WILEY | Hindawi

Research Article

RLMD-PA: A Reinforcement Learning-Based Myocarditis Diagnosis Combined with a Population-Based Algorithm for Pretraining Weights

Seyed Vahid Moravvej [iD],[1,2] Roohallah Alizadehsani,[3] Sadia Khanam,[4] Zahra Sobhaninia,[1] Afshin Shoeibi,[5] Fahime Khozeimeh,[3] Zahra Alizadeh Sani,[6] Ru-San Tan,[7,8] Abbas Khosravi,[3,9] Saeid Nahavandi,[3,9] Nahrizul Adib Kadri [iD],[10] Muhammad Mokhzaini Azizan [iD],[11] N. Arunkumar,[12] and U.Rajendra Acharya[13,14,15]

[1]Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran
[2]Department of Electrical and Computer Engineering, University of Kashan, Kashan, Iran
[3]Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Waurn Ponds, Victoria 3216, Australia
[4]Dhaka Dental College, Dhaka, Bangladesh
[5]Faculty of Electrical Engineering, FPGA Lab, K. N. Toosi University of Technology, Tehran, Iran
[6]Omid Hospital, Iran University of Medical Sciences, Tehran, Iran
[7]Department of Cardiology, National Heart Centre Singapore, Singapore, Singapore
[8]Duke-NUS Medical School, Singapore
[9]Harvard Paulson School of Engineering and Applied Sciences, Harvard University, Allston, MA 02134, USA
[10]Department of Biomedical Engineering, Faculty of Engineering, University Malaya, Kuala Lumpur 50603, Malaysia
[11]Faculty of Engineering and Built Environment, Universiti Sains Islam Malaysia, Bandar Baru Nilai 71800,
  Negeri Sembilan, Malaysia
[12]Department of Biomedical Engineering, Rathinam College of Engineering, Coimbatore, India
[13]Ngee Ann Polytechnic, Singapore 599489, Singapore
[14]Department of Biomedical Informatics and Medical Engineering, Asia University, Taichung, Taiwan
[15]Department of Biomedical Engineering, School of Science and Technology, Singapore University of Social Sciences, Singapore,
  Singapore

Correspondence should be addressed to Seyed Vahid Moravvej; sa.moravvej@alumni.iut.ac.ir and Muhammad Mokhzaini Azizan; mokhzainiazizan@usim.edu.my

Myocarditis is heart muscle inflammation that is becoming more prevalent these days, especially with the prevalence of COVID-19. Noninvasive imaging cardiac magnetic resonance (CMR) can be used to diagnose myocarditis, but the interpretation is time-consuming and requires expert physicians. Computer-aided diagnostic systems can facilitate the automatic screening of CMR images for triage. This paper presents an automatic model for myocarditis classification based on a deep reinforcement learning approach called as reinforcement learning-based myocarditis diagnosis combined with population-based algorithm (RLMD-PA) that we evaluated using the Z-Alizadeh Sani myocarditis dataset of CMR images prospectively acquired at Omid Hospital, Tehran. This model addresses the imbalanced classification problem inherent to the CMR dataset and formulates the classification problem as a sequential decision-making process. The policy of architecture is based on convolutional neural network (CNN). To implement this model, we first apply the artificial bee colony (ABC) algorithm to obtain initial values for RLMD-PA weights. Next, the agent receives a sample at each step and classifies it. For each classification act, the agent gets a reward from the environment in which the reward of the minority class

is greater than the reward of the majority class. Eventually, the agent finds an optimal policy under the guidance of a particular reward function and a helpful learning environment. Experimental results based on standard performance metrics show that RLMD-PA has achieved high accuracy for myocarditis classification, indicating that the proposed model is suitable for myocarditis diagnosis.

1. Introduction

Myocarditis is a condition that causes inflammation of the heart muscle [1]. It can affect heart pump function as well as electrical activation and conduction, resulting in heart failure and arrhythmia, respectively. The etiology is diverse, including infection (e.g., viral infections such as COVID-19 and parvovirus) [2], systemic inflammatory and autoimmune diseases, and drug reactions. Symptoms of myocarditis include chest pain, fatigue, and shortness of breath [3]. Patients with suspected myocarditis should seek cardiology advice for early diagnosis and treatment. Endomyocardial biopsy, an invasive procedure, is recommended in severe cases to confirm the diagnosis and to guide treatment [4]. Management comprises supportive measures, symptomatic heart failure therapy, antimicrobials for identified infective agents, and immunosuppression for severe inflammation. Early diagnosis and prompt institution of treatment can significantly reduce morbidity and mortality. Noninvasive cardiac imaging with cardiovascular magnetic resonance imaging (MRI) [5] can help clinch the diagnosis. However, MRI requires expert interpretation, which is manually intensive and subject to operator bias. In this regard, automated diagnostic systems can be developed that employ various machine learning and data mining algorithms to solve medical image classification problems efficiently [6]. They can be applied to reporting workflows to screen images automatically, saving physicians time, reducing errors, and enhancing diagnostic accuracy.

