IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Journal of New Materials for Electrochemical Systems
Vol. 28, No. 1, January 2025, pp 24-34
Journal homepage: http://iieta.org/journals/jnmes

A Novel Framework for Deep Convolutional Neural Network-Based Heart Failure Disease Prediction Using an Optimized Efficient net-B0 Model

JiniMol G [1*], Ajith Bosco Raj T [2]

[1] Department of CSE, Arunachala College of Engineering for Women, Vellichanthai, Tamil Nadu, India
[2] Department of ECE, PSN College of Engineering and Technology, Tirunelveli, Tamilnadu, India.

Corresponding Author Email: drbright.sa@gmail.com

https://doi.org/10.14447/jnmes.v28i1.a02

Received: April 09-2024
Accepted: December 20-2024

Keywords:
Heart Failure Diseases (HFDs), Deep Convolutional Neural Network (DCNN), Electronic Health Records (ERHs), EfficientNet-B0 model.

ABSTRACT

Convolutional neural networks (CNNs) have been widely used in medical decision support systems to accurately predict and diagnose various diseases. Because of their ability to identify relationships and hidden patterns in healthcare data, CNNs have been extremely successful in developing health support systems. One of the most important and useful application systems is in the prediction of Heart Failure Diseases (HDFs) by observing cardiac anomalies. Fundamentally, CNNs have multiple hyper parameters and various specific architectures, which are costly and impose challenges in selecting the best value among possible hyper parameters. Furthermore, CNNs are sensitive to hyper parameter values, which have a significant impact on the efficiency and behavior of CNN architectures. Datasets from Electronic Health Records (EHRs) have recently been used to diagnose a variety of diseases, including heart failure. In this paper, we proposed the Deep convolutional neural network algorithm (DCNN), which is one of the deep learning algorithms that has been successfully used to solve computer vision problems. In our work, EfficientNet-B0 is a type of DCNN model that is used with a transfer learning approach to recognize diseases in Heart Failure images. To determine the effect of transfer learning with fine tuning, we assessed the performance of all EfficientNet-B0 variants on this imbalanced multiclass classification task using metrics such as Specificity, Recall, Accuracy, F1-measure, and Confusion Matrices. However, the accuracy and parameters of EHRs-based HFDs diagnosis are limited by the lack of an appropriate feature set. The experimental results show that EfficientNet-B0 achieves higher accuracy 98.45% with fewer parameters than the five classical DCNN models, demonstrating that the DCNN-EfficientNet-B0 model achieves more competitive results on HFDs identification.

1. INTRODUCTION

Cardiovascular disease, also known as heart disease, is a major public health concern owing to the fact that it is the leading cause of death worldwide [1-3]. Heart disease impairs blood vessel function and causes coronary artery infections, especially in adults and the elderly. According to a World Health Organization (WHO) report, over 18 million deaths occur worldwide each year. Furthermore, the United States spends $1 billion per day on cardiovascular care. Hypertension, heart attacks, and stroke are the leading causes of death in the United States. Despite such dire consequences, heart disease is frequently preventable if detected early. As a result, it is critical to detect cardiac conditions early in order to treat cardiac patients effectively prior to a heart attack or stroke. To that end, automated cardiovascular prediction is one of the most critical and challenging tasks worldwide.

CNN has performed well in many image processing (IP) and computer vision (CV) competitions because it efficiently extracts features from images [4]. Image segmentation and classification [5, 6], video processing, speech recognition, and natural language processing (NLP) are some of the current applications of CNN. CNN's powerful learning ability stems primarily from the use of several feature extraction steps that can learn automatically from data. Furthermore, the availability of a large volume of data as well as advanced hardware technology has increased the amount of research done in CNNs. Recently, powerful CNN architectures such as Inception and 1D CNN [7] have been developed.

If the classifiers in deep convolution neural networks (DCNN) are of the same type, the network is homogeneous; if they are of different types, the network is heterogeneous [8]. Every DCNN is trained with its own feature space, and in some cases, the features may contain noise that includes duplicate and unwanted data. In such cases, the training time is longer, and the false-positive rate is higher. The feature selection technique is used to solve this problem. The feature selection technique is primarily employed in classification ensembles. It provides better results with optimised features when using feature selection for the DCNN [9Swarm and other algorithms are used to obtain the optimised feature subset. Many features algorithms, such as PSO, genetic algorithms, support vector machines, and machine-learning algorithms,

are used to optimise. [10] created the artificial crow colony algorithm to solve feature selection problems. The ABC has been used in a variety of domains since its inception.

Surgical treatment of CVD is a difficult task, especially in developing countries due to a lack of trained personnel, testing equipment, and other resources [11]. Physical examinations, despite their high cost, are not without flaws and imperfections. Electronic health records (EHRs) have recently demonstrated a significant potential to provide useful insights for clinical medicine research. A large body of work has been presented using EHRs for CVD as well, and they can be very useful for early CVD diagnosis. Both machine learning and deep learning models have been used to predict CVD; however, both have limitations. The provided prediction accuracy and F measures, for example, are insufficient. Similarly, the results are not generalizable, as using a different dataset might show very different results concerning true positive rates. The last but most important problem is feature engineering which has a direct influence on the performance of prediction models. Predominantly, the EHRs have a low number of features and a good fit of the model is not achieved which leads to poor prediction performance.

We investigate the classification performance of the EfficientNets B0 on the EHR dataset of Cardiovascular images in this paper [12]. The dataset contains 10015 images from 12 different heart disease classes, including age, gender, type of chest pain, blood pressure, fat, glucose level, ECG result, heart rate, exercise angina, old peak, ST-slope, and heart disease. To perform transfer learning and fine-tune the CNNs for the EHR dataset, we used ImageNet pre-trained weights. Precision, Recall, Accuracy, F1 Score, Specificity, Roc Auc Score, and Confusion Matrices were used to assess the performance of EfficientNets B0 on this imbalanced multiclass classification task. This paper also presents the per-class classification exactitudes for all five models in the form of Confusion Matrices. Our best model, the EfficientNet B0, in particular, achieved an accuracy of 98.45%. Our findings show that the EfficientNets B0 outperforms other algorithms for heart failure disease classification on the EMR dataset.

This research addresses these issues by combining features extracted by a Deep Convolutional Neural Network (DCNN) with machine learning models. In summary, the following contributions are made by this study:

- A novel method of using DCNN with deep learning models is developed, where CNN is used to enlarge the feature set and linear models are used for predicting HFDs.
- In this paper we use datasets from Electronic Health Records (EHRs) have recently been used to diagnose a variety of diseases, including heart failure.
- DCNNs are used to extract the features from the signal. Meanwhile, Cuckoo Search Algorithm (CSA) is used for hyperparameter tuning, by choosing the best model with 1D convolutional layers for feature extraction, each of which is followed by one max-pooling layer, followed by one dense layer, and finally an output layer.
- Performance evaluation is carried out on four different datasets for results generalization using accuracy, precision, recall, and F1 score. Performance analysis is realized with several state-of-the-art approaches, and deep learning DCNN and transfer learning based EfficientNet B0 models are also implemented. Results show significant improvement in HFDs prediction.

