Machine Learning Based Improved Heart Disease Detection with Confidence

https://doi.org/10.3991/ijoe.v19i08.37417

Anas Domyati, Qurban Memon(✉)
UAE University, Al Ain, United Arab Emirates
qurban.memon@uaeu.ac.ae

Abstract—One of the hardest jobs in medicine is to predict when someone will have a heart attack. Given how challenging it is to anticipate heart attack, there is an urgent need to automate the prediction process using diagnostic data, and at the very least generate an early warning. This research makes a contribution by making it easier to diagnose cardiac problems using machine learning methods applied on the well-known Cleveland heart disease dataset. Several performance indicators are utilized to evaluate each model's strength. It turns out that support vector machine and random forest produced some incredibly promising outcomes. An improved prediction of heart disease for an embedded platform is, thus, proposed, based on the computational complexity of each model and experimental results, where the advantages of several classifiers are accumulated. The approach suggests that, and only if, more than one of these classifiers detect heart disease, the detection of heart illness is possible with increased confidence. In the end, experimental findings are drawn to a conclusion, with potential future options for advancing this effort.

Keywords—robust heart attack detection, support vector machine, random forest, machine learning

1 Introduction

Coronary arteries become occluded and narrowed, which leads to heart failure. Coronary arteries regulate the blood flow to the heart. The typical symptoms of coronary disease include swollen feet, body weakness, breathing difficulties, and fatigue, among others. In the beginning, conventional investigative techniques were used to identify cardiac sickness, but it was later found that they were difficult. Because of shortage of medical diagnostic tools and healthcare professionals, heart disease detection and its treatment are incredibly challenging in underdeveloped regions. In a study conducted at World Health Organization [1], 17.90 million people passed away in 2016 as a result of cardiovascular disease. This amount is responsible for almost one-third of all deaths worldwide.

The findings of physical examinations, medical history of patients, and a physician's review of any pertinent symptoms are the main components of traditional invasive procedures for detecting heart disease [2]. Among the common processes, angiography is considered as one of the most accurate methods to identify heart issues. On the other

hand, angiography has a number of disadvantages, including relatively higher cost and a wide spectrum of unfavorable effects. In order to safeguard the patient's health, a precise and correct diagnosis of cardiac disease is seen to be absolutely necessary.

20% of patients at high risk for cardiovascular disease are underdiagnosed as a result of risk misclassification [3]. Common diagnostic methods rely on the knowledge and experiences of medical specialists, which increases the risk of errors, delays appropriate treatment, lengthens treatment times, and dramatically increases costs. Therefore, it is considered essential and crucial to get a precise diagnosis of cardiac disease in order to save the patient from suffering more harm.

Though machine learning techniques may be able to predict and categorize people with heart disease using a variety of variables using various machine learning models. However, hospitals all over the world have been said to be in need of an intelligent system that creates databases of such patients and can assist the doctor in forecasting the severity of the condition. The objective of the present work is to improve confidence of the examining doctor by evaluating the disease using an intelligent system that provides assessment based on more than one independent tool running on an embedded system. A thorough review of the literature is presented in the next subsection to highlight recent interest in this field.

2 Literature review

For a variety of applications, researchers have tried to create non-invasive healthcare related procedures based on machine learning approaches, including problems with traditional invasive-based techniques for heart problems detection. Researchers commonly use the Cleveland heart disease dataset. In this situation, the work in [4] for diagnosis of heart illness used a logistic regression classification technique and had a 77.1% accuracy rate. A multi-layer perceptron (MLP) classifier was also employed by Wankhede et al. [5] to reach an accuracy of 80%. Similar to this, Allahverdi et al. [6] combined neural networks to create a classification system for heart disease with 82.4% accuracy. Awang et al. [7] experimented with Naïve Bayes (NB) and Decision Tree (DT) for a follow-up study and discovered accuracy of 82.7% and 80.4% respectively for heart disease detection.

The authors Das et al. [8] have successfully created a non-invasive algorithm utilizing three classifiers to forecast the detection of stenosis in three arteries. In a distinct piece of study [9], the authors create a machine learning model that integrates five (5) various approaches, and they assert enhanced performance in terms of heart failure prediction. On the UCI dataset for predicting cardiac disease, the authors Elnawawy et al. [10] investigated NB and support vector machines for classification and discovered that the support vector machine surpassed Naive Bayes in terms of classification accuracy. The employment of diverse machine learning algorithms has inspired many to investigate its potent advantages in numerous areas including healthcare [11–12].