Excellent performance of in-depth models has been demonstrated in diverse applications, including natural language processing [7–9], computer vision, and medical image analysis [10, 11]. Deep learning-based algorithms converge with suitable weights to minimize the error between the real and predicted outputs. Typically, deep models use gradient-based algorithms as backpropagation to learn the weights. However, such optimization methods are sensitive to initial weights and may become trapped in local minima [12]. This issue is mainly encountered during classification [13]. Few researchers have shown that population-based meta-heuristic (PBMH) algorithms [14, 15] may help to overcome this problem [16]. Among PBMH algorithms, the ABC algorithm is one of the most effective optimizers [17, 18]. It emulates the behavior of bees in nature and, unlike traditional optimization algorithms, dispenses with the need to calculate gradients, thereby reducing the probability of getting stuck in local optimizations [19].

Classification performance in many machine learning algorithms may be adversely affected by imbalanced classification [20], which occurs when one class contains disproportionately more data than the others [21]. While imbalanced models may still attain reasonable detection rates for majority samples, the performance for minority samples is weak as minority class specimens can be difficult to identify due to their rarity and randomness. Also, misalignment of

minority class samples can result in high costs. Methods have been proposed to address the problem at two levels [22]: data level and algorithmic level. In the former [23–25], training data are manipulated to balance the class distribution by oversampling minority class and/or undersampling majority class [26]. For instance, the synthetic minority oversampling technique (SMOTE) generates new samples by linear interpolation between adjoining minority samples [24], whereas NearMiss undersamples majority samples using the nearest neighbor algorithm [25]. Of note, oversampling and undersampling can risk overfitting and loss of worthy information, respectively [27]. At the algorithmic level, the importance of the minority class can be raised using techniques [28–32] that include cost-sensitive learning, ensemble learning, and decision threshold adjustment. In cost-sensitive learning, different incorrect classification costs are attributed to the loss function for the whole class, with a higher cost being allocated to minority class misclassification. Ensemble learning systems train several subclassifications and then apply voting or combination to obtain better results. Threshold adjustment techniques train the classifier in the imbalanced dataset and modify the decision threshold during the test. Deep learning-based methods have also been suggested for imbalanced data classification [33–35]. The authors in Reference [36] introduced a new loss function for deep networks that could capture classification errors from both minority and majority classes. Reference [37] introduces a method that could learn the unique features of an imbalanced dataset while maintaining intercluster and interclass margins.

To the best of our knowledge, only one work [3] based on deep learning models has been proposed for the diagnosis of myocarditis. The authors developed an algorithm for classifying images based on CNN and the k-means algorithm [38], which has the following workflow: after the data preprocessing stage, the images were placed in several clusters, and each cluster was considered a class in which the CNN classified. The algorithm was repeated for different clusters, and all the results were combined for the final decision. The main problem with the method was that it considered the image matrix as a vector in k-means, which resulted in missed pixels around a specific pixel.

This paper presents a method based on the ABC algorithm and reinforcement learning called RLMD-PA that we believe would address the above mentioned problems. The RLMD-PA model poses the classification problem as a guessing game embodied in a sequential decision-making process. At each step, the agent receives an environmental state represented by a training instance and then executes a classification under the direction of a policy. If the agent performs classification perfectly, it will be given a positive reward and, otherwise, a negative one. The minority class is rewarded more than the majority class. The agent's goal is to accumulate as many rewards as possible during the

sequential decision-making process to classify the samples as correctly as possible.

The main contributions of this article are as follows: (1) we considered the classification problem of medical images as a sequential decision-making process. We presented a reinforcement learning-based algorithm for imbalanced classification; (2) instead of randomly weighting, we have developed an encoding strategy and calculated the optimal initial value using the ABC algorithm, and (3) this work is based on a new well-annotated MRI dataset acquired from Tehran's Omid Hospital that we have named the Z-Alizadeh Sani myocarditis dataset and made publicly downloadable.

The rest of the article is structured as follows: the second section is a brief overview of the ABC algorithm and its working. The third section introduces the proposed model. The fourth section presents the evaluation criteria, dataset, and analysis of the results. The last section states the conclusions and future works.

