ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2022.0380

OPTIMAL ENSEMBLE FEATURE SELECTION (OEFS) METHOD AND KERNEL WEIGHT CONVOLUTIONAL NEURAL NETWORK (KWCNN) CLASSIFIER FOR MEDICAL DATASETS

C. Sathish Kumar and P. Thangaraju
Department of Computer Science, Bishop Heber College, India

Abstract

Disease detection software that works automatically in healthcare domain refers to the proactive or reactive use of computerised data systems for diagnosis of diseases. Medical knowledge base, data processing, and data analytics are the three major components of the system. The procedures of data processing and data analytics are crucial. Data mining (DM) techniques were used to process these processes. DM is a tool for finding patterns in massive amounts of data and retrieving knowledge. Clinical and diagnostic evidence has created a slew of reliable timely detection services and other health-related technology in the DM and healthcare industries. Artificial Intelligence (AI) in Machine Learning (ML) includes classification and predictive analytics. Identifying key characteristics and developing a classification model to determine whether the cases are disease or not is a difficult task. Feature selection (FS) refers to the process of reducing the quantity of input features when developing a predictive model. Reducing the number of input features is desirable because it cuts the computational cost of modelling while also improving the model's performance in some cases. Instead of using a single feature selection, Optimal Ensemble Feature Selection (OEFS) solves a feature selection problem by integrating numerous feature selections. The OEFS method works by integrating the outputs of different single feature selection models like Divergence Weight Elephant Herding Optimization (DWEHO), Divergence Weight ButterFly Optimization Algorithm (DWBFO), and Differential Evolution (DE). By merging different subsets of features, Weighted Majority Voting (WMV) is used in finding the optimal feature subset. Classification model using Kernel Weight Convolutional Neural Network (KWCNN) classification is proposed. The convolution operation is a mathematical linear action across matrices that gives it its name. In terms of medical disease diagnosis, the proposed KWCNN classification performs quite well. To determine the performance of all classification algorithms, evaluation criteria such as sensitivity, specificity, f-measure, and accuracy were measured using a confusion matrix.

Keywords:

Medical Diseases, healthcare, Databases, Data Mining (DM), Artificial Intelligence (AI), Machine Learning (ML), Optimal Ensemble Feature Selection (OEFS), Divergence Weight Elephant Herding Optimization (DWEHO), Divergence Weight ButterFly Optimization Algorithm (DWBFO), Differential Evolution (DE), and Kernel Weight Convolutional Neural Network (KWCNN)

1. INTRODUCTION

A human being might suffer from various diseases in this world. Diseases can affect people not only physically, but also psychologically. Diseases develop mostly due to four factors: infection, deficiency, genetics, and body organ dysfunction. Doctors or medical professionals are in charge of detecting and diagnosing suitable disease, as well as providing medical therapies or treatments to cure or control it. After treatment, some disorders can be healed. In healthcare, Predictive Data Mining (PDM) is extremely important. The purpose of PDM in healthcare industry is to create models using electronic health records which apply patient-specific data to anticipate the expected result and assist physicians in taking decisions. Models for prognosis, diagnosis, and treatment planning can all be built using PDM. The symptoms that a patient exhibits, and also the results of clinical examinations and laboratory tests, may be indicative of more than one disease. Because clinical data offered by patients is imprecise, making a decision with total certainty is not practicable, and making an appropriate conclusion is a difficult undertaking. PDM approaches can be used to infer clinical recommendations for patients from data in electronic health records, with the help of historical data on clinical judgments made for patients with comparable symptoms. Clinicians can employ machine-learning-based Computer-Aided Diagnosis (CAD) systems as a secondary choice in decision-making and treatment planning.

Because of increased computer power and the accessibility of datasets from open-source sources, machine learning has grown in popularity as technology has progressed. Machine learning is used in healthcare in a number of ways. Images, patient data, and other sorts of information generated by the healthcare business can be utilised to identify trends and make predictions. Machine learning is utilised in healthcare to solve a number of problems [1]-[3]. As a result, building a machine learning model, training it on a dataset, and including specific patient data can help predict outcomes. The forecast outcome will be unique to that person because it will be determined by the information provided.

Kidney disease, heart disease and hepatitis are just a few of the diseases that are limiting people's lifestyles. Type-2 diabetes is a disease that can be avoided by maintaining a healthy weight, lifestyle, and other factors [4]. Chronic Kidney Disease (CKD) is a kidney disease defined by the progressive loss of kidney function over time [6] [7]. Hepatitis is a chief chronic liver disease that affects people all over the world. The liver is one of the largest and heaviest organs in the human body [5]. As a result, developing a smart diagnostic system for predicting disease is critical. To perform feature processing, data mining models like Principal Component Analysis (PCA) and Fisher Discriminant Analysis (FDA) are combined with machine learning models like Decision Trees (DTs), Logistic Regression (LR), Support Vector Machine (SVM), Naive Bayes (NBs), Neural Network (NN) models, ensembles of neural networks, and deep neural networks to create diagnosis models' [8]-[11].

The initial stage will be to decide upon the best classifier. Choosing the right classifier to utilise for the most accurate results, on the other hand, is a difficult task. However, not all characteristics are equally important in identifying a disease or its stage of progression. As a result, selecting the best collection of features to recognise disease is a key issue. Connection rejection, which decreases the unpredictability of calculations, as well as boosts the classifier's findings, are all advantages of using trivial features. As a result, feature selection is an extensively applied data preprocessing approach in data mining, and it is primarily utilised to minimize data by discarding unrelated and redundant attributes from any dataset [12]. Furthermore, this technique improves data comprehension, promotes better data visualisation, saves learning algorithm training time, and improves prediction performance. In the healthcare industry, there are several uses of relevant feature identification approaches. Variable selection approaches include filter methods, wrapper methods, ensemble methods, and embedding methods, to name a few. Filter techniques may fail to choose the most "useful" features because they disregard interaction among classifiers and the dependency of one feature on another. The wrapper technique has the drawback of having to be re-executed if another learning algorithm is required. Furthermore, with small training datasets, this approach is quite complex and susceptible to over-fitting. Embedded methods make judgments based on the classification algorithm. As a result, the classifier's hypothesis might influence feature selection, which may or may not work with another classifier. The majority of authors have been working on ensemble techniques for feature selection in recent times [13,14]. Ensemble-based feature selection approaches combine many feature subsets to discover an ideal subset of features using a mix of feature ranking to improve classifier accuracy.

A set of various feature selectors is chosen in the initial phase of the ensemble technique, and each selector offers a sorted order of features. The second phase uses several aggregation strategies to combine the specified subgroups of features [15]. For feature selection and decision-making, ensemble approaches, computer algorithms influenced by biological processes, and evolution can deliver improved results [16]. If the data has more significant and non-redundant features, various computer techniques function more effectively and produce more accurate results. Optimal Ensemble Feature Selection (OEFS) is carried out in this study by integrating models such like Divergence Weight Elephant Herding Optimization (DWEHO), Divergence Weight ButterFly Optimization Algorithm (DWBFO), and Differential Evolution (DE) to achieve better outcomes. For each dataset, an OEFS is used to select the most important feature set. By combining multiple feature subsets, Weighted Majority Voting (WMV) is applied for selecting an ideal subset of features. For the classification of medical disease diagnosis, the Kernel Weight Convolutional Neural Network (KWCNN) classification is introduced. In terms of medical disease diagnosis, the proposed KWCNN classification performs admirably.

2. LITERATURE REVIEW

To obtain highly accurate prediction results, Ghosh et al [17] used four dependable techniques, including Support Vector Machine (SVM), AdaBoost (AB), Linear Discriminant Analysis (LDA), and Gradient Boosting (GB). These algorithms are tested using the UCI machine learning repository's online dataset. Later on, several performance evaluation measures were shown to demonstrate proper outcomes. Finally, these benchmarks can be used to identify the most efficient and optimized algorithms for the intended project.

