ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2022.0389

DESIGN OF CATEGORICAL DATA CLUSTERING USING MACHINE LEARNING ENSEMBLE

N. Yuvaraj and A. Jayanthiladevi
Institute of Computer Science and Information Science, Srinivas University, India

Abstract

Cluster analysis of data is a crucial tool for discovering and making sense of a dataset underlying structure. It has been put to use in many contexts and many different fields with great success. In addition, new innovations in the last decade have piqued the interest of clinical researchers, scientists, and biologists. As the number of dimensions in a data set grows, the consensus function of traditional ensemble clustering often fails to generate final clusters. The main problem with conventional ensemble clustering is exactly this. The proposed work employs a similarity measure between links to identify which clusters contain the unknown datasets. To this end, this study proposes employing an improved ensemble framework for clustering categorical datasets. More specifically, it employs ensemble machine learning methods to categorize data. Multiple machine learning algorithms are incorporated into this model. Objective performance indicators are used to compare a model to more traditional approaches to determine how effective each the proposed method is.

Keywords:

Base Clustering, Ensemble Clustering Clusters, Accuracy, Precision

1. INTRODUCTION

Understanding data sets requires the use of data cluster analysis, which is a crucial technique. It plays a crucial role in the procedures of data mining, machine learning, pattern recognition, and retrieval. For the purposes of cluster analysis, data sets are grouped together based on how much they have in common with one another. Clustering can be applied to many different types of data and has many different uses in many different areas, including image processing, pattern recognition, market analysis, and many more. The categorical data collected for this study is clustered in a novel way using an algorithm developed specifically for this study [1].

Reliable clustering outcomes can be obtained because the current clustering method can properly classify useless items like those with missing or null datasets. The algorithm uses an in-dataset clustering algorithm that is grounded in the algorithm internal criteria, which may include things like similarities or dissemination measures [2]. Cluster solutions can vary greatly between implementations of the same clustering algorithm on the same dataset. Thus, it is difficult to provide a precise evaluation of the impact that clustering has on this crucial matter. In order to assess the quality of the clustering results, we employ the cluster validity indices [3].

However, this major barrier can be overcome by combining the benefits of different clustering methods into a more intricate framework. This will unlock the full synergistic potential of the approaches. Compiling the solutions from the various base clusters into a concluding partition is the final step in completing the group profile [4]. The first step in putting this meta-level strategy into action is to build a cluster ensemble. The second step is to build the consensus function, or final partition. The clustering set presents a number of challenges, one of the most challenging being finding the optimal consensus function that will maximize the results of a single clustering algorithm [5].

Connecting link-based ensemble with data clustering over categorical datasets, this study fills a void in the literature. This research addresses a gap in the literature. This study uses a method called link-based ensemble to get rid of datasets that have missing or unknown data points [6].

In this paper, we propose using a link-based technique to clean up these incomplete datasets by removing any potentially misleading or incorrect data points. The proposed work employs a similarity measure between links to identify which clusters contain the unknown datasets. Another helpful function it serves is connecting link analysis and data clustering. This is achieved by enhancing the clustering performance over categorical data at three levels: the base clustering, the ensemble clustering, and the final data partitioning. Base clusters are generated by the clustering algorithm and then used to build direct or indirect cluster ensembles. To further develop and refine the cluster-association matrix, the ensemble clustering approach is used.

2. PROBLEM FORMULATION

This study provides background on the clustering issue inherent in categorical data. At the beginning of the chapter, there is a discussion of the many different approaches to the overarching problem of cluster analysis. Second, research into these methodologies concentrates specifically on the unique circumstance of categorical data, which differs from numerical data in that it cannot be quantified. In addition to that, new methods and standards that are specifically geared toward this field are presented [7].

The problem of partitioning a set of objects into groups in which objects within a group share similarity while objects across groups share differences is referred to informally as data clustering. Data clustering is a term that has been given to this problem. The term category data clustering describes the method of grouping information entities in accordance with their various classes of information [8]. An attribute values fall into a categorical domain if they can't both be present and absent at the same time. To restate, the data does not include any sort of order or a distinct distance function, and there is no mapping from the categorical to the numerical values. Furthermore, there is no semantically sensitive mapping.

The clustering problem has received extensive study for many years and across many disciplines because of its practical importance in the real world. The importance of precise data measurement and analysis increases as more data is amassed. The method that will be used in this case relies heavily on the concept of clustering. Researchers have been able to process larger

datasets and generate more reliable findings thanks to the rise in the number of people working on research projects in recent years.