The section of this document is organised as follows. Introduction and a review of the literature are presented in Chapter 1. The DRCNN-ResNet Rs Transfer learning framework for Brain tumour Detection and Classification is demonstrated in Chapter 2. With discussion, Chapter 3 presents the conclusion. Chapter 4 completes the manuscript's conclusion.

2. RELATED WORKS

Wang, Zhu, Li, Yin, and Zhang (2020) proposed using actual data from an Electronic Health Records (EHRs) information system to predict HF mortality in a feature rearrangement-based DL (FRDLS). The proposed FRDLS was successful in predicting HF mortality for three distinct targets: in-hospital, one-month, and one-year mortality predictions. First, the system selects and reprocesses features from the EHRs system. The collected data from the EHRs system was then modelled using a feature rearrangement-based Convolution Net (FReaConvNet). One limitation of this study was that the proposed system performance was dependent on two hyperparameters that needed to be adjusted over time. Furthermore, the prediction labels are death or alive, simplifying the complex HF problem. Finally, the proposed method did not handle multi-label, multi-class and other prediction tasks.

Reddy et al. (2021) [15] proposed an attribute evaluator-based machine learning-based system. The authors employ ten distinct machine learning models from various categories, including Bayesian-based models, tree-based models, rules-based models, and so on. To achieve high accuracy in heart disease prediction, they use all of the Cleveland dataset's attributes as well as the optimal attributes determined by the three attribute evaluators. The results show that sequential minimal optimization achieves an accuracy of 85.148% with the full set of attributes and 86.468% with the optimal set of attributes. Similarly, Perumal et al. (2020) [16] use the Cleveland dataset to develop a heart disease prediction model. The study uses feature standardization and feature reduction using principal component analysis for training the models. Findings indicate that accuracy scores of 87% and 85% can be accomplished with LR and SVM, respectively.

Amin et al. [21] sought to identify key features and data mining techniques for improving heart disease prediction accuracy. Using various attribute combinations and seven machine learning models, a series of predictive models are created. The best-performing model achieves an accuracy of 87.4%, according to the results. Mohan et al. [22-24] propose a heart disease prediction model (HRFLM) based on machine learning that combines a linear model with RF features. HRFLM works with a variety of feature configurations and classification techniques. The proposed HRFLM achieves the highest accuracy of 88.7%, according to experimental results. Similarly, uses six machine learning techniques to predict heart disease. The maximum accuracy of 85% is obtained using LR on the Statlog dataset. Both individual and ensemble learning approaches such as J48, MLP, Bayes Net, NB, RF, and random tree (RT) are investigated for CVD prediction. RT proved to be highly accurate with an accuracy of 70.77%. The

study subsequently employed a new-fangled method that achieved an accuracy of 80%.

Radhimeenakshi (2016) [4] proposed a Decision Tree and Support Vector Machine for classification of heart disease. In terms of accuracy as measured by a confusion matrix, he concluded that the decision tree classifier outperforms SVM. R.W.Jones et al. (2017) [5] proposed an artificial neural network-based heart disease prediction technique. They trained the neural network using a self-administered questionnaire. A backpropagation algorithm was used to train the neural network, which had three hidden layers. Using the Dundee rank factor score, the architecture was validated and achieved a 98% relative operating characteristic value on the dataset.

Ankita Dewan et al. (2015) [6] compared the performance of genetic algorithms and backpropagation for neural network architecture training. They came to the conclusion that backpropagation algorithms perform better with very little error on the dataset. SY For training the artificial neural network, Huang et al. (2018) [7] proposed a learning vector quantization algorithm. They trained the network with 13 clinical features and achieved nearly 80% accuracy on the dataset. Jayshril S. Sonawane et al. [8] proposed a new artificial neural architecture that can be trained with a vector quantization algorithm and random order incremental training. They also trained with 13 clinical features and achieved 85.55% accuracy on the dataset. Majid Ghonji Feshki et al. (2016) [9] used four different classification algorithms which include C4.5, Multilayer Perceptron, Sequential Minimal Optimization, and feed-forward backpropagation. They concluded that the PSO algorithm with neural networks achieved the best accuracy of around 91.94% on the dataset.

R. R. Manza et al. (2019) [10] proposed an Artificial Neural Network in the hidden layer with a large number of Radial Basis Function neurons. On this architecture, they achieved around 97% accuracy. For feature selection, P. Ramprakash et al. [1-2] proposed a deep neural network and two statistical models. They used a variety of techniques to avoid over- and under-fitting. They were 94% accurate, 93% sensitive, and 93% specificity.

3. METHODOLGY

Deep learning (DL) is also known as hierarchical or deep-structured learning. DL, in contrast to task-based methods, is a type of ML technique that is based on learned-data representation, and the learning can be supervised, unsupervised, or semi-supervised. The models of DL are loosely inspired by the functioning of biological nervous systems, such as how information is processed and communicated in them. These DL techniques, however, are structurally and functionally distinct from human brains. Because of these differences, they are incompatible with neuroscience evidence. Deep convolutional neural networks (DCNN), DL networks, recurrent NN, and deep belief networks are DL architectures that have been used in various research areas such as recognising human speech, computer vision (CV), audio recognition, natural language manipulation, machine translation, filtering social sites, drug design, bioinformatics, medical image processing, board game programmes, and material examination. These advanced machine learning models outperformed, and in some cases outperformed, humans.

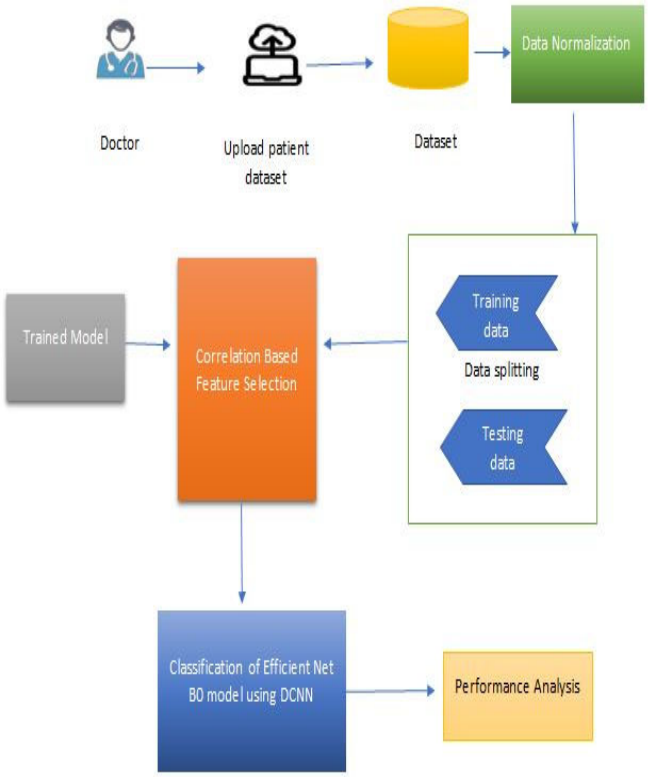


Figure 1: The Block Diagram of proposed methodology

The proposed method has been discussed using the block diagram in Fig 1. We would be able to understand how the process works and its details using this. We will use various processes to identify heart disease in this method. This method is useful for examining the heart's condition and avoiding time delays in treating the patient.