Sartakhti et al [18] presented a new machine learning algorithm which combines SVM and Simulated Annealing (SA). Simulated annealing is a stochastic method for solving tough optimization problems that is currently in widespread use. The SVM has been extensively researched and has been effectively proven as a prediction tool in recent years because to its various distinct advantages. The data for this dataset was gathered from the UCI machine learning database. Using 10-fold cross validation, the classification accuracy is determined. The method's classification accuracy is 96.25 percent, which looks more promising when compared with other classification approaches in the literature for this topic.

Pearson Correlation, Recursive Features Elimination (RFE), and Lasso Regularization are three features-based methods suggested by Yadav and Pal [19]. Different feature selection approaches are applied to the data table in order to improve prediction. The first experiment used Pearson Correlation on M5P, Random Tree (RT), Reduced Error Pruning (REP), and Random Forest (RF) ensemble methods. The second experiment used Pearson Correlation on M5P, Random Tree (RT), Reduced Error Pruning (REP), and Random Forest (RF) ensemble methods. The above four tree-based algorithms are subjected to RFE in the second experiment. Lasso Regularization is used as an alternative to tree-based techniques in the third experiment. After that, the performance was evaluated and classification accuracy, precision, and sensitivity were calculated. When compared to other algorithms in the prior methodologies, the RF ensemble method predicted better results.

To test the validity and importance of FS in the Alizadeh Sani CHD dataset, Qin et al [20] proposed a numerous assessment criterion to quantify features, along with a heuristic search method and seven typical classification algorithms. On this foundation, an unique algorithm based on Different Feature Selection that integrates multiple FS approaches into the Ensemble Algorithm was developed (EA-MFS). To increase data diversity, a bagging approach is used. For functional perturbation, the aforementioned MFS methods are used, a major voting technique is presented to bring out the decision results, and selective integration is performed for understanding the difference of base classifiers in the ensemble process. In comparison to a single FS approach, the EA-MFS algorithm might more thoroughly characterise feature relationships, improve classification impact, and be more resilient.

Aim et al [21] advocated that key features and data mining approaches be identified to increase the accuracy of cardiovascular disease prediction. Different combinations of characteristics and seven classification algorithms were applied to create prediction models: k-Nearest Neighbor (kNN), Decision Tree (DT), Naive Bayes (NBs), Logistic Regression (LR), SVM, Neural Network, and Vote (a hybrid technique with NBs and LR). As per the results of the research, the heart disease prediction model constructed by combining important selected features and the best-performing data mining approach (i.e. Vote) obtains an accuracy of 87.4% in predicting heart disease.

Using the advantages of ensemble learning, Nilashi et al [22] proposed a precise technique for hepatitis disease diagnosis. Adaptive Neuro-Fuzzy Inference System (ANFIS) ensembles for hepatitis disease prediction, Non-linear Iterative Partial Least Squares (NIPLS) for data dimensionality reduction, Self-Organizing Map (SOM) for clustering task, and Non-linear Iterative Partial Least Squares (NIPLS) for data dimensionality reduction. For the identification of the most important features in the experimental dataset, Decision Trees (DTs) are used. The proposed method is put to the test on a real-world dataset, and the outcomes are compared to the most recent findings from previous studies.

Christo et al. [23] introduced bioinspired feature selection techniques and a gradient descendent backpropagation neural network for classification using bioinspired algorithms. Data preprocessing, feature selection, and classification are all applied to the clinical data. Missing values were handled by hot deck imputation, and data transformation was done with min-max normalization. Wrapper technique uses bioinspired algorithms such as Differential Evolution (DE), Lion Optimization (LO), and Glowworm Swarm Optimization (GWO), with AdaBoostSVM classifier accuracy as fitness function. Each bioinspired algorithm chooses one of three feature subsets. The best features from the three feature subsets are chosen using Correlation-based Ensemble Feature Selection (CBEFS). A gradient descendent backpropagation neural network is trained using the optimum features chosen through correlation-based ensemble feature selection. To train and test the classification performance, a ten-fold cross-validation technique was used. The classification accuracy was assessed using the Hepatitis dataset and the Wisconsin Diagnostic Breast Cancer (WDBC) dataset from the UCI Machine Learning repository. The Wisconsin Diagnostic Breast Cancer dataset has a precision of 98.47%, whereas the Hepatitis dataset has a precision of 95.51%. The proposed architecture can be customized to create clinical decision-making systems for any health condition, assisting physicians with medical diagnosis.

Jongbo et al [24] combined a bagging ensemble technique using efficient feature selection technique to produce a consistent and accurate predictive model suitable for rightly identifying diseased from non-diseased patients on a CKD dataset. The study used a real patient dataset of 400 instances with 24 conditional attributes and a decisional class that was derived from the UCI machine learning repository. To choose the optimal collection of characteristics for the prediction model, the Random Forest (RF) technique was utilised.

3. PROPOSED METHODOLOGY

In high-dimensional biomedical datasets, selecting relevant and deleting redundant features is critical for improving the performance of machine-learning algorithms for enhancing detection prediction accuracy and minimising algorithm construction time. It is also planned to use a deep learning classifier to improve the efficacy of the proposed method. In this work, Optimal Ensemble Feature Selection (OEFS) is proposed based on combining multiple feature selection models such as Divergence Weight Elephant Herding Optimization (DWEHO), Divergence Weight ButterFly Optimization Algorithm (DWBFO) and Differential Evolution (DE) will obtain better results. Weighted Majority Voting (WMV) is used to combine various feature subsets for finding the best feature subset. Kernel Weight Convolutional Neural Network (KWCNN) classification is introduced for classifying multiple diseases. Proposed method was implemented in an efficient manner, which further enhanced the classifier performance in relation to efficiency and accuracy. The overall framework of the research work is depicted in Fig.1.

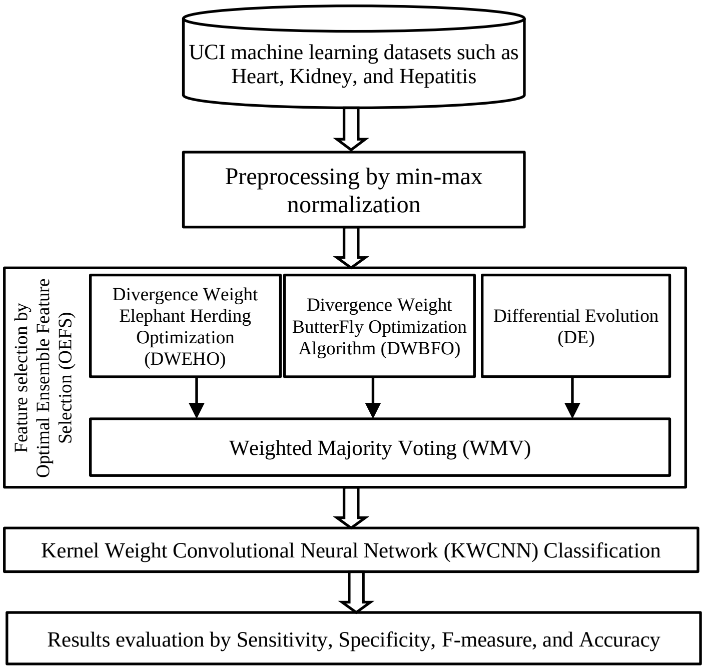
UCI machine learning datasets such as Heart, Kidney, and Hepatitis
Preprocessing by min-max normalization
Feature selection by Optimal Ensemble Feature Selection (OEFS)
Divergence Weight Elephant Herding Optimization (DWEHO)
Divergence Weight ButterFly Optimization Algorithm (DWBFO)
Differential Evolution (DE)
Weighted Majority Voting (WMV)
Kernel Weight Convolutional Neural Network (KWCNN) Classification
Results evaluation by Sensitivity, Specificity, F-measure, and Accuracy