Objects with similar characteristics are stored together in the data clusters, while those with different characteristics are kept in their own sets. This definition presumes that it is possible to determine with reasonable accuracy whether or not a set of data points can be categorized as belonging to a single cluster. The vast majority of clustering algorithms utilize numerical attributes to categorize data points. Because it is easiest to do so with numerical attributes. This allows for the use of well-established metrics based on geometric analogies to define the degree of similarity between two objects (or dissimilar). Data value underlying semantics form the basis for all of the definitions. As we have access to data on distances, we can define a scoring system for the set (e.g., the medium square distance between each point and its representative). The difficulty of clustering arises when attempting to maximize the precision of quality measurement through the grouping of points [9].

Clustering data into meaningful categories is more challenging than doing the same with numbers. Thanks to the critical distance functions, we can separate the relevant geometric properties from the data facts. By their very nature, categorical facts prevent us from giving a precise and clear definition of any kind of distance or distinction. Categorical information appears to be more difficult to classify than numerical data, as it has more granular characteristics to divide into subcategories [10]. There are many different kinds of data available now, and it has been predicted that there will not be sufficient procedures for clustering them in the near future. As long as all available algorithms measure similarity in the same way, according to the same standards, that will be enough. That is to say, the number of irregular value attributes calculated in parallel is the same as the number of conventional value attributes.

To get around these limitations, the cluster ensemble technology employs categorical data sets; this is generally accepted as an efficient solution for clustering algorithms. Both the solidity and the clustering have undergone dramatic enhancements. It has a lot going for it, but it not perfect. The biggest problem is that it often results in inaccurate data clusters, even though it does a lot right. The quality of the data partition decreases as the degree to which data points are linked in the information or binary clustering ensemble matrix increases. This is because it leads to a decrease in data partition quality as many zero entries become unidentified. To deal with the deteriorating quality of data partitions, a linked strategy based on a refined cluster association matrix has been put into place. Within the cluster, a connection is made that helps determine which data points were previously obscured and consequently improves the quality of the data partition. The cluster determines this connection by measuring the degree to which its constituent parts are alike. The purpose of this technique is to enhance the quality, precision, and reliability of the cluster by combining the multiple partitions that exist across different clusters into a single clustering solution.

However, there are a plethora of data sets where the data objects are defined by numerical attributes that are inherently unique from one another. Such data sets and collective values are the focus of a current lexical investigation.

3. PROPOSED ENSEMBLE CLUSTERING

In this chapter, we introduce a simple algorithm called k-means that can be used to quickly and easily establish initial cluster locations. Most algorithms are computationally complex because they aim for more precise results. The primary motivation for doing so is to guarantee that the proposed framework has adequate storage for data, allowing for faster computations. Large categorical data sets are difficult to analyze with traditional methods. However, conventional clustering techniques suffer from a number of drawbacks and cannot effectively cluster all data sets. Adding a matrix to the mix causes these clustering ensemble methods to perform poorly in comparison to more traditional approaches. To complicate the selection of the centroid in k-means clustering, duplicate data in the clusters can arise.

Depending on the initial clustering of a dataset, different attributes within that dataset may take on more or less importance. As a result, this method is recommended for use during the clustering process to help deal with the unreliability of categorical data by providing an early warning about the importance of each attribute. You can see a diagram of the system architecture of such a process, labeled as Fig.1, here.

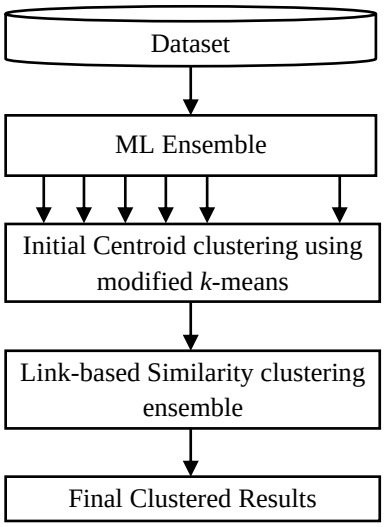

Fig.1. Proposed Model

3.1 CLUSTER ENSEMBLE

A database is created by running the firefly algorithm over a set of data that has been organized into clustered categories. Using consensus clusters in unsupervised learning is similar to employing ensemble methods in that it allows you to avoid the standard entity problem. This relates to the requirements of obtaining a large number of unique input clusters from the targeted data sets and locating a consensus cluster from the baselines. Consensus clusters, created from the same dataset through multiple iterations of an algorithm, can help bring into harmony seemingly conflicting clustering results. There are three methods for achieving this goal:

- **Direct Ensemble**: The term Direct Ensemble, abbreviated as DE, describes the procedure of transforming a categorical data point into a collection without resorting to basic clusters. Prior to the creation of this cluster ensemble, the problem of clustering categorical data had not been applied

to cluster ensembles. The term can be compared to a cluster, which consists of a group of items that share many characteristics. This formalism allows us to directly transform categorical data into a cluster ensemble. No axiomatic types are necessary for this change to take place. This method yields high-quality clustering information because it allows for more variety within an existing cluster ensemble. This is because the accuracy of clustering all data attributes is higher than the accuracy of clustering any one partition attribute.

- **Full-Space Ensemble**: The Full-Space Ensemble is constructed on top of a network of nodes (a basic clustering algorithm is obtained from an original category dataset). The firefly algorithm is used to generate a rough clustering after an initial determination of the cluster center. The firefly algorithm generates artificial instability due to both the fixed modes and the random modes, which choose the cluster from the initial clustering. The cluster is chosen by both of these processes.
- **Subspace Ensemble:** The Subspace Ensemble is an extensive cluster ensemble that accounts for many different types of information. Using the results from the Firefly algorithm processing of the data subspace, a cluster ensemble can be generated. The firefly algorithm is used by the cluster ensemble to make clustering decisions based on the properties of a given subspace, with both deterministic and stochastic approaches being taken into account. If the dataset hasn't been preprocessed, the search should center on the relationships between concepts. The proposed method is meant to perform an automated parsing of the dataset in order to find useful data.

Early in the algorithm iteration process, each node sees a subset of the entire dataset. The agents use a simple clustering algorithm in an effort to discover the local partition of their data on their own. The first of two figures illustrating this preparatory step is provided below. Particularly, various algorithms such as k-means, K-nearest neighbor, SVM, NB, and ANN can be used to obtain different local results by adjusting the parameters of these algorithms in various ways, such as by changing the number of clusters, the initial random centers, the dissimilarity metrics that are used, and so on.

3.2 MACHINE LEARNING ENSEMBLE

This novel approach is advantageous because it eliminates the necessity for all nodes to begin with the same local dataset before arriving at a final consensus on the clusters and their centroids. This opens up the possibility of utilizing numerous datasets in research and development. Based on the assumption that all agents start with the same number of clusters but different centers, we will proceed with the K-means algorithm as our standard clustering procedure. Below, we'll go into greater depth about K-means clustering. This is done to avoid a situation where the cluster composition varies from instance to instance and every local model has an unknown number of clusters.

The Fig.3 shows the outcomes of the preliminary cluster analysis performed on the toy selection. The Fig.3 shows that the errors can be quite different at each node. These findings are sensitive to how the axes are set up at the outset. Some clusters, for example, are divided into smaller groups while others are left intact.

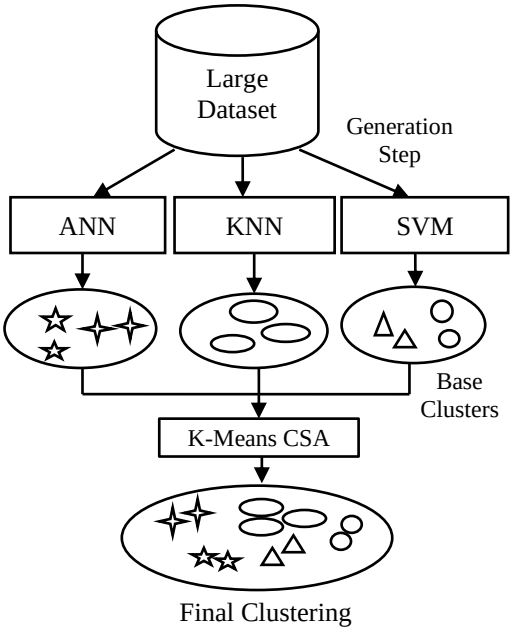

Fig.1. ML Ensemble Clustering

4. K-MEANS CENTROID SELECTION ALGORITHM FOR FINAL ENSEMBLE

To pinpoint the optimal hub, researchers employ the k-mean-variance technique, which takes into account the overall average distance between points. The Euclidean distance between pairs of points that have been user-averaged from the mean is used to determine the ordering of the clusters. The proposed method is designed to handle data that combines numerical and categorical components. To begin, consider the following equation for pinpointing the cluster epicenter:

$$C{:}C = \{1, 2,\ldots,k\} \text{ is } 1 + (C - 1) \times M/k. \quad (1)$$

where, K - sample points that are selected from the M initial cluster data points to be represented.