3.1 Dataset

This method was evaluated using Kaggle's Cleveland dataset, Cardiovascular Disease Prediction dataset, and Kaggle's cardiovascular disease Electronic Health Records (EHRs) dataset. Our forecast results show that we outperformed previous systems proposed by other writers. An EHR is a patient's electronic database, also known as an EMR (Electronic Medical Record). It is a computer-readable, computerised record of a patient's clinical state that licenced medical practitioners enter. The records include a patient's vital signs, prognosis, and medical exam findings. For our research, In Table 1 we used an electronic medical record dataset provided by Morgan Stanley as part of their health coder challenge for our research. The data can be found online from a variety of sources, including a free source of data.

Table 1. EHRs Dataset

SI No	Attribute	Description
1	Age of Patient	This attribute contains the age of a patient (years).
2	Gender	This attribute contains the type of gender M = male, F = female, TG-Trans gender

3	Chest pain type	This attribute contains the type of the chest pain experienced by the patient TA = typical angina, ATA = Atypical angina, NAP = non-anginal pain, Asy = asymptomatic
4	Blood Pressure	Resting blood pressure of the patient in mmHg
5	Fat	Serum cholesterol in mm/dL
6	Glucose level	Fasting blood sugar [1 = if fasting BS > 110 mg/dL, 0 = otherwise]
7	Resting ECG result	Normal = 1, ST = having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of >0.05 mV) LVH = showing probable or definite left ventricular hypertrophy by Estes' criteria
8	Heart rate	Max heart rate 71–202
9	Exercise Angina	Exercise-induced angina 1 = true; 0 = false
10	Old Peak	ST depression induced by exercise relative to rest
11	ST-Slope	Slop or the peak exercise ST segment Up = upsloping, Flat = flat, Down = downsloping
12	Heart Disease	Binary Target, Class 1 = heart disease, Class 0 = normal

In comparison, one Kaggle dataset with approximately seventy thousand patient records reports on cardiovascular illness. However, this Kaggle dataset only contains a few metrics for each patient record. Factors to consider include age, gender, type of chest pain, blood pressure, fat, glucose level, ECG result, heart rate, exercise angina, old peak, ST-slope, and heart disease.

3.2 Pre-processing

Pre-processing includes two stages: removal of redundant data and normalisation. Pre-processing prepares the system for further diagnosis. The dataset received from the Doctor is then processed after pre-processing.

3.2.1 Removal of redundant data

The most important component of global optimizers, partial redundancy elimination (PRE), generalises the elimination of regular subexpressions and loop-based invariant calculations. Existing PRE implementations fail to eliminate redundancies because they rely on code motion. In fact, we discovered that code mobility alone cannot eliminate 73 percent of loop-invariant statements. In dynamic terms, traditional PRE removes only half of the strictly partial redundancies. Control flow restructuring is required to obtain a complete PRE. However, the resulting code duplication may increase code size. This research focuses on completing a PRE while maintaining a reasonable rate of code growth. First, we describe a method for completely eliminating partial redundancies using a combination of code motion and control flow restructuring. In contrast to current comprehensive approaches, we use restructure to remove barriers to code movement rather than true optimization.

3.2.2 Normalization

Normalization is a necessary step before solving any problem statement. Normalization is especially important for fuzzy systems, cloud services, and other methods of changing data, such as reducing or increasing the size of various inputs before they are used in a later step. Normalization methods include "inMax normalisation," "Z-score normalisation," and "Decimal scaling normalisation," to name a few. We will discuss a new normalisation method called Integer Scaling Normalization, which is based on these other methods. We'll also use various data sets to demonstrate how our proposed normalisation method works. The heart disease database contains a large number of features, each with its own set of numerical values, making the computational task more difficult. As a result, a normalisation strategy is used to normalise values in the range of 0 to 1 and minimises numerical complexity during the simulation process of Heart Failure disease prediction. There are several approaches to data normalisation. In the proposed system, another well-minimum-maximum normalising approach is used.

$$\text{EWnorm} = \frac{E^{ie} - EW_{min}}{EW_{Max} - EW_{min}} \text{Y[new max -new min]} + \text{new-min} \quad \text{---} \quad (1)$$

The training and testing datasets are separated from the input dataset. There is a variable in the training dataset that must be predicted or identified as an output. All prediction and classification algorithms Apply the patterns discovered in the training sample to the test dataset.

3.2.3 Correlation based feature selection (CBFS)

Feature selection is a powerful technique for reducing dimensionality, eliminating extraneous input, and improving learning accuracy. In terms of efficiency and efficacy, many current feature selection approaches are severely hampered by the curse of data dimensionality. This study employs Fast ("Correlation Based Feature Selection") FCBF, a version of (FCBF) dubbed ("Fast Correlation Based Feature Selection") FCBF, and Fast Correlation Based Feature Selection in parts (FCFBiP). The FCFBiP is shown to be more efficient than the FCBF and FCBF when these featured options are compared. Wrapper selection algorithms are used by FCBF. In the FCBF feature selection process, all features are considered first, followed by sequential backward removal. Procedures for determining feature reliance before using symmetric uncertainty values. These steps are repeated until the dataset cannot be removed any longer. FCBF employs the sequential forward selection method. These methods perform better when it comes to selecting features from the quality dataset of features with respect to k. (FCBF) eliminates sections that are only tangentially related to the class. Each iteration will remove only one feature, resulting in a more balanced removal process. This process is repeated until no further elimination is possible. Nonetheless, a k-feature data subset is created, and cross-validation is used. FCBFiP is a new FCBF variantThe number of features in the original dataset is divided into P parts in each elimination phase, and the feature with the lowest score is deleted. The previous

version consists of two major steps: the first is to evaluate relevant features to the target class based on their relationship, and the second is to evaluate relevant aspects to the target class based on their relationship. The second stage involves searching for duplicates in the given data and then deleting one characteristic at a time from the entire dataset while selecting the majority subset list.

4. PROPOSED METHODOLOGY

4.1 Deep Convolution Neural Network Using

DL enables mathematical models with multiple processing units to learn data representations at various levels of abstraction. These methods have advanced the state-of-the-art in speech recognition (Alom et al., 2019), visual object recognition (Baozhou et al., 2020), object detection (Liu et al., 2020), and a variety of other fields such as drug discovery and genomics (Lee et al., 2019). Meanwhile, training a large neural network to solve a simple problem causes the NN to memorise the answers and over-fit the training data. Building a simple NN to solve a complex problem, on the other hand, fails to generate the correct answers for training and testing data, resulting in an under-fitting problem (Liu et al., 2020). Therefore, choosing the best architecture of the model is important, and this is done by tuning. Tuning means training more than one model and picking the best, which could cause a computational issue.

To provide a better solution to optimization problems, the Cuckoo Search Algorithm (CSA) is combined with the DCNN. The DCNN is a hybrid of four existing algorithms: VGG-19, ResNet-152, DenseNet-201, and Inception ResNet-V2. In the proposed model, the CSA algorithm, the artificial crow colony is used to find the features and the subset of feature generator, and DCNN is used to evaluate the feature subset obtained. The DCNN is used to locate the feature proposed by the EfficientNet B0 model. The proposed EfficientNet B0 assists the DCNN in constructing the best feature subset. The EfficientNet B0 trained algorithm and DCNN improve the CS algorithm's performance (Fig. 2).