Fig.1. Proposed OEFS model and KWCNN classification for medical diseases diagnosis

3.1 PRE-PROCESSING

The min-max normalization approach is used to pre-process the datasets. When features are on a different scale, normalization is typically used to preserve the proportion of significance among them. When dealing with attributes on multiple scales, normalization is usually essential; otherwise, the effectiveness of a substantial, similarly essential attribute (on a smaller scale) may be weakened because of values of another attribute on a larger scale. The min-max normalization approach is used to pre-process datasets having a wide range of properties. One of the most prevalent methods of data normalization is min-max normalization. The minimum value of each feature is converted to a 0, the highest value is converted to a 1, and all other values are converted to a decimal between 0 and 1. In this process, all of the values are converted to a single scale between 0 and 1, emphasising the significance of the attribute despite its low value range on the scale. The original data goes through a linear transformation in this data normalisation procedure. The minimum and maximum values from the data are retrieved, and each value is replaced by the Eq.(1) [25] [26].

$$v' = \frac{v-(A)}{(A)-(A)}\left(new_{\max}(A) - new_{\min}(A)\right) + new_{\min}(A) \qquad (1)$$

From $new_{min}(A)$ to $new_{max}(A)$, transform the data from measured units to a new interval. Where A is the attribute data and min(A) and max(A) are the minimum and maximum absolute value of A respectively. Each data entry's new value is denoted by v'. The old value of each data entry is v. $new_{max}(A)$, $new_{min}(A)$ are the maximum and minimum value of the range respectively (i.e. the required boundary values of the range) [27].

3.2 FEATURE SELECTION BY OPTIMAL ENSEMBLE FEATURE SELECTION (OEFS)

Optimal Ensemble Feature Selection (OEFS) approach is introduced which combines the subsets retrieved from different methods using the feature-class. The method combines the feature subsets chosen by different feature selection methods such as Divergence Weight Elephant Herding Optimization (DWEHO), Divergence Weight ButterFly Optimization Algorithm (DWBFO) and Differential Evolution (DE) using Weighted Majority Voting (WMV). If all selectors choose a common feature for a certain rank, that feature is chosen without utilising the WMV approach and included in the optimal subset.

3.3 DIVERGENCE WEIGHT ELEPHANT HERDING OPTIMIZATION (DWEHO)

Elephant Herding Optimization (EHO) is a heuristic intelligence algorithm inspired by elephants' nomadic habits. The elephant herd largely has the following two features, based on observation and study of elephants. The first distinguishing feature of the elephant herd is that it is divided into several clans, each with its own patriarch and individuals who follow the patriarch's orders. Another distinguishing feature of the herd is the absence of an adult male elephant. When young elephants reach adulthood, they will be separated from the other elephants. The main idea of EHO is inspired by these two features, and it is divided into two parts: clan update and separating [28]. The elephant herd's first feature can be abstracted as a clan updating operator, and the update method is given using Eq.(2),

$$x_{n,i,j}=x_{i,j}+r*a*(x_{b,i}-x_{i,j})*F_W \tag{2}$$

where $x_{i,j}$ and $x_{n,i,j}$, are the old and new feature positions of elephant j in clan i respectively; α∈[0, 1] denotes the scaling factor and $x_{b,i}$ denotes the feature position in clan i with the best fitness value. r is a random number in the range [0,1] with a normal distribution. Most individuals (features) in Eq.(2) have been updated, but the matriarch in each clan is not updated. Let us understand that when a specific feature value is noticed, it provides a given quantity of information for the target feature in order to compute the weight value of each feature in the DWEHO. The difference between the target feature's prior and posterior distributions defines the amount of information contained in a given feature value. Eq.(3) calculates the Kullback-Leibler (KL) measure of divergence, which is used to determine the range of a feature value fv_{ij}.

$$KL\left(fv_{ij}\right)=\sum_{c}P\left(fv_{ij}\right)\log\log\left(\frac{P\left(fv_{ij}\right)}{P(c)}\right) \tag{3}$$

where fv_{ij} specifies the j value of the i^{th} feature in the training samples. The weighted average of the KL measurements across the feature values is known as the feature weight. As a result, the weight of feature i, denoted as $F_{Wavg}(i)$ by Eq.(4),

$$F_{Wavg}(i)=\sum_{c}P\left(fv_{ij}\right)KL\left(fv_{ij}\right) \tag{4}$$

In this Eq.(4), $P(fv_{ij})$ indicates the probability that the feature i has the value of fv_{ij}. The weight $F_{Wavg}(i)$ is biased towards feature with many values, therefore, the number of records related with each feature value is too small to perform any effective learning. The weight of feature i in its final form, given as $F_W(i)$ is defined by Eq.(5),

$$F_W(i)=\frac{\sum_{j|i}P\left(fv_{ij}\right)\sum_{j|i}P\left(fv_{ij}\right)\log\log\left(\frac{P\left(fv_{ij}\right)}{P(c)}\right)}{-Z\sum_{j|i}P\left(fv_{ij}\right)\log\log\left(P\left(fv_{ij}\right)\right)} \tag{5}$$

where Z specifies the normalization constant which is computed by Eq.(21),

$$Z=\frac{1}{n}\sum_{i}F_W(i) \tag{6}$$

In this Eq.(6), the number of selected features in the training data is represented by n. In this work, the normalized version of $F_W(i)$ (Eq.(6)) is provided to ensure that $\sum_{i}F_W(i)=n$.

Finally, this weight value is updated to DWEHO algorithm. Based on the weight, importance of features is selected. Therefore, the update process of the matriarch for feature selection process is shown in Eq.(7)-Eq.(8).

$$x_{n,i,j}=\beta*x_{c,i} \tag{7}$$

$$x_{c,i}=\frac{1}{n_i}\sum_{j=1}^{n_i}x_{i,j} \tag{8}$$

where β is a scale factor between 0 and 1. In clan i, the centre position (feature position) is $x_{c,i}$ which may be computed using Eq.(8). Clan i has the elephant number n_i. In Eq.(3), the update of the matriarch position (feature position) is associated with the information of all members (features) in the clan. From the elephant herd's second feature, the separating operator may be abstracted. The Eq.(9) depicts the separation procedure,

$$x_{w,i}=x_{min}+r*(x_{max}-x_{min}) \tag{9}$$

where $x_{w,i}$ denotes the position (feature position) with worst fitness value (classification accuracy) in clan i; x_{max} and x_{min} are the upper and lower bounds of the elephant's position (feature position), respectively and r is a random number with a normal distribution between 0 and 1.

Algorithm 1: DWEHO Algorithm

```
Step 1:  Initialization of number of populations via the number of features and parameters
Step 2:  Fitness evaluation via classification accuracy and their feature position
Step 3:  While $t<T_{max}$ do
Step 4:  For $i$=1 to $n_c$ do
Step 5:  For $j$=1 to $n_j$ (the number of elephants (Features) in one clan) do
Step 6:  Update $x_{i,j}$ and generate $x_{n,i,j}$ based on the Eq.(2), generate feature weight $F_W(i)$ by Eq.(3)-Eq.(5)
Step 7:  If $x_{i,j}=x_{b,i}$ then
Step 8:  Update $x_{i,j}$ and generate $x_{n,i,j}$ based on the Eq.(7)- Eq.(8)
Step 9:  End if
Step 10: End for
Step 11: For $i$=1 to $n_c$ do
```

Step 12: Replace the worst elephant in clan *i* by Eq.(9)

Step 13: End for

Step 14: Evaluate individuals (features) according to their new position

Step 15: End while

Algorithm 1 shows the working procedure of proposed DWEHO algorithm. It starts with initialization of population via the number of features in the dataset and then evaluate the fitness value based on that eliminate worst features in the clan, then start the procedure with *t* iteration to T_{max}. For each features two operations such as clan updating, and the other is separating is performed by step 6 to step 8. Once these operations are performed, then remove the worst elephant from the clan via the step 11 to step 13. Finally find the best features in the step 14. Similarly, the flowchart for the proposed model is illustrated in Fig.2.