Several categorization algorithms have been examined in a survey study [13] to predict the risk level of patients based on different parameters. The researchers discovered that the amount of criteria used increased risk accuracy. With encouraging results, a hybrid recommender system [14] based on combination of deep learning with multiple kernel learning has also been put out to identify cardiac disease.

A deep learning network has also been studied by Tomov et al. [15], who used k-way cross validation and the Matthews correlation coefficient to construct, evaluate, and optimize several designs for detecting cardiac disease using publicly accessible databases. In a different work [16], the authors proposed a system that combines the MapReduce algorithm with data mining techniques. In this, a vast collection of instances are entered into this system. Based on calculated accuracy of over 90%, the authors claim that the MapReduce technique might be utilized to estimate heart disease risks in the clinic.

A CardioXNet technique is suggested [17] for low resource settings that uses end-to-end architecture and light computing to enable automatic detection of five (5) kinds of cardiac auscultation. Based on their findings, the authors assert that the suggested approach is appropriate for employing mobile phones for CVD screening. In a different study, the authors Ali et al. [18] pay close attention to every facet of performance, where the removal of features that do not contribute sufficiently and the use of layered genetic algorithms are recommended with an accuracy of over 90%. For recent study on this subject, reader is referred to works in references [19–20].

The remainder of this study is divided into the following sections. The proposed technique that includes dataset selection, methodology, preprocessing of images, and selection of features, is detailed in section three. The results of the categorization are presented in section four, which also includes parameter evaluation. A possible architecture of the embedded system is discussed in section five, and conclusions are discussed in section six. It should be noted here that the preliminary results of this research are reported in [21].

3 Proposed approach

Developing a dataset that accurately represents the pattern of the targeted class is the first stage in creating a computational model. A problem-related, and properly organized dataset has a significant impact on how effectively a computational model performs. Two well-known heart disease datasets—the Hungarian dataset and the Cleveland dataset—are typically utilized when discussing the significance of the dataset. These datasets are both online and can be found at machine learning repository of the University of California, Irvine and Kaggle repository, respectively. The Cleveland dataset comprises 303 examples, each with unique target labels and 13 features that are chosen for training. The dataset is divided into two categories: cardiac disease is present or not.

The proposed approach's methodology is depicted in Figure 1 and can be divided into parts: Pre-processing, feature selection, training and validation strategy, classification algorithms, and performance evaluation. Each part is discussed in further detail below.

3.1 Data preprocessing

Pre-processing of data aids in converting unstructured data into data that can be processed. In the dataset used, one attribute has four missing values in 176 of the 303 observations, and another has two missing values. The equivalent majority mark in the total instances is used to replace these missing values. The dataset contains a total of 138 instances without heart disease and 165 cases with heart disease.