This algorithm primary goal is to pick every single cluster point in the center selection procedure so that no datasets are left in storage.

The first part of the algorithm is meant to aid in finding the cluster center, and the second part is meant to aid in fusing all of the data points into a single cluster. The first technique averages all the data to find out which node in the cluster (k) is the hub. The second method determines which node within the cluster is the most central by taking the distance between nodes within the cluster and using it to make the determination. This procedure will be repeated until the k cluster centers have been determined.

The second phase involves organizing the data according to where each cluster is located. This can be accomplished either numerically (for a fixed ensemble) or categorically (for a random ensemble). This is why we make a clear distinction between quantitative and qualitative features. Each data point is connected to the head of the cluster to which it belongs using this distance estimate, and this process is repeated until all data points in the

database have been connected to the heads of the clusters to which they belong.

Type III aims to offer an alternate method to diversity generation so that data subspaces within an ensemble can be utilized. The type III set $n \times d$ can be used to create clusters with a predetermined or arbitrary number of members. The formula $q = q_{min} + [\delta (q_{max}-q_{min})]$ must be used to generate the data subspace for a given set of data ($n \times d$). By applying the following formula, we can find the value of q for a given ($n \times q$), where n is the total number of data points, d is the total number of attributes, q_{max} and q_{min} are the upper and lower limits of the subspace, and q is a uniform random variable. For any given n_q, this formula yields the value of q. (0,1).

Q is assumed to have a range from $0.85d$ to $0.75d$, with $0.85d$ being the maximum q_{max} and $0.75d$ the q_{min}. From the first d attributes to the last n clusters, everything is picked at random $\delta \in$ (0,1). Using the formula $h = \lfloor 1 + \delta d \rfloor$, we can compute the index value for each data point. The attribute chosen (h) in this formula is one of several possible choices. Clustering ensembles are built using the k-modes, but the subspace attribute set can be generated using either a random-k or a fixed-k approach.

4.1 LINK BASED SIMILARITY

The proposed algorithm helps to organize the undiscovered connections into clusters C_d. Therefore, we build a graph $G = (V;W)$ with a set of weighted edges, W, and a set of vertices, V, to represent the intercluster and intracluster connections $C_e \in V$. The symbol for this graph is $G = (V;W)$. The edge parameter $w_{de} \in W$, defined as | $w_{ide} = |L_d \cap L_e|/|L_d \cup L_e|$, represents the fraction of common members between the C_d and C_{eV} clusters and serves as the edge weight that connects the two. The $w_{de} \in (0,1)$ function would be used here; it would return w_{de} if the clusters were identical and wde0 otherwise. We would take the midpoint if the value was between 0 and 1. Weights are much closer to 1 than they would be otherwise if there are many similarities between clusters and vice versa. When removing duplicates from a cluster, hash values (or Division Reminder Hashing) are often used. This is done to prevent the L_d and L_e from appearing twice in the data.

As shown in Eq.(2) and Eq.(3), Jaccard normalization is used to transform the values before similarity calculations are performed.

$$R_0(a,b) = \begin{cases} 0 \text{ if } a \neq b \\ 1 \text{ if } a = b \end{cases} \quad (2)$$

$$R_{K+1}(a,b) = C \frac{|L(a) \cap L(b)|}{|L(a) \cup L(b)|} + \frac{1}{|L(a) \cup L(b)||L(b)|} \sum_{a' \in \frac{L(a)}{L(b)}} \sum_{b' \in L(b)} R_K(a',b') + \frac{1}{|L(a) \cup L(b)||L(p)|} \sum_{b' \in \frac{L(a)}{L(b)}} \sum_{a' \in L(b)} R_K(a',b') \quad (3)$$

Finally, we use the recursive C-Rank formula demonstrated in Eq.(4) and Eq.(5) to estimate the degree of similarity between two independent variables a and b.