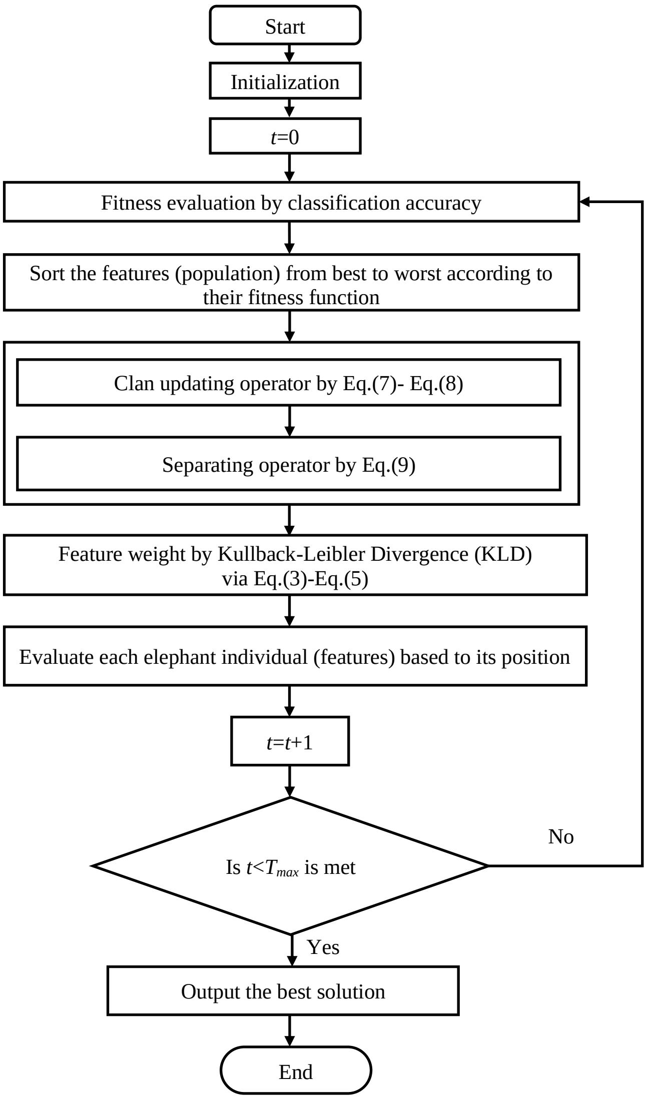
Start
Initialization
t=0
Fitness evaluation by classification accuracy
Sort the features (population) from best to worst according to their fitness function
Clan updating operator by Eq.(7)- Eq.(8)
Separating operator by Eq.(9)
Feature weight by Kullback-Leibler Divergence (KLD) via Eq.(3)-Eq.(5)
Evaluate each elephant individual (features) based to its position
t=t+1
Is t<T_max is met
No
Yes
Output the best solution
End

Fig.2. Flowchart of DWEHO algorithm

3.4 DIVERGENCE WEIGHT BUTTERFLY OPTIMIZATION (DWBFO) ALGORITHM

In this work, feature selection is done using Divergence Weight ButterFly Optimization Algorithm (DWBFO) to select the optimal features from the given dataset. DWBFO is new nature-inspired algorithm that mimics food search (higher accuracy with selected features) and the mating behaviour of the butterflies for solving classification issues in disease diagnosis. The proposed DWBFO Algorithm is mainly centred on the foraging strategy of butterflies, that use their sense of smell for optimal selection of features for determining the location of nectar partner [29] [30]. On the basis of scientific interpretations, it is discovered that butterflies possess a precise sense of finding the location of the source of fragrance.

A butterfly's fitness (classification accuracy) is associated with the strength of its fragrance, i.e., as a butterfly moves from one location to another, its fitness varies. The entire concept of sensing and processing the modality in the DWBFO Algorithm is built on three key terms: sensory modality (*c*), stimulus intensity (I_n) and power exponent (*a*) for optimal selection of features [30]. In DWBFO Algorithm, I_n is correlated with the fitness (accuracy) for the selection of features. Based on these concepts, in DWBFO Algorithm, as a function of the physical intensity of stimulus the fragrance is formulated by Eq.(10),

$$b_f = cIn^a \tag{10}$$

where b_f is the perceived magnitude of the fragrance, i.e. how strong the fragrance is observed by other butterflies, *c* is the sensory modality that is determined by classification accuracy, In is the stimulus intensity and a is the power exponent that depends on modality. As a result, *a* and *c* are in the range [0,1]. If *a* = 0, on the other hand, the fragrance released by any butterfly cannot be detected by other butterflies at all. As such, the parameter a determines the algorithm's behaviour. Yet another significant parameter is *c* which is used to determine the speed of convergence and how the DWBFO algorithm works. To explain the previous discussions in terms of a search algorithm, the aforementioned characteristics of butterflies are generalized as follows:

1. Every butterfly is supposed to release some fragrance that enables other butterflies (features) to attract one another (features).
2. Each butterfly will fly to the best butterfly releasing the most fragrance, either at random or toward it.
3. The landscape of the objective function affects or determines the stimulus intensity of a butterfly.

Three phases are there in DWBFO such as (1) Initialization phase, (2) Iteration phase and (3) Final phase. In every run of DWBFO, the initialization phase is executed first, then the search of optimal features is executed in an iterative manner and at the last phase, the algorithm finally terminates when the best optimal selection solution is obtained. During the initialization phase, classification accuracy and its solution space is computed in DWBFO algorithm. The parameter values used in DWBFO are also assigned. The positions of butterflies (features) are generated at random in the feature selection search space, using their fitness values and fragrance [30] [31]. After finishing initialization phase then algorithm starts the iteration phase. With every iteration, all butterflies in feature selection solution space move to new positions and after that their classification accuracy values are calculated. In the algorithm, the first fitness values are computed for all the butterflies on various positions in the solution space. After that these butterflies will emit fragrance at their positions by using Eq.(10). In the global search phase, the butterfly moves a step toward the fittest solution (g^*) (optimal features) that is represented by Eq.(11):

$$x_i^{t+1} = x_i^t + \left(r^2 g^* - x_i^t\right) \times f_i * F_W \tag{11}$$

where x_i^t denotes the solution vector x_i for i^{th} butterfly in the iteration number *t*. Here, *g** denotes the current best selected feature solution identified from all the solutions in the current iteration. Fragrance of i^{th} butterfly is denoted by f_i and $r \in [0,1]$ represents a random number. The Local search phase can be presented by Eq.(12),

$$x_i^{t+1} = x_i^t + \left(r^2 x_j^t - x_k^t\right) \times f_i * F_W \tag{12}$$

The j^{th} and k^{th} butterflies from the feature selection solution space are x_i^t and x_k^t respectively. The Eq.(12) becomes a local random walk when x_j^t and x_k^t belong to the same swarm and $r \in [0,1]$ is a random number. The Search for the food and a mating partner by butterflies can happen at both a local and global scale to optimally select the features from the dataset. Switch probability *p* is used in DWBFO to switch between common global searches to intensive local search. Until the stopping criteria are matched, the iteration process continues. When the iteration phase concludes, the algorithm produces the best solution found with the best fitness. In the Eq.(11)-Eq.(12), feature weight is also added to DWBFO algorithm to select optimal number of features in the dataset. The overall steps involved in the proposed DWBFO algorithm are shown in the algorithm 2. In the algorithm 2, initial population are generated via number of features in the dataset (Step 1), and then stimulus intensity In_i at x_i (Step 2) is computed based on the sensor modality *c*, power exponent a (Step 3). These factors are generated via the classification accuracy. Then it starts with stopping criteria (Step 4), for every butterfly in the dataset the fragrance value is computed (Step 6). After that find the best feature in the population (Step 8) and generated random number r (Step 10). If *r<p* then move towards the best butterfly by Eq.(11), else move randomly by Eq.(12). Then update a value (Step 17), and evaluate individuals according to their new position (Step 18). Finally end the process via end while (Step 19). The overall flow of the proposed DWBFO is given in Fig.3.