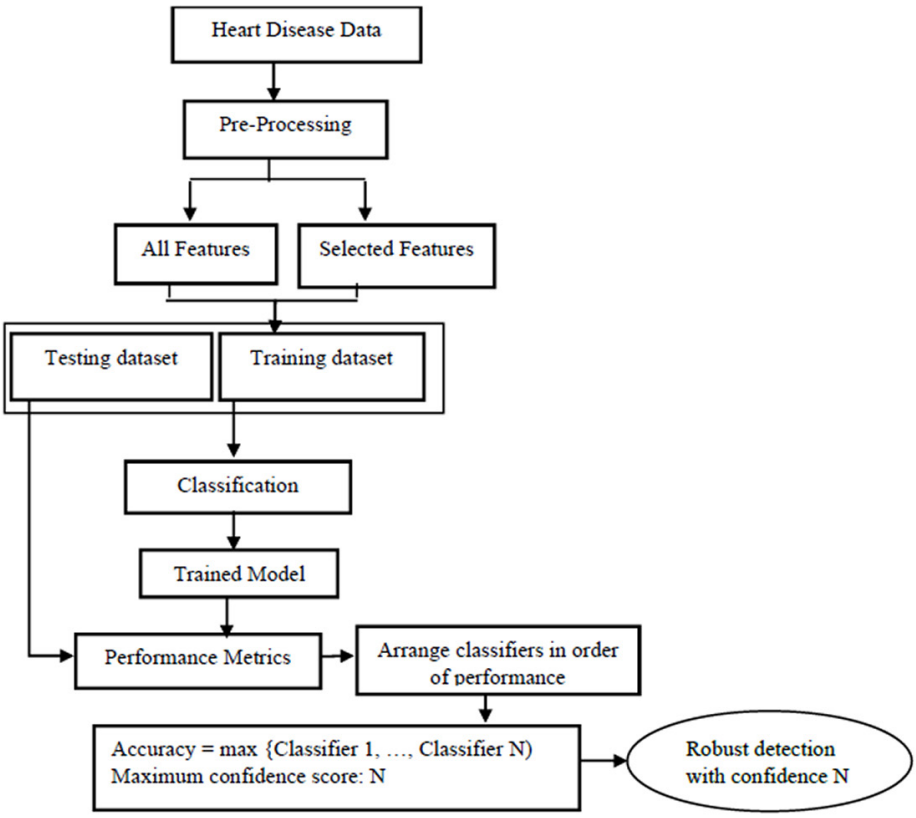

Fig. 1. Methodology for higher confidence in detection

3.2 Feature selection

The method for feature selection must produce an ideal feature sub-space in the dataset, since occasionally, the classification performance may be negatively impacted by irrelevant features. The most significant characteristics of heart disorders are identified by feature selection. Additionally, eliminating irrelevant features enhances and speeds up learning, and aids in the construction of a substantially superior model. Each attribute's weight demonstrates how well it can distinguish across classes.

The Relief feature selection method [22] is a supervised strategy that demonstrates importance to the binary or multiclass objective. Weights are used to standardize the attributes. A higher weight is given to an attribute value if it distinguishes between same class instances and instances of other classes. The weight of a feature is altered during this iterative process. The final subset is chosen from the attributes whose weight exceeds a cut-off. The filter selects the optimal subset in this case to be given to the algorithm for training. The algorithm then chooses examples at random from this optimized subset.

When relevance weights are viewed as the ratio between the number of explained concept changes and the number of explained occurrences, the Relief weights may be easier to understand. If a change can be described in a variety of ways, the estimate will give credit to all possible explanations, and if the explanation involves many features, the estimate will give credit to all of them. To further understand this, Table 1 presents a Boolean problem where Class is represented by the phrase (F1 ∩ F3) ∪ (F1 ∩ F2) and where all three attributes are relevant. It can be claimed that F_1 is in charge of assigning classes in the first instance of Table 1, however F_1 or F_2 would assign classes 0 in the second occurrence, so they share responsibility. F_1 or F_3 share responsibility in the third case, and so on. Given that there are two alternative pairs of feature values in the last instance, it is not sufficient for class to change. Estimates of the scores for all three features are provided by adding up the responsibilities.

Table 1. Dataset description showing responsible features that yield the change

Instances	Values of Features			Class	Responsible Features	Change in Score		
	F_1	F_2	F_3			F_1	F_2	F_3
I_1	1	1	1	1	F_1	1/8	0	0
I_2	1	1	0	1	F_1 or F_2	0.5/8	0.5/8	0
I_3	1	0	1	1	F_1 or F_3	0.5/8	0	0.5/8
I_4	1	0	0	0	F_2 or F_3	0	0.5/8	0.5/8
I_5	0	1	1	0	F_1	1/8	0	0
I_6	0	1	0	0	F_1	1/8	0	0
I_7	0	0	1	0	F_1	1/8	0	0
I_8	0	0	0	0	(F_1 and F_3) or (F_1 and F_2)	1/8	0.5/8	0.5/8
					Total Score	0.75	0.187	0.187

3.3 Training and testing

The Cleveland dataset consists of 303 samples, each of which is selected for training and has a distinct collection of 13 features in addition to the labels. Based on the presence or absence of heart illness, the data set is categorized. The entire dataset is divided into k pieces for K-Fold cross-validation. While the remaining k-1 data sections are used for testing, the first k-1 data portions are used for training. The procedure runs k times. The value of K = 10 is employed in this research for experimental work due to its high repeatability. Thus, 90% of the data portion is utilized for training and the remaining 10% for testing of the model at each iteration of ten-fold cross validation. The outcome is achieved by computing the mean of values in each phase.

3.4 Classification

There exist many classification algorithms, but in this research, the well-known algorithms are investigated. Below, each of them is discussed briefly.

Naive Bayes. The Naive Bayes classifier is another name for the highly scalable Bayesian classification tool, which is based on the Bayes theorem. A subset of the Bayesian network, it is a probability-based classifier. Conditional independence states that an attribute value is independent of the values of the other characteristics in a class. In order to comprehend, assume D represents training data with class labels, '*m*' represents the number of classes C_1, C_2, … C_m, and that '*n*' represents the number of characteristics in each tuple X. This classifier makes the prediction that a tuple X relates to the class with greater posterior probability and is conditioned on X. The Naïve Bayes classifier determines that tuple X relates to class C_i if and only if

$$\mathrm{P}(Ci \mid \mathrm{X}) > \mathrm{P}(Cj \mid \mathrm{X}) \quad for\ 1 \le j \le m, j \ne i \tag{1}$$

As a result, $P(Ci \mid \mathrm{X})$ is maximized.