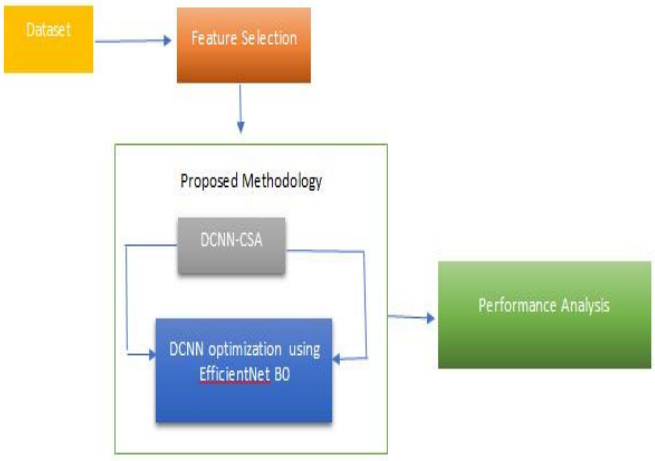


Figure 2 Proposed model for heart disease prediction Using DCNN

4.2 Implementation of DCNN-CSA

The Cuckoo Search Algorithm is an intelligent algorithm used for the feature optimization process and this colony and genetic algorithm increases the accuracy of the ensemble. The DCNN-CSA is combination of four existing models such as VGG-19, ResNet-152, DenseNet-201 and Inception ResNet-V2 with these algorithms, the ability of each feature in subset can be calculated, Fi in the feature subset. 10-fold technique is used to find the accuracy of available feature in the subset. Each employed crow is represented in binary string 0 or 1. The total number of features is the same as the length of the dataset, and it also represents feature selection done by the crow search. In the binary string, 1 represents the feature is selected and 0 represents feature is not selected. The total amount of onlooker and crow search is the same as the number of features in the dataset (Fig. 3).

The Cuckoo search is used after the genetic process for the initial population has been completed. The extraction of features from the dataset is used to initialise crows. Initially, Ci is a feature based on crows arbitrarily located in a search space, as defined by the following equation:

$$C_i = C - 1, C_2, \ldots, C_n, \text{ where, } i = 1, 2, 3 \ldots n \quad \text{-- (2)}$$

Few initial solutions are used at the starting stage for the meta-heuristics optimization models for improving by monitoring the contrast solution simultaneously. The opposition point definition can be written as per the following equation:

$$\hat{c}_i = g_j + h_j - c_i \quad \text{-- (3)}$$

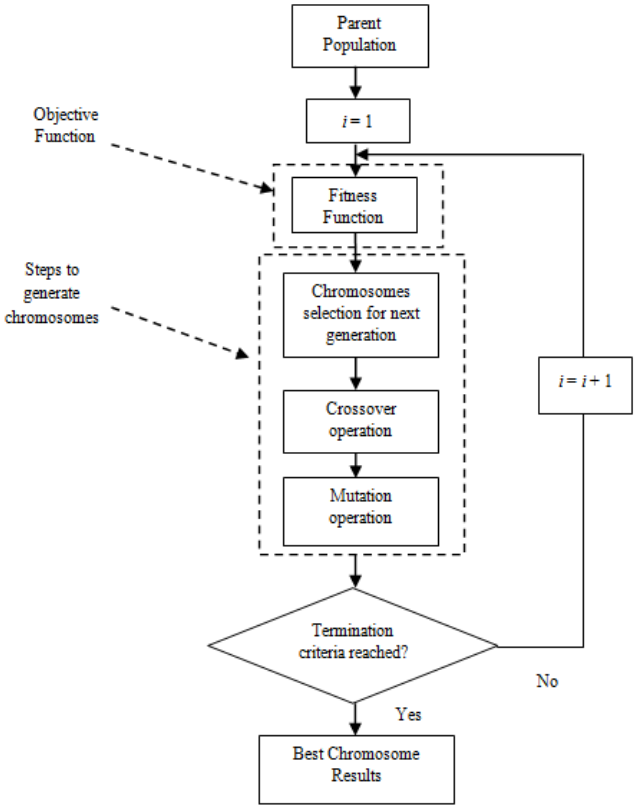


Figure 3 Flowchart process of proposed algorithm

The fitness function for the OCS can be calculated based on the basis function as per the following equation:

$$OC_i = MAX(Accuracy) \quad \text{-- (4)}$$

One of the flock crows is randomly chosen to form a novel position and the innovative position of the crow is obtained using the following equation:

$$PC^{i+iter+1} = \begin{cases} PC_{i+iter+1} + a_i \, x \, cl^{i.iter} \\ x\,(M^{j.iter} - PC^{i.iter}) \quad if\ a_j \geq APC^{j.iter} \\ ranPC \quad otherwise \end{cases} \quad \text{-- (5)}$$

The current position and memory of the upgraded crow is processed based on the following equation

$$M^{i.iter+1} = \begin{cases} PC^{i.iter} \\ M^{j.iter} \end{cases} otherwise \; c(PC^{j.iter+1}) > c(M^{j.iter}) \quad \text{-- (6)}$$

The fitness value of the crow's new position is controlled to be the highest in the sustained position. Crow often updates storage space with new locations. If multiple iterations are implemented, the best memory location corresponding to the target will be addressed as the best filtering feature set solution. The suitability of the role, the health assessment is done through the function of each crow:

$$\text{fitness}_1(S) = \frac{\sum_{j=1}^{m} accuracy\ (S)}{m} \quad (7)$$

$$\text{fitness}_2(S) = \text{consensus}(S) \quad (8)$$

$$\text{fitness} = \frac{\text{fitness1}(S) + \text{fitness2}(S)}{2} \quad (9)$$

The accuracy(S) is the predicted value in the ensemble classifier, consensus(S) represents the accuracy of classification on the S feature subset. The fitness can be evaluated using mean accuracy and consensus, mean accuracy checks whether the features have the power for accurate classification, the DCNN tries for feature optimization. The mean accuracy helps in increasing the ability of feature subset generalization. The second phase of fitness evaluation is consensus; it finds whether the feature subset has optimality in classification for producing high consensus classification. The crow search passes the information to the onlooker crow and it checks the likelihood of feature selection using Eq. (7), the new solution given by onlooker crow is represented as V_i, using the mean and consensus value of the feature the crow search points out the feature selected by the onlooker crow. If the newly obtained value V_i is larger than x_i, then the crow search points to the feature in the feature subset that has crown previously selected and the new one. If the value of V_i is less compared to X_i, then crow search features will be used for further process, and the newly selected feature is omitted. The Vi is obtained using the following formula:

$$P_i = \frac{fitness}{\sum_{i=1}^{m} fitness} \quad (10)$$

$$V_i = X_i + \mu_i\,(X_i - X_j) \quad (11)$$

where X_i is the selected feature's accuracy, X_j is the accuracy of feature selected by onlooker crow. μ_i is the randomly generated number in range (0, 1).