Algorithm 2: Divergence Weight Butterfly Optimization (DWBFO) Algorithm

```
Input: Medical dataset
Objective function: Classifier accuracy, $f(x)$, $x=(x_1,x_2,\ldots,x_{dim})$; $dim$ = number of dimensions
Output: Selection of Optimal Features
Step 1:  Generate initial population of butterflies via number of features in the dataset
Step 2:  Stimulus intensity $I_i$ is found by classification accuracy
Step 3:  Define sensor modality $c$ power exponent and switch probability $p$
Step 4:  While stopping criteria do not met do
Step 5:  Calculate the fragrance using Eq.(10)
Step 6:  End for
Step 7:  Find the best butterfly $g*$
Step 8:  For each butterfly population ($X$) do
Step 9:  Generate random number $r$
Step 10: If r<p then
Step 11: Move towards the best butterfly (optimal solution) by Eq.(11)
Step 12: Else
Step 13: Move randomly using Eq.(12)
Step 14: End if
Step 15: End for
Step 16: Update the value of $a$
Step 17: Evaluate individual (features) according to their new position
Step 18: End while
Step 19: Output the best solution found
```

The DWBFO algorithm is focused to enhance the accuracy of the classifier using optimal selection of features over the given medical dataset efficiently.

From this, optimal features are selected, then DE algorithm is also introduced for feature selection.

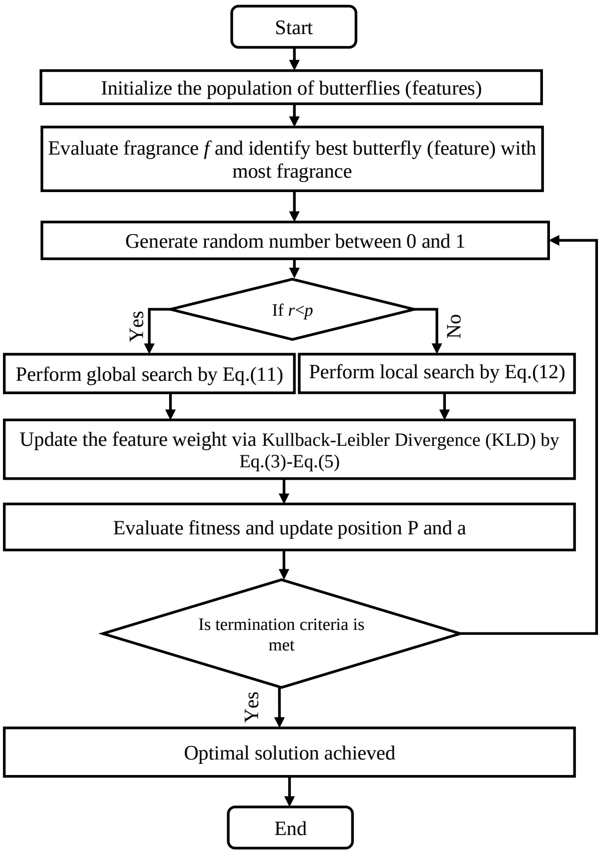
Start
Initialize the population of butterflies (features)
Evaluate fragrance f and identify best butterfly (feature) with most fragrance
Generate random number between 0 and 1
If r<p
Yes
No
Perform global search by Eq.(11)
Perform local search by Eq.(12)
Update the feature weight via Kullback-Leibler Divergence (KLD) by Eq.(3)-Eq.(5)
Evaluate fitness and update position P and a
Is termination criteria is met
No
Yes
Optimal solution achieved
End

Fig.3. Flowchart of Divergence Weight Butterfly Optimization (DWBFO) Algorithm

3.5 DIFFERENTIAL EVOLUTION (DE)

Differential Evolution (DE) is an evolutionary algorithm which searches for traits and is based on an ant colony. DE is a simple and effective strategy that provides the benefits that many optimization methods [32] [33] require. DE makes use of factors that are similar to mutation, selection, and crossover. The effectiveness of DE is determined by how the target vector and difference are handled during the exploring operation for obtaining a task vector. The vector of weight difference was added to the ys_1 which is the third member for building a trial vector among the members of two population, ys_2 and ys_3. This action terms mutation. Using the given Eq.(13), a mutant vector is generated for each target vector $y(I,G)$, j = 1,2,3,…,M mutant vector,

$$W_{j,G+1} = y_{s_1,H} + G\left(y_{s_2,H} - y_{s_3,H}\right) \tag{13}$$

where s_1, s_2, s_3∈{1,2,…,N_P} are integers selected at random, they must be distinct and should be specific from one another and also unique through the operating index j. Scaling factor F(0,1) that the particular population comprises. The introduction of crossover is done so as to increase the variety in relation to perturbed factor vectors. The Eq.(14) depicts the trial vector,

$$V_{j,H+1} = \left(v_{1,j,H+1}, v_{2,j,H+1}, \ldots, v_{E,j,H+1}\right) \tag{14}$$

where

$$v_{kj,H+1} = \begin{cases} W_{kj,H+1} & if\ rand\left(0,1\right) \le d_s \\ y_{kj,H+1} & otherwise \end{cases} \tag{15}$$

where H stands for the current population (features in the dataset) and the trial vector k^{th} j^{th} for the dimension of $v_{kj,H}$. The probability of crossover d_s(0,1) is a user defined value that controls the portion in relation to frequently occurring parameter values which can be replicated through the mutant. In terms of generation, selection is the stage where you choose a vector from the target and trial vectors with the goal of creating an individual. Instead of employing the Crossover and Mutation operators, the proposed DE approach uses encoding of solution (changing real code to binary string) to minimise the classifier's computing time for medical datasets while classifying diseases. Given below are the stages involved in this proposed DE method.

Stage 1: Encoding of Solution

A binary string is used to represent each individual solution from the population. The length of the solution corresponds to different dataset features. The feature selection is indicated by the binary code (1) from the solution, and vice versa. Therefore, S = [F_1,F_2,…,F_m] where m is the number of features in each dataset.

Stage 2: Initial population

Set the DE's total population size to 50, and it will generate a random solution with real values ranging from 0 to 1. The Eq.(16) is used to convert the real-valued solution to binary value,

$$S_{p,q}^{(i)} = \begin{cases} 1 & S_{p,q}^{(i)} > rand \\ 0 & otherwise \end{cases} \tag{16}$$

where, Rand is a random number uniformly distributed between 0 and 1.