Random Forest. A categorization scheme based on trees is called Random Forest. As the name implies, the application creates a forest with a sizable number of trees. Using a randomly chosen subset of the training data, it builds a number of trees. The process is then repeated with numerous samples picked at random and a decision is made by the majority. Let's say that the training set is $\mathrm{X} = x_i$ ($i = 1 - N$) and the responses are $\mathrm{Y} = y_i$ ($i = 1 - N$). Following training, the prediction for testing sample x' may be made by majority vote or by averaging the predictions from many regression trees, denoted by $f(x')$ on x'.

$$j = \frac{1}{B}\sum_{b=1}^{B} f(x') \tag{2}$$

Decision tree. Each internal node in a decision tree structure stands for a property being tested, each branch for an outcome, and each leaf for a separate class. The tree grows by selecting the "Best Feature/Attribute" from a set of attributes from the root node, then divides. Calculating two more metrics, "Entropy" and "Information Gain," usually yields the "Best Attribute," which is the one with the most useful information. The term "Entropy" describes how homogeneous a dataset is, but "Information Gain" describes how quickly characteristics' entropies rise or fall [23], and is represented as:

$$I(p(y_1), p(y_2), p(y_3) \ldots\ldots p(y_k)) = -\sum_{i=1}^{k} p(y_i) log_2 p(y_i) \tag{3}$$

where y_i are possible features and $p(y_i)$ are corresponding probabilities.

Support vector machine. Outstanding but adaptable machine learning methods, support vector machines (SVMs) can be used for regression and classification. The goal of SVM is to classify datasets in order to locate the most extreme peripheral hyperplane. The mathematical formula for the one-vs.-one SVM algorithm for two-class classification is [24]:

$$Q = \frac{1}{2}\|w\|^2 - \sum_i \propto_i \left[y_i(\bar{w}.x_i + b) - 1 \right] \tag{4}$$

where $\propto_i$, x_i, y_i, b and w denote constants, input, output class, bias, and final weights, when Q is set to minimum [24], and:

$$\sum_i \propto_i y_i = 0, \; where \; 0 \leq \propto_i \leq C; \tag{5}$$

where C (a constant), needs to be satisfied. The weight vector and the decision for resulting class is computed as [24]:

$$\overline{w} = \sum_i \propto_i \left[y_i (x_i) \right]; \left(\sum_i \propto_i \; y_i \overline{x}_i . \overline{u} + b \right) \geq 0 \tag{6}$$

K-nearest neighbor. The supervised learning technique K-nearest neighbors (KNN) is used for both regression and classification. The k-nearest neighbors (KNN) technique computes the likelihood that a data point will relate to one group or another based on which group the data point is closest to. There are several ways to determine this distance, the most popular being the Euclidian, Manhattan (for continuous), and Hamming distances (for categorical).

3.5 Performance metrics

Several evaluation criteria, including recall/sensitivity, precision, *F*1-score and accuracy—all computed from a matrix—were selected to assess classification performance. This matrix includes measurements-based true negatives (*TN*), false negatives (*FN*), false positives (*FP*), and true positives (*TP*). The following mathematical formulas represent the evaluation metrics:

Accuracy: This is calculated to indicate the overall model performance.

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) * 100 \tag{7}$$

Recall (Sensitivity): This is characterized as the proportion of heart patients who are classified to all heart patients, and it can be computed as:

$$Sensitivity = TP/(TP + FN) * 100 \tag{8}$$

Precision: This is determined by dividing the positive score by the projected positive score. It is stated mathematically as:

$$Precision = TP/(TP + FP) * 100 \tag{9}$$

F1-score: This metric represents a measure of precision and recall. The value is between 1 and 0. The model's performance increases as the value gets nearer to 1. It is determined by the following equation:

$$F1 = (2 * (Precision * Recall))/(Precision + Recall) \tag{10}$$

3.6 Robustness and higher confidence score

The accuracy and increased confidence score can be calculated once several classifier metric measurements are provided for testing data. For robustness, different tests should be conducted. In this work, this is equivalent to using multiple classifiers. Thus, testing becomes more focused and helps determine whether or not a system generates results with confidence. In hospitals, typically, a specialist or a physician views medical test reports before making any judgement. If the number of distinct test reports indicate the same results, the confidence of the doctor improves to make judgement in that direction. Thus, there is a direct relationship. In this research, this is equivalent to applying multiple and distinct classifiers. The more different classifiers identify the disease, the greater the confidence. Once values of all classifier accuracies used are arranged in descending order, the maximum accuracy thus achieved is:

$$Accuracy = max\{Classifier\ 1,\ldots\ Classifier\ 2,\ Classifier\ N\} \tag{11}$$
$$Maximum\ confidence = N$$

where classifiers used in the equation are those which detected in the same direction. Thus, a computing system having a processor with two cores with each core capable of two threads, the system can test four classifiers in parallel and generate a result with a maximum confidence of four and accuracy as per equation (11).