Therefore, when the crow search is allocated with a new feature, the onlooker crows make full use of it and a new configuration for the subset is produced, after this process, all the features are used for forming a new feature subset, the available content in features tries to move to better feature subset configuration. If the improvement is not made in crow search, then the employee becomes scout crow. Then, the new feature subset is assigned to the scout crow that is represented as follows:

$$X_{ij} = X^{max}{}_j + rand(0, 1)(X^{max}{}_j - X^{min}\,j) \quad (12)$$

where $X^{max}{}_j$ is the upper boundary value and $X^{min}{}_j$ is the lower boundary value. The upper and lower boundaries and the same process have crown carried out till the stopping criteria are achieved to get the best features. The pseudocode of the proposed algorithm is followed by

Algorithm 1 The pseudocode for the proposed DCNN-CSO
1: Feature set is taken as an initial population for the CSA.
2: Evaluate fitness function for the optimal feature.
3: for each reproduction of optimal feature do
4: Selecting chromosomes based on the fitness function of the feature.
5: Crossover of the randomly generated feature set. Mutate child chromosomes with its mutation probability.
6: New population generated based on CS
7: Obtain opposition process for initial population
8: Calculate fitness function for opposition processed population
9: Calculate fitness function for Optimal solution from CS process
10: Obtain best solution
11: Calculate the objective function for best solution
12: Generate new position
13: if new population = feasible then
14: Update Memory and Position
15: else
16: Stop with the current best population
17: if Max iteration is obtained then
18: return best solution
19: else
20: Repeat until the iteration process and go to step 12
21: End
22: Best Feature set is selected as optimal solution

Algorithm 2 Pseudo−code of Cuckoo Search (CS) algorithm.
Begin:
Initialize cuckoo population:n
Define d−dimensional objective function, f(x)
do Until iteration counter < maximum number of iterations
global Search:
generate new nest $x^{t+1}{}_i$
evaluate fitness of $x^{t+1}{}_i$
choose a nest j randomly from n initial nests.
if the fitness of $x^{t+1}{}_i$ better than that of $x^t{}_i$
replace j by $x^{t+1}{}_i$
end if
local search:
abandon some of the worst nests using probability switch.
create new nest
Evaluate and find the best.
end until
update final best
End

The obtained optimal solution is applied as input for the Cuckoo Search algorithm and evaluated for the fitness of each feature and based on the chromosome, the child is selected as features and mutation process for that child-chromosome is

selected and repeated until the given iteration value. From this, the CSA can select the features based on their ranking, so the important feature is selected from the feature subset; the time consumption caused by the noisy and unwanted features is reduced. In case of large datasets, the classifier performance is reduced because of the huge number of features handled. In the CS algorithm, the features are selected based on their importance and classifier computation speed is enhanced.

4.3. Transfer learning

Transfer learning, also known as domain adaptation, is a highlevel concept that utilizes the knowledge acquired in a domain or task to solve related tasks. We leveraged this previously learned knowledge from the models trained on the ImageNet dataset and used their parameters for our task. However, our approach evaluates EfficientNet models on a medical image dataset of pigmented skin lesions. Due to the difference in the domains of the dataset images, we cannot directly use the pre-trained weights for inference and expect high performance. Thus, we performed a finetuning process.

In this step, the [trained] model's parameters are tweaked precisely to adapt to the new domain of the images. There are many ways to do fine-tuning. These include finetuning all or some parameters of the last few layers of a pretrained model or utilizing a pre-trained model as a fixed features' extractor from which features better feed each classifier, i.e., CNN for classification. We employed both transfer learning and fine-tuning in EfficientNets B0.

Figure 3 depicts the visualised modification for EfficientNet B0. The figure depicts the block diagram of the official EfficientNet B0 baseline network, as well as the enhancements we made to the EfficientNet B0 architecture, which are highlighted with a blue border. The base model (i.e., feature extractor blocks) remained unchanged; instead, the top layers of the EfficientNet B0 architecture were modified. Because the official B0 network has three top layers (global average pooling 2D, dropout, and dense layer), the model overfits. We modified the top layers and added additional dense, batch normalisation, and drop out layers on top of the B0 base architecture, using swish activation function instead of RELU activation function for dense (i.e., fully connected) layers.

EfficientNet B0 classifying models are combined to form an ensemble classifier in this paper. In this work, crows look for features, and the features chosen by each crow become the input to the classifier. The features are evaluated one at a time, and each one must be evaluated separately. The test subset is classified using the four existing classifier models GoogleNet, VGGNet, AlexNet, and ResNet. Following classification, the cuckoo search algorithm employs a formula to compute the ensemble's mean accuracy and consensus. The fitness of the features is calculated by taking the mean accuracy and consensus and averaging them. The fitness function is used during the feature selection process.

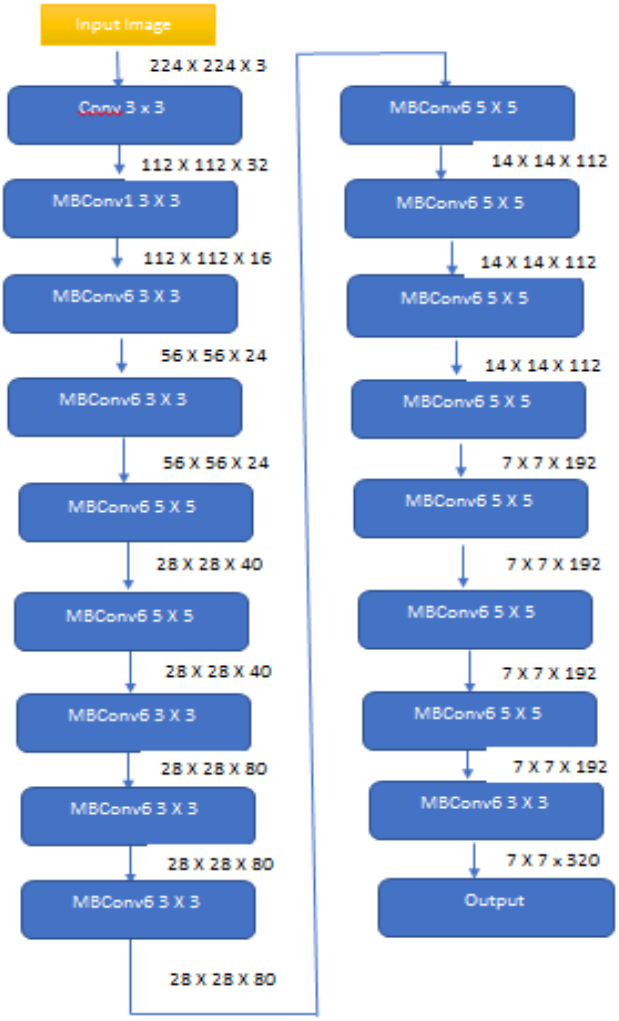


Figure 3 Block Diagram EfficientNet B0

4. RESULT AND DISCUSSION

The proposed architecture for heart disease prediction was built with scikit-learn and Kera's library, which enables the implementation of various deep learning algorithms. The development system includes an Intel i5 CPU and 8GB of RAM. It also has a GeForce 940 GPU, which aids in the training of the architecture. The paper makes use of the EMRS Kaggle dataset, which contains 300 patient samples with 12 different features. The dataset is split into two sections. The remaining 11% is used for validation, while the other 89% is used for training.

