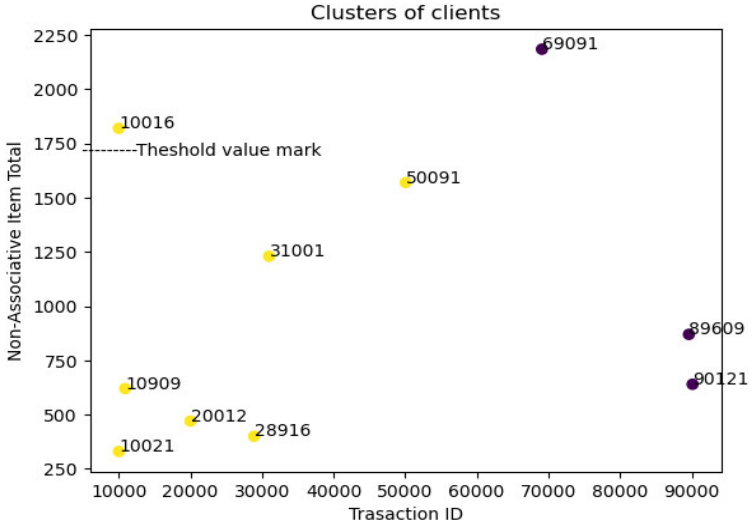

Figure 4. Cluster of points showing the transaction ID with the purchase amounts and the threshold value.

Generally, only the associated items were considered and the non-associated item sets were ignored, even though more processing costs were incurred to obtain the final result. Considering the non-associated items sets not only allows for the inclusion of cost-effective non-associated items, but also maximizes the processing power that might have been wasted when only finding the associated items.

Figures 2 and 3 display two clusters of associated items and non-associated item clusters, which completely ignore non-associated clusters. If we consider both clusters and utilize the threshold value, which is the base value (such as the median of the highest and lowest non-associated sets), we can consider non-associated items. This approach enables the inclusion of some transactions or customers for potential benefits. Additionally, some items with high costs purchased by customers in the non-associated cluster are valuable and could be considered as per the business rules.

This shows that one or more customers are added from the non-associated item sets marked by the threshold value and that some of the customer IDs can be considered along with the associated items sets.

The CNAIS algorithm was executed using an Intel Pentium 5 machine with 16 GB of RAM on Windows 10 OS. The transactions were stored for each transaction ID for a period of time and the results and graphs are provided. Since CNAIS returns the associated items, the non-associated item sets, and the total price of all the transactions, this algorithm may use more time in milliseconds when compared to the ordinary Apriori algorithm. Table 3 shows the execution time.

Table 3. Execution time in milliseconds.

Algorithm	ItemSet Mining		CLUSTER	Total-Time (ms)
CNAIS + CLUSTER	432.78	AIS, NAIS are generated	401.2	833.98
APRIORI + CLUSTER	410.13	Only AIS	409.2	819.33

AIS-Associated Item Sets
NAIS–Non-Associated Item Sets

The Figure 5 compares the execution times of the different techniques.

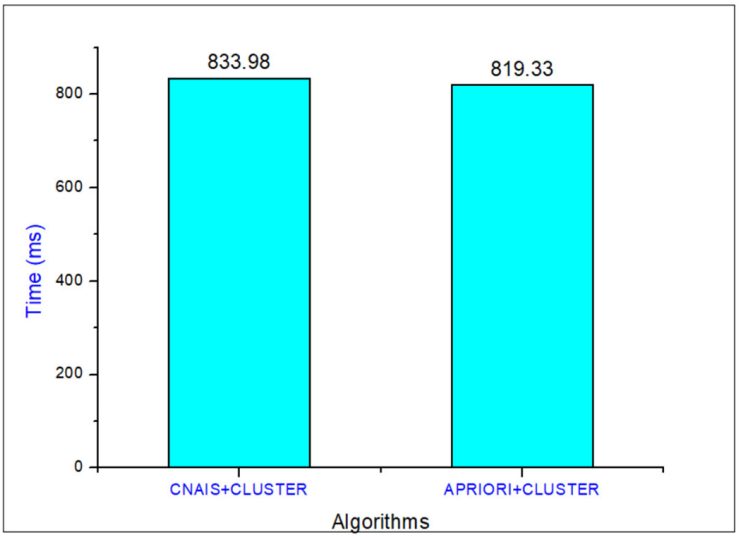

Figure 5. Graph showing time comparisons.

5. Conclusions

Association rule mining and clustering techniques were studied with examples and algorithms. The proposed CNAIS algorithm is given along with sample datasets. The

results and analyses are displayed with figures and tables and we have shown the aim of the algorithm, considering one or more non-associated item sets or transactions in the graphs. The proposed study focuses on the design and development of the Clustering of Non-Associated Items Set (CNAIS) technique within a transactional database. The development of the algorithm and its applications within datasets are given and the results are noted. Comparisons with state-of-the-art methods show that CNAIS exhibits better performance.