4 Experimental results

In simulations, the pre-processed dataset is utilized in the afore-mentioned classification methods to measure evaluation data. The training and testing was done using Google Collaboratory in python language. Using equations (7–10), the measured values are in shown Table 2 with full feature space and in Table 3 with selected feature space.

Table 2. Classification results on full feature space

	Classification	Precision	Recall	F-1 Score	Accuracy
1	NB	83.71	91.11	87.31	86.71
2	K-NN	88.01	88.01	88.01	86.81
3	DT	84.51	82.31	83.51	81.97
4	SVM	90.01	88.01	89.01	81.97
5	RF	93.71	88.21	90.16	90.16

Table 3. Classification results on selected feature space

	Classification	Precision	Recall	F-1 Score	Accuracy
1	NB	85.71	91.91	88.21	86.71
2	K-NN	86.06	84.41	87.21	87.11
3	DT	85.91	89.16	84.31	84.98
4	SVM	92.31	88.21	89.11	91.73
5	RF	93.71	89.21	91.16	94.50

According to Table 2, random forest fared better than all classifiers in terms of accuracy, while random forest and support vector machine together performed better than other classifiers with respect to precision, recall, and F1-score. While DT and NB exhibited inferior scores compared to the other classification techniques, NB performs the poorest in terms of sensitivity, specificity, and accuracy. As a result, when all classifier models are assessed on the same dataset, RF and SVM (as well as K-NN to some degree) stand out as prominent and robust classifiers. Some of the studies that were previously mentioned have attempted to build an ensemble of classifiers, but simply to compare them. This research has concluded that heart disease can be detected from findings of RF, SVM and other distinct classifier like K-NN, if all are three run in parallel, to produce a confidence with a score of three (3) using equation 11, and maximum accuracy of 90.16% using full feature space. Thus, the integration of classifiers improves the confidence of the medical specialist in true positives and true negatives of the results.

The simulation was repeated based on selected feature space, with results shown in Table 3. The measurements shown in Table 3 indicate that selected feature space improves the classifier performance. The results of Table 3 also suggest that SVM and RF are robust classifiers in selected feature space as well.

How much time or resource a model or an algorithm consume is also important for timely detection and prediction of a developed system. There are two types of computational complexities, namely run time/space complexity and training complexity. The focus is on run time/space complexity as it was assumed that the training platform will have enough resources and that the training is done offline. In other words, the preference is on timely performance over cost incurred during training. For this purpose, the relative computational cost of each algorithm planned for performance evaluation needs to be evaluated. To tabulate the complexity of machine algorithms, let us define the variables that are used to estimate the complexity of a machine algorithm, as follows:

Number of training examples = n, Number of neighbors/support vectors/decision trees = k, Dimensions in the data = d, Classes = c

It is obvious from Table 4 that training time complexities vary but run time complexity of random forest, K-NN and support vector machine are lower than all other machine learning algorithms. It should be noted, here, that processing time depends on the given parameters used in a particular algorithm and the machine used to run the classifier for testing. Thus, the processing time may vary depending on the actual machine used.

A comparison may also be made between contemporary machine learning techniques and the proposed approach. The findings indicate that our suggested model has a higher success rate and confidence than other models in literature. The results thus accumulated are displayed in Table 5.

Table 4. Training, run time complexity, processing time of used algorithms

ML Algorithm	Training Time Complexity	Run Time/Space Complexity	Processing Time (s)
NB	O ($d * n$)	O ($d * c$)	34.10
SVM	O (n^2)	O ($d * k$)	14.13 (RBF)
K-NN	O ($d * n * k$)	O ($d * n$)	18.22
DT	O (d *log(n)* n)	O (maximum tree depth)	24.4
RF	O (d *log(n) *n * k)	O (tree depth* k)	20.91

Table 5. Comparison between proposed and existing methods

Contribution	Algorithm	Confidence Score	Accuracy
Reference [25]	Hybrid framework	1	86.00
Reference [26]	HRFLM	1	88.70
Reference [27]	ANFIS	1	91.00
Reference [28]	ANN-fuzzy-AHP	1	91.10
Reference [29]	Stacked SVM	1	91.11
This work	RF and SVM implemented in Parallel	2	94.50