This phase discusses the dataset used in the proposed work's implementation and the results obtained when compared to other classifier algorithms.

5.1 Evaluation parameter

Several evaluation parameters, such as accuracy, precision, recall, and F1 score, are used to evaluate performance. These evaluation parameters, which include true positive (TP), true negative (TN), false positive (FP), and false negative (FN), can be calculated using the confusion matrix values:

- TP denotes when the model predicts a record as 0 and the actual label of the record is also 0;
- TN denotes when the model predicts a record as 1 and the actual label of the record is also 1;

- FP denotes when the model predicts the record as 1 and the actual label of the record is 0;

FN denotes when the model predicts the record as 0 and the actual label of the record is 1. Equations used to calculate accuracy, precision, recall, and F1 score are provided as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

5.2 Performance Analysis

This section describes the metrics used to evaluate the DCNN-based EfficientNet B0 model. There are no universal criteria for assessing the performance of any classification model. In the literature, we see a standard set of performance measures that vary depending on the user's needs. Precision, Recall, Accuracy, F1 Score, Specificity, Roc Auc Score, and Confusion Matrices are useful when classes are highly imbalanced (as shown in Table 1). As a result, we had to rely on them.

Confusion matrix

After the dataset has been presented in the form of a feature matrix, the next step is to divide it into different classes.

In our first example, we divided the dataset into two categories: heart disease and no heart disease. The model is then saved on the file system after being generated by using DCNN to classify the data into binary classes. The predicted heart disease confusion matrix, as shown in Fig. 4.

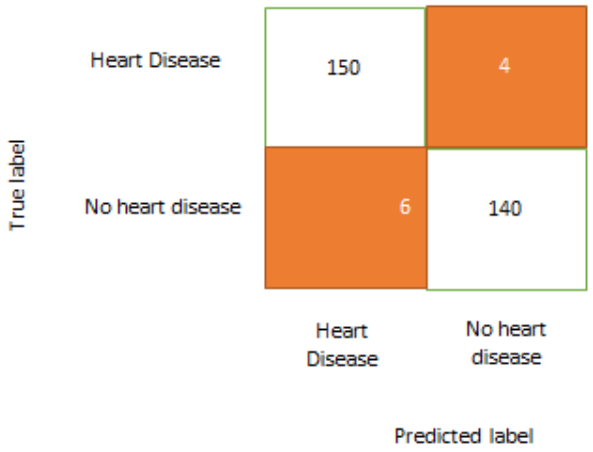


Figure 4 Confusion matrix

(a) Correlation Results

Patterson link results demonstrate how feature qualities influence target attributes. According to the correlation matrices, despite the primary principal components with patient attributes, no single statistic has a significant impact on stroke. Age, Gender, Chest pain type, Blood pressure, Fat, Glucose Level, ECG result, Heart Rate, Exercise Angina, Old Peak, ST-slope, and heart disease all play a role in heart disease.

Table 3 Correlation matrices among EMR datasets

Gender	1.8										
Chest Pain type	0.1	1.8									
Blood pressure	0.3	0.4	1.8								
Fat	0.1	0.2	0.4	1.8							
Glucose Level	0.1	0.2	0.5	0.4	1.8						
ECG result	0.2	0.1	0.4	0.3	0.3	1.8					
Heart rate	0.5	0.2	0.2	0.5	0.3	0.4	1.8				
Exercise Angina	0.5	0.1	0.3	0.2	0.1	0.1	0.2	1.8			
Old Peak	0.1	0.2	0.4	0.2	0.6	0.3	0.3	0.2	1.8		
ST-slope	0.2	0.6	0.1	0.7	0.4	0.2	0.2	0.3	0.2	1.8	
Heart Disease	0.2	0.1	0.3	0.4	0.3	0.2	0.4	0.2	0.4	0.5	1.8

The correlation matrix in Table 3 shows that none of the traits are significantly related to one another. As a result, each trait may contribute differently to HFD prediction. The two subsections that follow look into the importance of a specific trait in predicting HFDs.

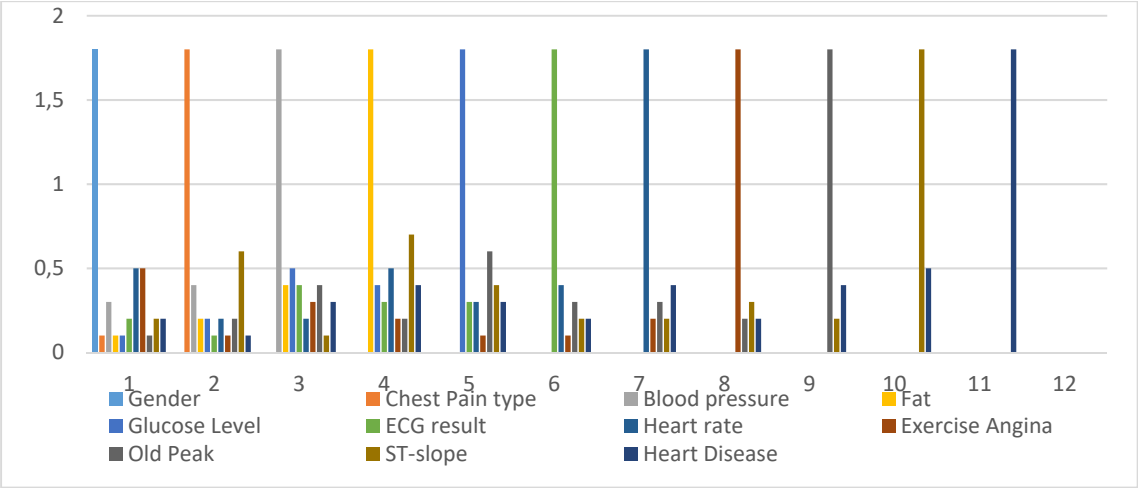


Figure 4. Importance of patient attributes in predicting the occurrence of stroke using EHR datasets

The scattered variance in Fig. 4 demonstrates that the various basic components explain various underlying phenomena. These phenomena can be investigated using variable loadings. Variation loadings are the contributions of multiple variables to a single major component.

5.3. Identification Performance of EfficientNet B0

The accuracy and loss values of the proposed EfficientNetB0 for training and testing datasets in 65 epochs are shown in Figures 5 and 6. Due to the effect of the transfer learning method, the accuracy of EfficientNet B0 exceeds 97% after the first epoch on the testing dataset. On the testing dataset, the highest accuracy of 98.122% is obtained.