Author Contributions: Conceptualization, V.B.M. and M.S.; methodology, V.B.M. and M.S.; software, V.B.M. and M.S.; validation, V.B.M. and M.S.; formal analysis, V.B.M. and M.S.; investigation, V.B.M. and M.S.; resources, V.B.M. and M.S.; data curation, V.B.M. and M.S.; writing—original draft preparation, V.B.M. and M.S.; writing—review and editing, V.B.M. and M.S.; visualization, V.B.M. and M.S.; supervision, V.B.M. and M.S.; project administration, V.B.M. and M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Madan Kumar, K.M.V.; Srinivas, P.V.S. Algorithms for Mining Sequential Patterns. *Int. J. Inf. Sci. Appl.* **2011**, *3*, 59–69.
2. Kumar, B.N.; Mahesh, T.R.; Geetha, G.; Guluwadi, S. Redefining Retinal Lesion Segmentation: A Quantum Leap with DL-UNet Enhanced Auto Encoder-Decoder for Fundus Image Analysis. *IEEE Access* **2023**, *11*, 70853–70864. [CrossRef]
3. Peltier, J.W.; Schibmwsky, J.A.; Schuhz, D.E. Interactive Psychographics: Cross-Selling in the Banking Industry. *J. Advert. Res.* **2002**, *4*, 7–22. [CrossRef]
4. Saravanan, C.; Mahesh, T.R.; Vivek, V.; Shashikala, H.K.; Baig, T. Prediction of Task Execution Time in Cloud Computing. In Proceedings of the 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 11–13 November 2021; pp. 752–756.
5. Kotsiantis, S.; Kanellopoulos, D. Association Rules Mining: A Recent Overview. *GESTS Int. Trans. Comput. Sci. Eng.* **2006**, *32*, 71–82.
6. Pasumarty, R.; Praveen, R.; Mahesh, T.R. The Future of AI-enabled servers in the cloud—A Survey. In Proceedings of the 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 11–13 November 2021; pp. 578–583.
7. Bellini, P.; Palesi, L.A.I.; Nesi, P.; Pantaleo, G. Multi Clustering Recommendation System for Fashion Retail. *Multimed. Tools Appl.* **2022**, *28*, 1573–7721. [CrossRef] [PubMed]
8. Sindhu Madhuri, G.; Chokkanathan, K.; Mahesh, T.R.; Musthafa, M.M.; Vanitha, K.; Vivek, V. MLPDR: High Performance ML Algorithms for the Prediction of Diabetes Retinopathy. In Proceedings of the 2023 International Conference on Network, Multimedia and Information Technology (NMITCON), Bengaluru, India, 1–2 September 2023; pp. 1–7.
9. Sindhu Madhuri, G.; Somashekhara Reddy, D.; Mahesh, T.R.; Rajan, T.; Vanitha, K.; Shashikala, H.K. Intelligent Systems for Medical Diagnostics with the Detection of Diabetic Retinopathy at Reduced Entropy. In Proceedings of the 2023 International Conference on Network, Multimedia and Information Technology (NMITCON), Bengaluru, India, 1–2 September 2023; pp. 1–8.
10. Han, J.; Kamber, M. *Data Mining: Concepts and Techniques*, 2nd ed.; Morgan Kaufmann Publishers: Burlington, MA, USA, 2006.
11. Mahesh, T.R.; Vivek, V. Image Classifications Methods Analysis with Different Methods to for Identifying best Image Layout with High Resolution. In Proceedings of the 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), Greater Noida, India, 27–29 January 2023; pp. 451–455.
12. Agrawal, R.; Srikant, R. Mining Sequential Patterns. In Proceedings of the 11th International Conference on Data Engineering, Taipei, Taiwan, 6–10 March 1995.
13. Ahalya, G.; Pandey, H.M. Data clustering approaches survey and analysis. In Proceedings of the 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), Greater Noida, India, 25–27 February 2015; pp. 532–537.
14. Xu, R.; Wunsch, D. Survey of clustering algorithms. *IEEE Trans. Neural Netw.* **2005**, *16*, 645–678. [CrossRef] [PubMed]
15. Ramakrishna, M.T.; Venkatesan, V.K.; Bhardwaj, R.; Bhatia, S.; Rahmani, M.K.I.; Lashari, S.A.; Alabdali, A.M. HCoF: Hybrid Collaborative Filtering Using Social and Semantic Suggestions for Friend Recommendation. *Electronics* **2023**, *12*, 1365. [CrossRef]
16. Gunasekaran, K.; Kumar, V.V.; Kaladevi, A.C.; Mahesh, T.R.; Bhat, C.R.; Venkatesan, K. Smart Decision-Making and Communication Strategy in Industrial Internet of Things. *IEEE Access* **2023**, *11*, 28222–28235. [CrossRef]

17. Kanungo, T.; Mount, D.M.; Netanyahu, N.S.; Piatko, C.D.; Silverman, R.; Wu, A.Y. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 881–892. [CrossRef]
18. Devarajan, D.; Alex, D.S.; Mahesh, T.R.; Kumar, V.V.; Aluvalu, R.; Maheswari, V.U.; Shitharth, S. Cervical Cancer Diagnosis Using Intelligent Living Behavior of Artificial Jellyfish Optimized with Artificial Neural Network. *IEEE Access* **2022**, *10*, 126957–126968. [CrossRef]
19. Karthick Raghunath, K.M.; Vinoth Kumar, V.; Venkatesan, M.; Singh, K.K.; Mahesh, T.R.; Singh, A. XGBoost Regression Classifier (XRC) Model for Cyber Attack Detection and Classification Using Inception V4. *J. Web Eng.* **2022**, *21*, 1295–1322.
20. Mahesh, T.R.; Sinha, D.K. Twitter Location Prediction using Machine Learning Algorithms. In Proceedings of the 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC), Bengaluru, India, 18–19 November 2022; pp. 1066–1070.