5 Embedded system architecture

For the design of an embedded intelligent system, the functional requirements include real time computational performance, storage, communication, easy algorithm implementation with low cost, etc. The hardware requirements include the following:

- Multi-core computer with application-specific integrated circuit controller
- RAM Memory and its controllers to run programs and store transient data.
- Interrupt controllers, which by standard make up a part of the CPU
- System HW timers for time-constrained events that
- Parallel and Serial I/Os
- Power monitors, which produce interrupt and reset at voltage drops
- Mass Storage Interfaces for highly integrated CPU and ASIC device
- Connection to High-speed network
- Realtime operating system such as Linux OS or ProConOS

Based on these requirements, the overall architecture can be devised as shown in Figure 2. The development computer shown in the Figure is the one, where machine learning model(s) are trained using Cleveland database. The sensors are the devices attached to the body of the patient to measure real time data and send them to an embedded system. An example of an embedded system is a 32-bit ARM9 processor (4 cache coherent cores) with real time Linux operating system, ProConOS as control kernel and SQlite embedded database. The embedded system runs the trained model(s) on patient data to provide real time predictive analysis to the physician. Thus, the cost to be borne by the hospital for this improved heart disease detection is dependent on this embedded system only, as other hardware is generally available at hospitals.

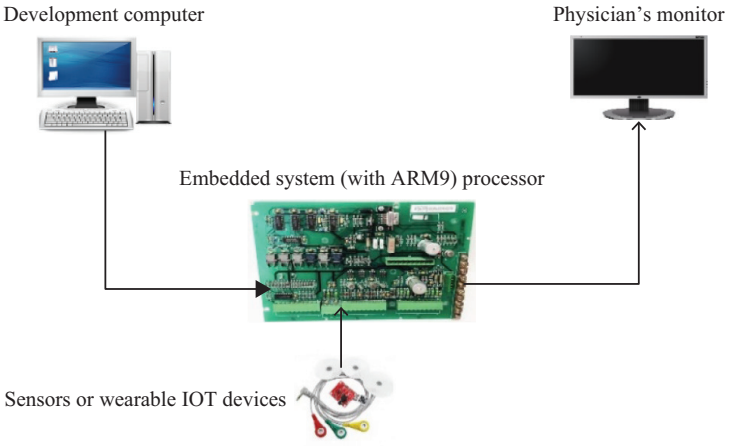

Fig. 2. Architecture of real time embedded system

6 Conclusions

The objective of this research was to figure out the effective machine learning system for detecting cardiac problems. On a well-known heart disease dataset, a chosen set of machine learning algorithms with feature selection was applied to assess the parameters for ranking. The computational complexity of each machine learning model and its performance were chosen to build robustness and increase confidence.

The results of the experiment demonstrate that the Random Forest was the top performer with accuracy of 94.5% followed closely by Support Vector machine algorithm with accuracy of 91.7% on selected feature space. Thus, these two top performing classifiers turned out to be promising classifiers for heart disease prediction. Since the testing of these classifiers is typically conducted on an embedded system for real time prediction, the characteristics of these classifiers can be combined to generate robustness in heart disease detection. For a robust heart disease prediction in an embedded platform, the proposed system stated that heart disease is detected with higher confidence if and only if more than one classifier detects it, and thus provide confidence of that majority vote. It turns out that employing fewer algorithms to detect cardiac disease is sufficient, as opposed to using an ensemble of algorithms. As far as future direction in this research is concerned, a number of strategies can be thought about. The first and foremost is to implement a detection system for heart illness in real time. For this, a possible architecture of a real time embedded system has been discussed in section five. To support this, a real time heart hardware implementation of RF and SVM have been documented [8, 30]. Thus, real time implementation of the whole system can be easily facilitated by deploying currently available embedded system products that implement classifiers in parallel.

Another direction is to facilitate the hospital cardiac team through online social network, similar to the works in [31–32] where patient data can be transported to the hospital system for real time monitoring. This study may also be improved by using a larger heart disease repository to improve training results which may lead to better detection results.