$$S(a,b) = \frac{C}{|L(a)||L(b)|} \sum_{i=1}^{|L(a)|} \sum_{j=1}^{|L(b)|} S(L_i(a), L_j(b)) \quad (4)$$

$$R_{K+1}(a,b) = \frac{C}{|L(a)||L(b)|} \sum_{i=1}^{|L(a)|} \sum_{j=1}^{|L(b)|} R_K(L_i(a), L_j(b)) \quad (5)$$

where,

$L(a)$ - undirected link neighbours set in a dataset $a \in L_d$

$L(b)$ - undirected link neighbours set of dataset $b \in L_e$

$R_{K+1}(a,b)$ - a similarity score between the a and b at K+1.

$R_{K+1}(a,b)$ - a similarity score between the a and b at K.

C - decay factor $\in$ [0,1].

5. RESULTS AND DISCUSSION

In this section, we present the outcomes of our performance analysis of the HFALCE and MKLCE ensemble methods. Both existing methods and those that are proposed are put to the test and evaluated using a benchmark dataset. The proposed method is tested with a wide range of ensemble configurations. The effectiveness of the proposed system is evaluated by comparing the quality of the clustering with that of existing methods, and by using a number of different performance metrics.

5.1 DATASET SETTINGS

In Table.1 you can see the 20 newsgroups, the Zoo dataset, the breast cancer dataset, the primary tumor dataset, and the lymphographic dataset that were used in the evaluation. The parameters of the data sets include data points (N), attributes (d), attribute values (AV), and classes (K).

For illustration, the 20-newsgroup data set contains 1,002 documents drawn from 2 newsgroups and 6,084 unique terms. In particular, the nominal value is derived from the overall frequency ($f \in \{0,1\ldots,\infty\}$) with which the keyword appears in the documents. This value is Yes if the keyword occurs more than zero times per document f>0, and No otherwise. The evaluation makes use of categorical data and compares its results to those obtained by using established clustering and ensemble techniques.

Table.1. Description of Datasets

Dataset	Data Points (N)	Attributes (D)	Attribute Values (AV)	Classes (K)
Zoo	101	16	36	7
Lymphography	148	18	59	4
Primary Tumor	339	17	42	22
Breast Cancer	683	9	89	2
20 Newsgroup	1000	6084	12168	2

5.2 SIMULATION SETTINGS

The source code for the proposed system is made available in a Java development environment, and the simulation runs on a personal computer running Windows 10 with a CPU clocked at 3.00 GHz by Intel(R) i7-6950X.

The subsequent clustering ensemble is used for two purposes: first, to assess the ensemble overall quality; and second, to empirically compare the various ensembles. In this case, we'll be zeroing in on these five individual components of an ensemble:

- Type-I Ensemble
- Type-II Ensemble (Fixed-k)
- Type-II Ensemble (Random-k)
- Type-III Ensemble (Fixed-k)
- Type-III Ensemble (Random- k).

5.3 PERFORMANCE METRICS

Cohesion, variance, precision, recall, clustering accuracy (CA), normalized mutual information (NMI), and the rand index were among the performance metrics used to assess the proposed approach. All of these measurements were compared and contrasted with one another (RI).

5.4 RESULTS AND EVALUATION

We compare the results of proposed method to those of other existing methods, and draw some conclusions about the relative merits of each. Finally, we compare proposed method, as well as some other existing methods, to assess the efficacy of the proposed link-based clustering ensemble methods.

To conclude this study, we evaluate the proposed model against the state-of-the-art methods in terms of the aforementioned metrics as well as cohesion clustering variance, precision, and recall rate. The Table.2-Table.6 show that the proposed model outperforms other existing methods in terms of accuracy, cohesion clustering, variance, precision, and recall rate.