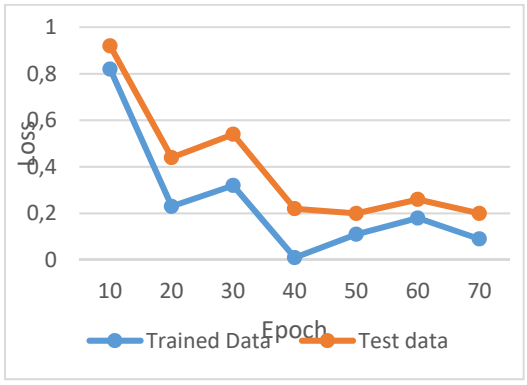


Figure 5 Loss Epoch

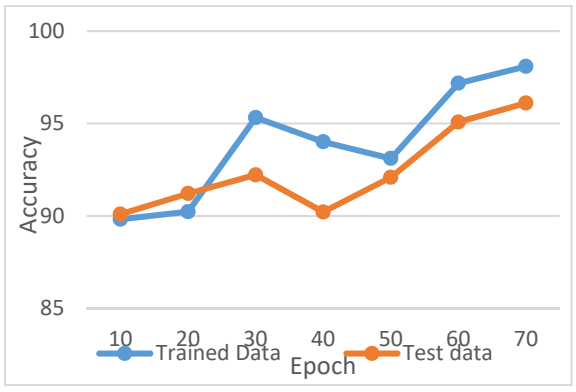


Figure 6 Accuracy Epoch

5.4 ROC & AUC curves

Figures 7 and 8 show the ROC curves comparing the performance of EfficientNet-B0. Plotting the true positive rate on the Y-axis and the false positive rate on the X-axis yields these curves. Meanwhile, by varying the cut-off value, these curves analyse the model's score. Identification outcomes perform better when the AUC (area under the ROC) is high. It is worth noting that the AUC for all categories is greater than 0.98.

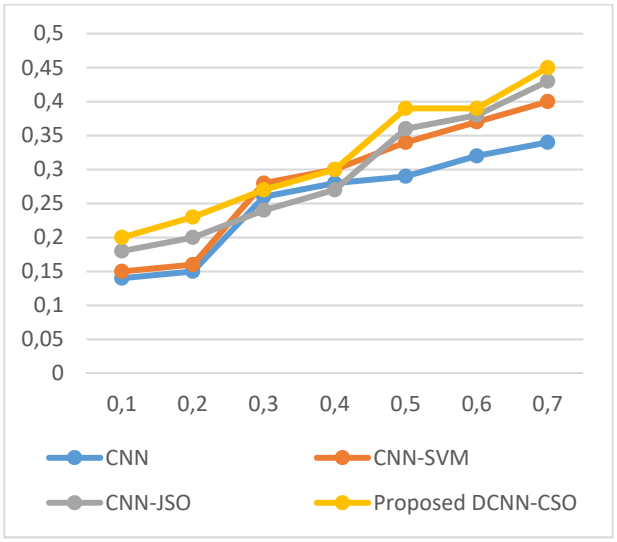


Figure 7 ROC curve

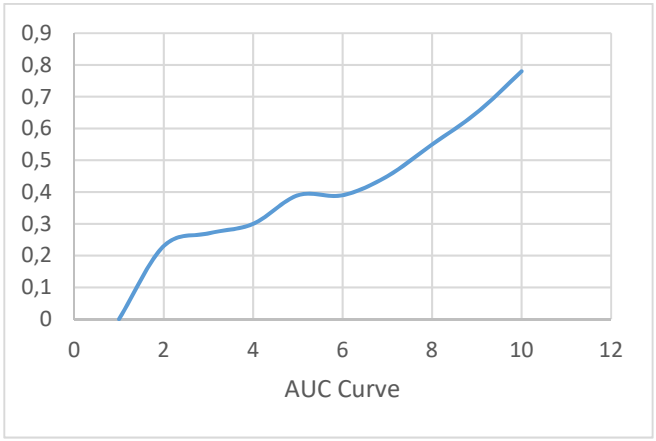


Figure 8 AUC curve

5.5 Comparision Of proposed DCNN-EfficientNet B0 model

When compared to four classical DCNN models, the proposed EfficientNet B0 performed the best. The table compares EfficientNet B0's average precision, recall, f1-score, and AUC to four existing models: VGG-19, ResNet-152, DenseNet-201, and Inception ResNet-V2. It is clear that EfficientNet B0 outperforms four DCNN models.

Table 3 Identification performance comparison among different DCNN models

Model	Precision	Recall	F1-score	AUC
VGG-19	0.68	0.76	0.66	0.88
ResNet-152	0.76	0.75	0.72	0.87
DenseNet-201	0.56	0.78	0.78	0.89
Inception ResNet-V2	0.77	0.87	0.85	0.96
EfficientNet-B0	0.89	0.90	0.91	0.98

Table 3 employs five classifier models to predict HFP performance (Fig 9). The DCNN classifier model is used to analyse the results and assess the accuracy, precision, recollection, f-1 rating, and TN frequency, which is the highest of the other classifiers at 98.45%.

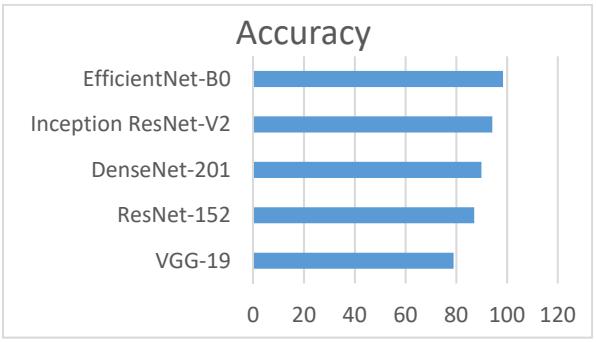


Figure 9 Accuracy

5.6 Performance Comparison

However, while the proposed model's studies have an accuracy of more than 98.45%, this model can only identify four classes of HFPs in the DCNN classifier. Although the models proposed in can identify more than four classes of DCNN, their accuracy is lower than the model proposed in this paper. While the references in Table 4 used different datasets, the EfficientNet Bo model, as developed in this paper, has more DCNN categories and achieves more competitive results.

Table. 4. Comparison with existing studies for DCNN-Efficient Net B0 models

SI No	Author	Models	Accuracy
1	Bi, C.; Wang, et.al	CNN using MobileNet	73.50%
2	Yan, Q. et.al	CNN using VGG	95.01%
3	Wang, L. et.al	CNN using ResNet	94.99%
4	Pradhan, P. et.al	DenseNet-201	96.76%
5	Proposed	DCNN using EfficientNet-B0	98.45%

4. CONCLUSION

Prediction of heart diseases at an earlier stage may save lives from heart attacks. A good classification algorithm can assist a physician in predicting the presence of cardiovascular disease before it occurs. This study focuses on predicting a possible heart disease using a kaggle-available EMRs dataset. This dataset includes both cardiac test parameters and general human habits. DCNNs, on the other hand, have outperformed heart disease prediction in recent studies. Electronic health records (EHRs) have recently been used for diagnosing several diseases and show the potential for HFDs diagnosis as well. However, the lack of an appropriate feature set limits the accuracy and efficacy of EHR-based HFDs diagnosis. We trained EfficientNets B0 on the DCNN using transfer-learning on ImageNet pre-trained weights and fine-tuning the Convolutional Neural Networks. We evaluated the performance of all EfficientNet B0 variants on this imbalanced multiclass classification problem using measures such as Precision, Recall, Accuracy, F1 Score, and Confusion Matrices to examine the impact of transfer learning and fine-tuning. When compared to existing models, the proposed model, the EfficientNet B0, achieved the highest accuracy of 98.45%.