7 References

[1] Arabasadi, Z., Alizadehsani, R., Roshanzamir, M., Moosaei, H., Yarifard, A., "Computer aided decision making for heart disease detection using hybrid neural network-genetic algorithm," *Computer Methods and Programs in Biomedicine*, 141, 19–26, 2017, https://doi.org/10.1016/j.cmpb.2017.01.004

[2] Olaniyi, E., Oyedotun, O., Khashman, A., "Heart Diseases Diagnosis Using Neural Networks Arbitration," *International Journal of Intelligent Systems Technologies and Applications*, 7, 12, 75–82, 2015, https://doi.org/10.5815/ijisa.2015.12.08

[3] Chow, E. J., et al. "Underdiagnosis and Undertreatment of Modifiable Cardiovascular Risk Factors Among Survivors of Childhood Cancer," *Journal of the American Heart Association*, 11, 12, 2022, https://doi.org/10.1161/JAHA.121.024735

[4] Detrano, R., Janosi, A., Steinbrunn, W., Pfisterer, M., Schmid, J., Sandhu, S., Guppy, K., Lee S., Froelicher, V., "International Application of a New Probability Algorithm for the Diagnosis of Coronary Artery Disease," *The American Journal of Cardiology*, 64, 5, 304–310, 1989, https://doi.org/10.1016/0002-9149(89)90524-9

[5] Gudadhe, M., Wankhade, K., Dongre, S., "Decision Support System for Heart Disease Based on Support Vector Machine and Artificial Neural Network," *International Conference on Computer and Communication Technology*, 741–745, 2010, https://doi.org/10.1109/ICCCT.2010.5640377

[6] Kahramanli, K., Allahverdi, N., "Design of a Hybrid System for the Diabetes and Heart Diseases," *Expert Systems with Applications*, 35, 82–89, 2008, https://doi.org/10.1016/j.eswa.2007.06.004

[7] Palaniappan, S., Awang, R., "Intelligent Heart Disease Prediction System Using Data Mining Techniques," *International Journal of Computer Science and Network Security*, 8, 8, 290–294, 2008, https://doi.org/10.1109/AICCSA.2008.4493524

[8] Das, R., Türkoglu, I., Şengür, A., "Effective Diagnosis of Heart Disease through Neural Networks Ensembles. *Expert Systems with Applications*, 36, 7675–7680, 2009, https://doi.org/10.1016/j.eswa.2008.09.013

[9] Mahmoodi, D., Soleimani, A., Khosravi, H., Taghizadeh, M., "FPGA Simulation of Linear and Nonlinear Support Vector Machine," *Journal of Software Engineering and Applications*, 4, 320–328, 2011, https://doi.org/10.4236/jsea.2011.45036

[10] Elnawawy, M., Sagahyroon, A., Shanableh, T., "FPGA Based Network Traffic Classification Using Machine Learning," *IEEE Access*, 8, 175637–175650, 2020, https://doi.org/10.1109/ACCESS.2020.3026831

[11] Memon, Q., Hassan, M., "Cluster Analysis of Patients' Clinical Information for Medical Practitioners and Insurance Companies," *International Journal of Online and Biomedical Engineering*, 16 (4), pp. 128–138, 2020, https://doi.org/10.3991/ijoe.v16i04.13119

[12] Memon, Q., Valappil, N. "On Multi-Class Aerial Image Classification using Learning Machines," *Computer Vision and Recognition Systems Using Machine and Deep Learning Approaches: Fundamentals, Technologies and Applications*, Chap. 15, pp. 351–384, IET Digital Library, 2021, https://doi.org/10.1049/PBPC042E_ch15

[13] Thomas, J., Princy, R., "Human heart disease prediction system using data mining techniques," *International Conference on Circuit, Power and Computing Technologies*, pp. 1–5, 2016, https://doi.org/10.1109/ICCPCT.2016.7530265

[14] Manogaran, G., Varatharajan, R., Priyan, M., "Hybrid Recommendation System for Heart Disease Diagnosis based on Multiple Kernel Learning with Adaptive Neuro-Fuzzy Inference System," *Multimedia Tools and Applications*, 77, pp. 4379–4399, 2018, https://doi.org/10.1007/s11042-017-5515-y

[15] Tomov, N., Tomov, S., "On Deep Neural Networks for Detecting Heart Disease," 2018, ArXiv, abs/1808.07168

[16] Nagamani, T., Logeswari, S., Gomathy, B., "Heart Disease Prediction using Data Mining with Mapreduce Algorithm", *International Journal of Innovative Technology and Exploring Engineering*, 8 Issue-3, 2019.