Stage 3: Fitness function

This study is used for measuring the single positive integer output. In order to aid the M-SVM in accurately classifying the instances with reduced classification error, the fitness of the produced solution is expressed as follows in Eq.(17),

$$fitness\left(S_p^{(i)}\right) = Error\_rate\left(S_p^{(i)}\right) \tag{17}$$

The result is a classifier's error rate, which is alternatively specified as the testing error rate by Eq.(18),

$$Error\_rate\left(S_p^{(i)}\right) = 100 \times \frac{\text{No. of misclassified instances}}{\text{Number of records}} \tag{18}$$

Stage 4: Finding new solutions

The best/worst solution of the fitness stage is used to create a new solution. The lowest or the worst fitness stage solution is considered the best solution i.e. it obtains with reduced error rate at a generation i. Here, $S_{wt}^{(i)}$ denotes the worst solution and $S_{bt}^{(i)}$ denotes the best solution at an iteration i. Using such constraints, the position (q) of an old solution $S_{p,q}^{(i)}$ is thus formulated by Eq.(19),

$$S_{p,q}^{(i)} = S_{p,q}^{(i)} + A\left|S_{bt,q}^{(i)} - S_{p,q}^{(i)}\right| + B\left|S_{wt,q}^{(i)} - S_{p,q}^{(i)}\right| \tag{19}$$

If the random numbers A and B are in between the 0 and 1, digitalization is used to convert real to binary values for each position in the next generation i+1 using the Eq.(20),

$$S_{p,q}^{(i)} = \begin{cases} 1 & S_{p,q}^{(i+1)} > rand \\ 0 & otherwise \end{cases} \tag{20}$$

Stage 5: Termination criteria

The termination condition occurs if the following conditions are met: 1) fitness convergence rate. 2) Threshold for iterative process and 3) the total number of iterations

3.6 WEIGHTED MAJORITY VOTING (WMV)

The Prediction accuracy can support the decision of those qualified features, that allows it to give much importance to their decision in the vote and as a result, perhaps improve the overall performance beyond what SMV can achieve (where all feature selection methods have same weights). Each vote in WMV is weighted by the predictive accuracy value of the features via a classifier termed as Acc. The Eq.(21), represents the total number of votes for a class c_k,

$$T_k = \sum_{i=1}^{M} Acc\left(A_l\right) \times F_k\left(c_l\right) \tag{21}$$

More study is being done on the efficiency of hybridizing more than one technique to improve accuracy in the detection of heart disease.

3.7 KERNEL WEIGHT CONVOLUTIONAL NEURAL NETWORK (KWCNN) CLASSIFIER

Kernel Weight Convolutional Neural Network (KWCNN) use weighted sharing, down sampling, and local connection techniques which largely minimize the number of required parameters and also the neural network complexity. It is used for classification of multiple disease diagnosis. A convolutional layer, a pooling layer, and a fully connected layer make up the majority of CNNs. The convolutional layer is an important of CNN. Numerous convolutional kernels can be utilised in each convolutional layer for obtaining multiple feature maps from the disease dataset. The Eq.(22) is used to calculate the convolution layer,

$$x_j^l = \sum_{i \in M_j} x_j^{l-1} * k_{ij}^l * we_{ij}^l + b_j^l \qquad (22)$$

where x_j^{l-1} denotes the characteristic map of the previous layer's output, x_j^l is the output of the i^{th} channel of the j^{th} convolution layer, and $f(.)$ denotes the activation function. Here, M_j denotes the subset for the input feature maps used for calculating u_j^l, k_{ij}^l is a convolution kernel, we_{ij}^l denotes the weight of output of the i^{th} channel of the j^{th} convolution layer, and b_j^l denotes the corresponding bias value of j^{th} convolution layer. A pooling layer is usually placed between two convolutional layers. The major purpose of this layer is to minimize the dimensions of the feature map while still maintaining some scale invariance of the features. In the aforementioned Eq.(22), Gaussian Kernel is used to perform transformation for weight calculation, when there exists no previous knowledge about the data by Eq.(23).

$$we_{ij}^l = K(x, y) = e^{-\left(\frac{\|x-y\|^2}{2\sigma^2}\right)} \qquad (23)$$

Between the input data (x) and the output class (y), weight value is computed. If the kernel value between input data and output class is higher than the weight of the layer is increased else it is decreased. The pooling process is similar to the convolution process where it uses a sliding window very similar to a filter, but the calculation is much easier. Mean pooling employs the average value in an attribute range as the pooled value of the area. The function of the fully connected layer is to combine the numerous image maps acquired after the image has been passed through many convolution layers and pooling layers in order to obtain the high-layer semantic features for the dataset for subsequent disease classification.

4. RESULTS AND DISCUSSIONS

The following section provides simulations for the proposed model upon several datasets acquired from the UCI repository, such as heart disease, chronic kidney disease, and hepatitis. The proposed technique is tested on a high-end computer with an i7 processor and 8 GB of RAM. The proposed model's accuracy, specificity, sensitivity, f-measure, and classification performance are all evaluated.

Precision is termed as the percentage of correctly classified positive samples. The Eq.(24) is used to calculate this metric's estimate,

$$\text{Precision}=TP/(FP+TP) \qquad (24)$$

The term recall refers to the positive samples that are assigned to the total number of positive samples, which may be calculated using the Eq.(25),

$$\text{Recall}=TP/(TP+FN) \qquad (25)$$

The harmonic mean of recall and precision is given as F-measure, often known as F1-score and given in Eq.(26),

$$\text{F-measure}=(2*(Recall*Precision))/((Recall+Precision)) \qquad (26)$$

Specificity is important for distinguishing between correct and incorrect conclusions produced by the relevant classifier, which may be represented in Eq.(27),

$$\text{Specificity}=TN/(FP+TN) \qquad (27)$$

Accuracy is a metric which is extensively used for assessing the efficiency of the classifier. The Eq.(28) is used to represent accuracy,

$$\text{Accuracy}=(TP+TN)/(TP+FP+TN+FN) \qquad (28)$$

The *TP*, *TN*, *FP*, and *FN* stand for True Positive, True Negative, False Positive, and False Negative, respectively. Consider a dataset with two classes: true positive denotes the number of appropriate classifications from the first class, while true negative denotes the number of appropriate classifications from the second class. False positive is termed as the number of incorrectly predicted instances in the first class that correspond to the other class. False negative refers to the number of incorrectly predicted instances in the second class that are actually from the first. The Table.1 compares the performance of various classifiers and three feature selection strategies on a heart disease dataset.

Table.1. Metrics Results Comparison of Heart Disease Dataset with Various Classifiers Using DE, OFSM, and Proposed OEFS Algorithm

FS method	**Sensitivity (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	74.55	73.59	71.76	75.54	77.52	88.95
OFSM	91.57	93.72	93.85	94.25	96.14	97.61
OEFS	93.69	94.81	95.12	95.23	95.85	99.40
FS method	**F-Measure (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	69.36	70.12	72.46	74.73	78.71	88.41
OFSM	88.45	91.46	93.67	94.19	96.43	98.20
OEFS	90.63	92.48	94.17	96.39	97.01	99.40
FS method	**Specificity (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	70.51	72.67	74.18	76.53	78.71	85.71
OFSM	90.14	92.17	93.85	94.74	96.43	98.12
OEFS	91.46	93.54	94.81	97.03	97.76	98.52
FS method	**Accuracy (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	68.92	71.19	73.69	74.68	78.51	87.45
OFSM	77.71	92.76	93.93	95.47	96.47	98.01
OEFS	80.33	93.69	94.12	96.03	96.69	99.09

The Table.1 compares the classifiers' performance and feature selection strategies in terms of sensitivity for heart disease dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest sensitivity values of 93.69%, 94.81%, 95.12%, 95.23%, 95.85% and 99.40% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 88.95%, 97.61% and 99.40% respectively (Refer Table.1).

The Table.1 compares the classifiers' performance and feature selection strategies in terms of specificity for heart disease dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest specificity values of 91.46%, 93.54%, 94.81%, 97.03%, 97.76% and 98.52% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM, and OEFS algorithms with 85.71%, 98.12% and 98.52% respectively (Refer Table.1).