REFERENCES

[1] Ahmed, H., Younis, E. M., Hendawi, A., & Ali, A. A. (2020). Heart disease identification from patients' social posts, machine learning solution on spark. Future Generation Computer Systems, 111, 714–722.

[2] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. Artificial Intelligence Review, 53, 5455–5516.

[3] Lei, X., Pan, H., & Huang, X. (2019). A dilated cnn model for image classification. IEEE Access, 7, 124087–124095.

[4] Hussain, T., Muhammad, K., Ullah, A., Cao, Z., Baik, S. W., & de Albuquerque, V. H. C. (2019). Cloud-assisted multiview video summarization using cnn and bidirectional lstm. IEEE Transactions on Industrial Informatics, 16, 77–86.

[5] Widiastuti, N. (2019). Convolution neural network for text mining and natural language processing. In IOP Conference Series: Materials Science and Enineering (Vol. 662, p. 052010). IOP Publishing.

[6] Baozhou, Z, Hofstee, P., Lee, J., & Al-Ars, Z. (2020). Sofar: Shortcut-based fractal architectures for binary convolutional neural networks. arXiv preprint arXiv:2009.05317.

[7] Eren, L., Ince, T., & Kiranyaz, S. (2019). A generic intelligent bearing fault diagnosis system using compact adaptive 1d cnn classifier. Journal of Signal Processing Systems, 91, 179–189.

[8] Nagarajan SM, Deverajan GG, Chatterjee P, Alnumay W, Ghosh U (2021) Effective task scheduling algorithm with deep learning for internet of health things (ioht) in sustainable smart cities. Sustain Cities Soc 71:102945

[9] Nagarajan SM, Muthukumaran V, Murugesan R, Joseph RB, Munirathanam M (2021) Feature selection

model for healthcare analysis and classification using classifier ensemble technique. Int J Syst Assur Eng Manag

[10] Paragliola G, Coronato A (2021) An hybrid ECG-based deep network for the early identification of high-risk to major cardiovascular events for hypertension patients. J Biomed Inform 113:103648

[11] Ghwanmeh, S.; Mohammad, A.; Al-Ibrahim, A. Innovative artificial neural networks-based decision support system for heart diseases diagnosis. J. Intell. Learn. Syst. Appl. 2020, 5, 35396.

[12] Wang, Z., Zhu, Y., Li, D., Yin, Y., & Zhang, J. (2020). Feature rearrangement based deep learning system for predicting heart failure mortality. Computer Methods and Programs in Biomedicine, 191, 105383.

[13] Reddy, K.V.V.; Elamvazuthi, I.; Aziz, A.A.; Paramasivam, S.; Chua, H.N.; Pranavanand, S. Heart Disease Risk Prediction Using Machine Learning Classifiers with Attribute Evaluators. Appl. Sci. 2021, 11, 8352.

[14] Perumal, R. Early prediction of coronary heart disease from cleveland dataset using machine learning techniques. Int. J. Adv. Sci. Technol. 2020, 29, 4225–4234.

[15] Amin, M.S.; Chiam, Y.K.; Varathan, K.D. Identification of significant features and data mining techniques in predicting heart disease. Telemat. Inform. 2019, 36, 82–93.

[16] Mohan, S.; Thirumalai, C.; Srivastava, G. Effective heart disease prediction using hybrid machine learning techniques. IEEE Access 2019, 7, 81542–81554.

[17] Dwivedi, A.K. Performance evaluation of different machine learning techniques for prediction of heart disease. Neural Comput. Appl. 2018, 29, 685–693.

[18] Ashraf, M.; Ahmad, S.M.; Ganai, N.A.; Shah, R.A.; Zaman, M.; Khan, S.A.; Shah, A.A. Prediction of Cardiovascular Disease Through Cutting-Edge Deep Learning Technologies: An Empirical Study Based on TENSORFLOW, PYTORCH and KERAS. In International Conference on Innovative Computing and Communications; Springer: Berlin/Heidelberg, Germany, 2021; pp. 239–255.

[19] Radhimeenakshi, S., 2016, March. Classification and prediction of heart disease risk using data mining techniques of Support Vector Machine and Artificial Neural Network. In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 3107-3111). IEEE.

[20] Shen, Z., Clarke, M., Jones, R.W. and Alberti, T., 2019, October. Detecting the risk factors of coronary heart disease by use of neural networks. In Proceedings of the 15th Annual International Conference of the IEEE Engineering in Medicine and Biology Societ (pp. 277-278). IEEE.

[21] Dewan, A. and Sharma, M., 2015, March. Prediction of heart disease using a hybrid technique in data mining classification. In 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 704-706). IEEE.

[22] Chen, A.H., Huang, S.Y., Hong, P.S., Cheng, C.H. and Lin, E.J., 2011, September. HDPS: Heart disease prediction system. In 2011 computing in cardiology (pp. 557-560). IEEE.

[23] Sonawane, J.S. and Patil, D.R., 2014, March. Prediction of heart disease using learning vector quantization algorithm. In 2014 Conference on IT in Business, Industry, and Government (CSIBIG) (pp. 1-5). IEEE.

[24] Feshki, M.G. and Shijani, O.S., 2016, April. Improving the heart disease diagnosis by evolutionary algorithm of PSO and Feed Forward Neural Network. In 2016 Artificial Intelligence and Robotics (IRANOPEN) (pp. 48-53). IEEE.

[25] Hannan, S.A., Mane, A.V., Manza, R.R. and Ramteke, R.J., 2010, December. Prediction of heart disease medical prescription using radial basis function. In 2010 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-6). IEEE.

[26] Radhimeenakshi, S., 2016, March. Classification and prediction of heart disease risk using data mining techniques of Support Vector Machine and Artificial Neural Network. In 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 3107-3111). IEEE.

[27] Shen, Z., Clarke, M., Jones, R.W. and Alberti, T., 1993, October. Detecting the risk factors of coronary heart disease by use of neural networks. In Proceedings of the 15th Annual International Conference of the IEEE Engineering in Medicine and Biology Societ (pp. 277-278). IEEE.

[28] Dewan, A. and Sharma, M., 2015, March. Prediction of heart disease using a hybrid technique in data mining classification. In 2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 704-706). IEEE.

[29] Chen, A.H., Huang, S.Y., Hong, P.S., Cheng, C.H. and Lin, E.J., 2011, September. HDPS: Heart disease prediction system. In 2011 computing in cardiology (pp. 557-560). IEEE.

[30] Sonawane, J.S. and Patil, D.R., 2014, March. Prediction of heart disease using learning vector quantization algorithm. In 2014 Conference on IT in Business, Industry, and Government (CSIBIG) (pp. 1-5). IEEE.

[31] Feshki, M.G. and Shijani, O.S., 2016, April. Improving the heart disease diagnosis by evolutionary algorithm of PSO and Feed Forward Neural Network. In 2016 Artificial Intelligence and Robotics (IRANOPEN) (pp. 48-53). IEEE.

[32] Hannan, S.A., Mane, A.V., Manza, R.R. and Ramteke, R.J., 2010, December. Prediction of heart disease medical prescription using radial basis function. In 2010 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-6). IEEE.