[17] Shuvo, S., Ali, S., Swapnil, S., Al-Rakhami, M., Gumaei, A., "CardioXNet: A Novel Lightweight Deep Learning Framework for Cardiovascular Disease Classification Using Heart Sound Recordings," *IEEE Access*, 9, 2021, pp. 36955–36967, https://doi.org/10.1109/ACCESS.2021.3063129

[18] Ali, S., et al., "An Optimally Configured and Improved Deep Belief Network (OCI-DBN) Approach for Heart Disease Prediction Based on Ruzzo-Tompa and Stacked Genetic Algorithm," *IEEE Access*, 8, 2020, pp. 65947–65958, https://doi.org/10.1109/ACCESS.2020.2985646

[19] Georgieva-Tsaneva, G., Gospodinova, E., Bogdanova, G., Dimitrova, D. (2023). An Information Cardiac Platform to Support Healthcare, International Journal of Online and Biomedical Engineering, 19, 03, pp. 144–153, https://doi.org/10.3991/ijoe.v19i03.35217

[20] Shaaf, Z. F., Muhammad Mahadi Abdul Jamil, Radzi Ambar. (2023). A Convolutional Neural Network Model to Segment Myocardial Infarction from MRI Images, *International Journal of Online and Biomedical Engineering*, 19(02), pp. 150–162. https://doi.org/10.3991/ijoe.v19i02.36607

[21] Domyati, A., Memon, Q., "Robust Detection of Cardiac Disease Using Machine Learning Algorithms," *Proceedings of the 5th International Conference on Control and Computer Vision*, Association for Computing Machinery, New York, NY, USA, 52–55, 2022, https://doi.org/10.1145/3561613.3561622

[22] Chikhi, S., Benhammada, S., "ReliefMSS: A Variation on a Feature Ranking Relief Algorithm," *International Journal of Business Intelligence and Data Mining*, 4, pp. 375–390, 2009, https://doi.org/10.1504/IJBIDM.2009.029085

[23] Azad, C., Jha, V., "Genetic Algorithm to Solve the Problem of Small Disjunct in the Decision Tree Based Intrusion Detection System," *International Journal of Computer Network and Information Security*, 2015, pp. 56–71. https://doi.org/10.5815/ijcnis.2015.08.07

[24] Cortes, C., Vapnik, V., "Support-Vector Networks," *Machine Learning*, pp. 273–297, 1995, https://doi.org/10.1007/BF00994018

[25] Haq, A., Li, J., Memon, M., Nazir, S., Sun, R., "A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms", *Mobile Information Systems*, 2018, Article ID 3860146. https://doi.org/10.1155/2018/3860146

[26] Mohan, S., Tirumalai, C., Srivastava, G., "Effective Heart Disease Prediction using Hybrid Machine Learning Techniques," *IEEE Access*, 7, 2019, pp. 81542–81554. https://doi.org/10.1109/ACCESS.2019.2923707

[27] de Silva, A., Leong, P., Grammar-Based Feature Generation for Time-Series Prediction, Springer, Berlin, 2015. https://doi.org/10.1007/978-981-287-411-5

[28] Samuel, O., Asogbon, G., Sangaiah, A., Fang, P., Li, G., "An Integrated Decision Support System Based on ANN and Fuzzy AHP for Heart Failure Risk Prediction," *Expert Systems and Applications*, 68, pp. 163–172, 2017, https://doi.org/10.1016/j.eswa.2016.10.020

[29] Ali, L., et al., "An Optimized Stacked Support Vector Machines Based Expert System for the Effective Prediction of Heart Failure," *IEEE Access*, 7, pp. 54007–54014, 2019, https://doi.org/10.1109/ACCESS.2019.2909969

[30] Paul, A., Shill, P., Rabin, R., Murase, K., "Adaptive Weighted Fuzzy Rule-Based System for the Risk Level Assessment of Heart Disease," *Applied Intelligence*, 48, 7, 1739–1756, 2018, https://doi.org/10.1007/s10489-017-1037-6

[31] Memon, Q., Mustafa, A., "Exploring Mobile Health in a Private Online Social Network," *International Journal of Electronic Healthcare*, 8, 1, pp. 51–75, 2015, https://doi.org/10.1504/IJEH.2015.071643

[32] Ali, F. I., Ali, T. E., Al_dahan, Z. T., "Private Backend Server Software-Based Telehealthcare Tracking and Monitoring System," *International Journal of Online and Biomedical Engineering*, 19, 01, pp. 119–134, 2023, https://doi.org/10.3991/ijoe.v19i01.32433

8 Authors

Anas Domyati started his Master's degree in Electrical Engineering at Department of Electrical Engineering at UAE University, Al Ain in 2020., and completed it in June 2022. His research interests include Machine Learning applications in health.

Qurban Memon obtained his Ph.D. degree from University of Central Florida, Orlando, US. Currently, he is working as Associate Professor at UAE University, College of Engineering, United Arab Emirates. He has edited two books, and published over a hundred research papers in his academic career. He has executed research grants and development projects in the area of intelligent systems, security and networks. He has served as a technical committee member of many international journals and conferences, as well as session chair at various conferences.

Article submitted 2022-12-14. Resubmitted 2023-04-16. Final acceptance 2023-04-16. Final version published as submitted by the authors.