The Table.1 compares the classifiers' performance and feature selection strategies based on f-measure for heart disease dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest f-measure values of 90.63%, 92.48%, 94.17%, 96.39%, 97.01% and 99.40% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM, and OEFS algorithms with 88.41%, 98.20% and 99.40% respectively (Refer Table.1).

The Table.1 compares the classifiers' performance and feature selection strategies in terms of accuracy for heart disease dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest accuracy values of 80.33%, 93.69%, 94.12%, 96.03%, 96.69% and 99.09% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM, and OEFS algorithms with 87.45%, 98.01% and 99.09% respectively (Refer Table.1).

The Table.2 compares the performance of various classifiers and three feature selection strategies on CKD dataset.

Table.2. Metrics Results Comparison of Chronic Kidney Disease (CKD) Dataset with Various Classifiers using DE, OFSM, and Proposed OEFS Algorithm

FS method	**Sensitivity (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	70.46	73.72	72.37	81.24	85.14	95.54
OFSM	91.61	93.73	93.32	95.78	96.64	97.95
OEFS	92.32	94.15	94.91	98.78	99.19	99.59
FS method	**F-Measure (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	72.18	70.62	73.49	78.18	84.15	95.35
OFSM	88.62	92.16	93.46	94.67	95.83	98.15
OEFS	90.42	92.47	95.41	98.78	98.99	99.50
FS method	**Specificity (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	68.41	71.16	74.14	80.16	83.18	92.20
OFSM	87.49	90.64	92.67	94.57	95.73	97.41
OEFS	91.46	93.54	94.81	98.03	98.03	99.33
FS method	**Accuracy (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	72.79	72.65	71.37	79.96	82.71	94.26
OFSM	91.64	93.27	93.72	94.51	97.14	97.75
OEFS	93.15	94.41	94.89	98.50	98.75	99.25

The Table.2 compares the classifiers' performance and feature selection strategies in terms of sensitivity for CKD dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest sensitivity values of 92.32%, 94.15%, 94.91%, 98.78%, 99.19% and 99.59% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 95.54%, 97.95% and 99.59% respectively (Refer Table.2).

The Table.2 compares the classifiers' performance and feature selection strategies in terms of specificity for CKD dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest specificity values of 91.46%, 93.54%, 94.81%, 98.03%, 98.03% and 99.33% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 92.20%, 97.41% and 99.33% respectively (Refer Table.2).

The Table.2 compares the classifiers' performance and feature selection strategies based on f-measure for CKD dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest f-measure values of 90.42%, 92.47%, 95.41%, 98.78%, 98.99% and 99.50% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 95.35%, 98.15% and 99.50% respectively (Refer Table.2).

The Table.2 compares the classifiers' performance and feature selection strategies in terms of accuracy for CKD dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest accuracy values of 93.15%, 94.41%, 94.89%, 98.50%, 98.75% and 99.25% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 94.26%, 97.75% and 99.25% respectively (Refer Table.2).

The Table.3 compares the performance of various classifiers and three feature selection strategies on hepatitis dataset.

Table.3. Metrics Results Comparison of Hepatitis Dataset (HD) with Various Classifiers Using DE, OFSM, and Proposed OEFS Algorithm

FS method	**Sensitivity (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	82.92	81.75	85.73	79.84	82.45	92.10
OFSM	94.31	94.14	95.72	90.77	96.76	99.13
OEFS	95.71	96.53	97.54	98.26	98.26	99.26
FS method	**F-Measure (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	78.19	82.19	84.82	86.15	90.21	92.92
OFSM	80.47	84.65	86.63	89.79	95.45	98.70
OEFS	85.62	86.91	90.54	97.83	98.26	99.13
FS method	**Specificity (%)**					
	RF	**GBT**	**ANN**	**SVM**	**M-SVM**	**KWCNN**
DE	75.51	77.46	78.62	79.15	81.43	82.92
OFSM	80.46	82.61	84.74	87.63	93.51	95.00
OEFS	84.62	87.53	90.67	92.50	95.00	97.50
	Accuracy (%)					

FS method	RF	GBT	ANN	SVM	M-SVM	KWCNN
DE	73.16	71.42	74.17	70.27	76.82	89.67
OFSM	93.57	92.63	95.15	90.43	96.61	98.06
OEFS	93.76	94.54	96.49	96.77	97.41	98.70

The Table.3 compares the classifiers' performance and feature selection strategies in terms of sensitivity for hepatitis dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest sensitivity values of 95.71%, 96.53%, 97.54%, 98.26%, 98.26% and 99.13% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 92.10%, 99.13% and 99.26% respectively (Refer Table.3).

The Table.3 compares the classifiers' performance and feature selection strategies in terms of specificity for hepatitis dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest specificity values of 84.62%, 87.53%, 90.67%, 92.50%, 95.00% and 97.50% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 82.92%, 95.00% and 97.50% respectively (Refer Table.3).

The Table.3 compares the classifiers' performance and feature selection strategies based on f-measure for hepatitis dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest f-measure values of 85.62%, 86.91%, 90.54%, 97.83%, 98.26% and 99.13% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 92.92%, 98.70% and 99.13% respectively (Refer Table.3).

The Table.3 compares the classifiers' performance and feature selection strategies in terms of accuracy for hepatitis dataset. For the RF, GBT, ANN, SVM, M-SVM, and KWCNN classifiers, the proposed OEFS-based feature selection algorithm provides the highest f-measure values of 93.76%, 94.54%, 96.49%, 96.77%, 97.41% and 98.70% respectively. The proposed KWCNN classifier also delivers the best results for DE, OFSM and OEFS algorithms with 89.67%, 98.06% and 98.70% respectively (Refer Table.3).

5. CONCLUSION AND FUTURE WORK

Data Mining (DM) is a vibrant research area in medical disease diagnosis that is increasingly expanding. The death rate from these diseases can be reduced if the disease is detected early. In this paper, Feature Selection (FS) and deep learning classifiers are used to create an effective automated disease diagnostic model. The disease diagnosis was chosen for three critical diseases: heart disease, chronic kidney disease (CKD), and hepatitis. The focus of this study is on heterogeneous ensembles of feature selection approaches, which have shown to be very promising in dealing with the feature selection problem. To achieve better results, the OEFS algorithm is proposed, which combines multiple feature selection models such as DWEHO, DWBFO, and DE. The weight values of the features are computed in addition in both the DWEHO and DWBFO algorithms to improve the feature selection results. Kullback-Leibler (KL) is a divergence measure that is used to calculate the weight of a feature based on its range of values. For DWEHO and DWBFO, this weight is used to choose which features to update in the population. Combining DWEHO, DWBFO, and DE subsets of features, Weighted Majority Voting (WMV) is used to determine an ideal subset of features. For the classification of multiple diseases, the Kernel Weight Convolutional Neural Network (KWCNN) is introduced. When there is no prior knowledge about the data, the KWCNN classifier uses a Gaussian kernel to conduct transformation and weight calculation. The accuracy of the classifier improves as a result of this. Extensive research on disease datasets from many domains have demonstrated that using the OEFS method can result in a significant increase in stability without sacrificing predictive performance. This research emphasises the need of developing an OEFS algorithm that achieves a good balance between final predictive performance and selection process stability. When compared to existing classifiers, the findings reveal that the proposed classifier offers higher values for all measures. Furthermore, as a future line of research, it may be worthwhile to investigate the full potentiality of hybrid ensemble approaches-based classifiers, in which variety is introduced both at the data and algorithmic levels.

REFERENCES

[1] S. De and B. Chakraborty, "Disease Detection System (DDS) using Machine Learning Technique", *Proceedings of International Conference on Machine Learning with Health Care Perspective*, pp. 107-132, 2020.