Table.2. Ac**curacy**

Ensemble Algorithm	Accuracy (%)
Proposed HFALCE	0.9184
Proposed MKLCE	0.8871
CO+SL	0.8062
CO+AL	0.6273
CSPA	0.7310
HGPA	0.7176

Table.3. Cohesion

Ensemble Algorithm	Cohesion
Proposed HFALCE	0.8796
Proposed MKLCE	0.8529
CO+SL	0.8286
CO+AL	0.8103
CSPA	0.7932
HGPA	0.7803

Table.4. Variance

Ensemble Algorithm	Variance
Proposed HFALCE	0.7913
Proposed MKLCE	0.7639
CO+SL	0.7517
CO+AL	0.7410
CSPA	0.7292
HGPA	0.7047

Table.5. Precision

Ensemble Algorithm	Precision
Proposed HFALCE	0.8171
Proposed MKLCE	0.7985
CO+SL	0.7802
CO+AL	0.7733
CSPA	0.7650
HGPA	0.7543

Table.6. Recall

Clustering Algorithm	Recall
Proposed HFALCE	0.8504
Proposed MKLCE	0.8356
CO+SL	0.8219
CO+AL	0.8102
CSPA	0.7940
HGPA	0.7820

The proposed MLE is one of two methods that can be used to successfully cluster categorical data. The simulation results demonstrate that the proposed MLE outperforms the state-of-the-art techniques. The results indicate that the proposed MLE produces more effective clustering ensembles than the currently used methods because it makes use of machine learning algorithms during the initial clustering stage.

6. CONCLUSION

The clustering process consists of three steps: first, clustering is performed using the proposed MLE method; second, clustering is performed using ensemble clustering; and third, clustered datasets are decomposed using bipartite graph partitioning. The clustering process is broken up into stages, each of which employs a unique clustering method. Beginning with k-means, K-nearest neighbor, SVM, NB, and ANN algorithms for initial base clustering, continuing with link-based ensemble clustering for intermediate clustering, and finally with feature-based partitioning to decompose the clustered datasets, the MLE method clusters datasets on multiple levels. The MLE strategy was put forward as a way to improve the effectiveness of education.

In terms of clustering accuracy, normalized mutual information, adjusted rand index, average score, precision, recall, variance, and cohesion, the proposed MLE method outperforms the state-of-the-art method and other existing methods when applied to categorical datasets. The proposed MLE method superior performance compared to the aforementioned alternatives demonstrates this point. Effective base clustering with the firefly algorithm contributes to the better results in the proposed MLE compared to the base clustering with the modified k-means centroid algorithm. Unfortunately, none of the currently available approaches make use of base clustering to filter out superfluous information. Consequently, MLE may be more effective than other approaches.

REFERENCES

[1] L. Bai and J. Liang, "A Categorical Data Clustering Framework on Graph Representation", *Pattern Recognition*, Vol. 128, pp. 1-13, 2022.

[2] R. Brnawy and N. Shiri, "Improving Quality of Ensemble Technique for Categorical Data Clustering Using Granule Computing", *Proceedings of International Conference on Database and Expert Systems Applications*, pp. 261-272, 2021.

[3] G. Pole and P. Gera, "Cluster-Based Ensemble Using Distributed Clustering Approach for Large Categorical Data", *Proceedings of International Conference on ICT Analysis and Applications*, pp. 671-680, 2021.

[4] I. Khan and R. Hedjam, "Ensemble Clustering using Extended Fuzzy k-Means for Cancer Data Analysis", *Expert Systems with Applications*, Vol. 172, pp. 114622-114633, 2021.

[5] D.T. Dinh, V.N. Huynh and S. Sriboonchitta, "Clustering mixed Numerical and Categorical Data with Missing Values", *Information Sciences*, Vol. 571, pp. 418-442, 2021.

[6] I. Singh, N. Kumar and S. Jain, "A Multi-Level Classification and Modified PSO Clustering based Ensemble Approach for Credit Scoring", *Applied Soft Computing*, Vol. 111, pp. 107687-107698, 2021.

[7] B.A. Hassan and T.A. Rashid, "A Multidisciplinary Ensemble Algorithm for Clustering Heterogeneous Datasets", *Neural Computing and Applications*, Vol. 33, No. 17, pp. 10987-11010, 2021.

[8] K. Parish Venkata Kumar and M. Jogendra Kumar, "Concept Summarization of Uncertain Categorical Data Streams Based on Cluster Ensemble Approach", *Proceedings of International Conference on Pervasive Computing and Social Networking*, pp. 385-398, 2022.

[9] V. Shorewala, "Early Detection of Coronary Heart Disease using Ensemble Techniques", *Informatics in Medicine Unlocked*, Vol. 26, pp. 1-16, 2022.

[10] I.B. Ayinla and S.O. Akinola, "An Improved Ensemble Model using Random Forest Branch Clustering Optimisation Approach", *University of Ibadan Journal of Science and Logics in ICT Research*, Vol. 7, No. 2, pp. 8-19, 2021.