[2] M. Fatima and M. Pasha, "Survey of Machine Learning Algorithms for Disease Diagnostic", *Journal of Intelligent Learning Systems and Applications*, Vol. 9, No. 1, pp. 1-16, 2017.

[3] J.G. Richens, C.M. Lee and S. Johri, "Improving the Accuracy of Medical Diagnosis with Causal Machine Learning", *Nature Communications*, Vol. 11, No. 1, pp.1-9, 2020.

[4] S. Chatterjee, K. Khunti and M.J. Davies, "Type 2 Diabetes", *The Lancet*, Vol. 389, pp. 2239-2251, 2017.

[5] E.M. El Houby, "A Survey on Applying Machine Learning Techniques for Management of Diseases", *Journal of Applied Biomedicine*, Vol. 16, No. 3, pp. 165-174, 2018.

[6] E. Dovgan, Y.C. Li and S. Syed Abdul, "Using Machine Learning Models to Predict the Initiation of Renal Replacement Therapy among Chronic Kidney Disease Patients", *Plos One*, Vol. 15, No. 6, pp. 1-13, 2020.

[7] G. Ahmad, B.S. Khan and M.S. Aslam, "Automated Diagnosis of Hepatitis B using Multilayer Mamdani Fuzzy Inference System", *Journal of Healthcare Engineering*, Vol. 2019, pp. 1-7, 2019.

[8] N.K. Kumar and D. Vigneswari, "Hepatitis-Infectious Disease Prediction using Classification Algorithms", *Research Journal of Pharmacy and Technology*, Vol. 12, No. 8, pp. 3720-3725, 2019.

[9] G. Manogaran, R. Varatharajan and M.K. Priyan, "Hybrid Recommendation System for Heart Disease Diagnosis based on Multiple Kernel Learning with Adaptive Neuro-Fuzzy Inference System", *Multimedia Tools and Applications*, Vol. 77, No. 4, pp. 4379-4399, 2018.

[10] A.D. Dolatabadi, S.E.Z. Khadem and B.M. Asl, "Automated Diagnosis of Coronary Artery Disease (CAD) Patients using Optimized SVM", *Computer Methods and Programs in Biomedicine*, Vol. 138, pp. 117-126, 2017.

[11] M. Abdar, M. Zomorodi-Moghadam, R. Das and I.H. Ting, "Performance Analysis of Classification Algorithms on Early Detection of Liver Disease", *Expert Systems with Applications*, Vol. 67, pp. 239-251, 2017.

[12] J. Tang, S. Alelyani and H. Liu, "Feature Selection for Classification: A Review", *Proceedings of International Conference on Data Classification: Algorithms and Applications*, pp.1-33, 2014.

[13] D. Guan, W. Yuan and M.K. Rasel, "A Review of Ensemble Learning based Feature Selection", *IETE Technical Review*, Vol. 31, No. 3, pp. 190-198, 2014.

[14] V. Bolon Canedo and A. Alonso-Betanzos, "Ensembles for Feature Selection: A Review and Future Trends", *Information Fusion*, Vol. 52, pp. 1-12, 2019.

[15] N. Hoque, M. Singh and D.K. Bhattacharyya, "EFS-MI: An Ensemble Feature Selection Method for Classification", *Complex and Intelligent Systems*, Vol. 4, No. 2, pp. 105-118, 2018.

[16] M.J. Reddy and D.N. Kumar, "Computational Algorithms Inspired by Biological Processes and Evolution", *Current Science*, Vol. 103, No. 4, pp. 370-380, 2012.

[17] P. Ghosh, F.J.M. Shamrat and A.A. Khan, "Optimization of Prediction Method of Chronic Kidney Disease using Machine Learning Algorithm", *Proceedings of International Joint Symposium on Artificial Intelligence and Natural Language Processing*, pp. 1-6, 2020.

[18] J.S. Sartakhti, M.H. Zangooei and K. Mozafari, "Hepatitis Disease Diagnosis using a Novel Hybrid Method based on Support Vector Machine and Simulated Annealing (SVM-SA)", *Computer Methods and Programs in Biomedicine*, Vol. 108, No. 2, pp. 570-579, 2012.

[19] D.C. Yadav and S. Pal, "Prediction of Heart Disease using Feature Selection and Random Forest Ensemble Method", *International Journal of Pharmaceutical Research*, Vol. 12, No. 4, pp. 56-66, 2020.

[20] C.J. Qin, Q. Guan and X.P. Wang, "Application of Ensemble Algorithm Integrating Multiple Criteria Feature Selection in Coronary Heart Disease Detection", *Biomedical Engineering: Applications, Basis and Communications*, Vol. 29, No. 6, pp. 1-13, 2017.

[21] M.S. Amin, Y.K. Chiam and K.D. Varathan, "Identification of Significant Features and Data Mining Techniques in Predicting Heart Disease", *Telematics and Informatics*, Vol. 36, pp. 82-93, 2019.

[22] M. Nilashi, H. Ahmadi E. Akbari, "A Predictive Method for Hepatitis Disease Diagnosis using Ensembles of Neuro-Fuzzy Technique", *Journal of Infection and Public Health*, Vol. 12, No. 1, pp. 13-20, 2019.

[23] V.R. Elgin Christo, B. Minu and A. Kannan, "Correlation-Based Ensemble Feature Selection using Bioinspired Algorithms and Classification using Backpropagation Neural Network", *Computational and Mathematical Methods in Medicine*, Vol. 2019, pp. 1-17, 2019.

[24] O.A. Jongbo, T.A. Olowookere and A.O. Adetunmbi, "Performance Evaluation of an Ensemble Method for Diagnosis of Chronic Kidney Disease with Feature Selection Technique", *Proceedings of International Conference on Decision Aid Sciences and Application*, pp. 959-965, 2020.

[25] C. Saranya and G. Manikandan, "A Study on Normalization Techniques for Privacy Preserving Data Mining", *International Journal of Engineering and Technology*, Vol. 5, No. 3, pp. 2701-2704, 2013.

[26] Z. Liu, "A Method of SVM with Normalization in Intrusion Detection", *Procedia Environmental Sciences*, Vol. 11, pp. 256-262, 2011.

[27] A. Kiran and D. Vasumathi, "Data Mining: Min-Max Normalization Based Data Perturbation Technique for Privacy Preservation", *Proceedings of International Conference on Computational Intelligence and Informatics*, pp. 723-734, 2020.

[28] G.G. Wang, S. Deb and L.D.S. Coelho, "Elephant Herding Optimization", *Proceedings of International Symposium on Computational and Business Intelligence*, pp. 1-5, 2015.

[29] S. Arora and S. Singh, "Butterfly Optimization Algorithm: A Novel Approach for Global Optimization", *Soft Computing*, Vol. 23, No. 3, pp. 715-734, 2019.

[30] M. Tubishat, M. Alswaitti, S. Mirjalili and T.A. Rana, "Dynamic Butterfly Optimization Algorithm for Feature Selection", *IEEE Access*, Vol. 8, pp. 194303-194314, 2020.

[31] M. Alweshah, "Solving Feature Selection Problems by Combining Mutation and Crossover Operations with the Monarch Butterfly Optimization Algorithm", *Applied Intelligence*, Vol. 51, No. 6, pp. 4058-4081, 2020.

[32] W. Yi, Y. Zhou and J. Mou, "An Improved Adaptive Differential Evolution Algorithm for Continuous Optimization", *Expert Systems with Applications*, Vol. 44, pp. 1-12, 2016.

[33] Y. Chen, W. Xie and X. Zou, "A Binary Differential Evolution Algorithm Learning from Explored Solutions", *Neurocomputing*, Vol. 149, pp. 1038-1047, 2015.