2. Background

2.1. Artificial Bee Colony Algorithm.

Artificial bee colony (ABC) introduced by Karaboga and Basturk [39] is one of the most efficient algorithms for optimizing numerical problems. It is straightforward, robust, and population-based [19]. The algorithm emulates the intelligent foraging behavior of bees to arrive at the optimal solution. There is a list of food sources that bees seek out over time to get to the best positions. The algorithm involves three groups of bees: employed bees, onlooker bees, and scout bees. Employed bees discover the positions of food sources, whereas onlooker bees wait in the hive for the nectar from food positions to be sent by employed bees. Onlooker bees use the information to select food source positions. Once an employed bee has exhausted the food source, it becomes a scout bee to search for new positions randomly. The number of employed bees equals the number of unemployed (onlooker and scout) bees. The steps for optimizing an algorithm using the ABC algorithm are as follows:

(1) Initialization: in the first step, an initial population S of size C is formed from the positions (solutions), as in

$$s_i^j = s_{\min}^j + \text{rand}(0,1)\left(s_{\max}^j - s_{\min}^j\right), \tag{1}$$

where i represents the i-th position, each solution s_i is D dimensions, and D means the number of parameters that must be optimized. $s_{\min}^j$ and $s_{\max}^j$ are the smallest and largest values in s^j, respectively.

(2) Employed bee phase: at this point, new solutions are recognized by searching the neighborhood for current potential solutions. To keep the population size constant, the quality of new solutions is evaluated. If it is better than the previous ones, it will be replaced; otherwise, it will remain fixed. This step can be formed as follows:

$$v_i^j = s_i^j + \varphi_i^j\left(s_i^j - s_k^j\right), \tag{2}$$

where k is a random solution such that $k \neq i$. φ_i^j is a random number picked from the interval [0, 1]. The potentially new solution v_i is obtained by changing only one element of s_i.

(3) Onlooker bee phase: for the onlooker bees update, one solution is stochastically elected from the potential solutions, that is, one of the open facility solutions, according to the probability relation p_i anticipated as follows:

$$p_i = \frac{\text{fit}(s_i)}{\sum_{n=1}^{C} \text{fit}(s_i)}. \tag{3}$$

The selection process follows the equation provided: the more appropriate a solution is, the higher the chance it will be selected. If the chosen employed bee scores higher than the current onlooker bee's current solution, the current solution replaces the previous one. This process is repeated for all onlooker bees in population S.

(4) Scout bee phase: a solution that does not improve its fit after some repetitions can get the algorithm caught up in local optimization [40]. To prevent this, once the solution's fit does not improve after t iterations, the algorithm will discard it, and a new solution will be supplied according to equation (2).

(5) Algorithm end condition: although different conditions can be defined for the end of the algorithm, the term termination is repeated in this study, which means that the algorithm ends after MaxItr iterations.

The complete ABC algorithm is given in Algorithm 1.

2.2. Reinforcement Learning.

Reinforcement learning [41] is an important branch of machine learning that encompasses many domains. Reinforcement learning can achieve relatively good classification results because it can effectively learn the compelling features of noisy data. In Reference [42], the authors defined classification as a sequential decision problem that used several factors to interact with the environment in order to learn an optimal policy function. Due to the complex simulation between the factors and the environment, the run time was inordinately prolonged. The model presented in [43] is a classification based on reinforcement learning provided for noisy text data. The proposed structure comprises of two classifiers: sample selector and relational classifier. The former selects a quality sentence from the noisy data by following the agent, whereas the latter classifier learns acceptable quality performance from clean data and gives a delayed reward to the sample selector for feedback. Finally, the model yields a superior classifier and quality dataset. The authors in Reference [44] proposed a solution for time series data in which the reward function and Markov process are explicitly defined. In various specific applications [45–48], reinforcement learning has been applied to learn the efficient features. These models promote valuable features for the classification, which leads to higher rewards that guide the agent to select more worthy features.

Input: D: dimensions of every solution, C: population size, limit: number of cycles, MaxItr: maximum number of iterations;
(1) Initialize a population of solutions $S = [s_1, s_2, \ldots, s_C]$ using equation (1);
(2) Itr = 1;
(3) **while** Itr ≤ MaxItr **do**
(4)   //Employed Bee Phase
(5)   **for** $i = 1$ *to* C **do**
(6)   Produce new solution x_{new} using equation (2);
(7)   Calculate the fitness f_{new} for x_{new};
(8)   Replace x_{new} with x_i if better;
(9)   Calculate the probability p for every solution in S using equation (3);
(10)   //Onlooker Bee Phase
(11)   **for** $i = 1$ *to* C **do**
(12)   **if** rand $(0, 1) < p_i$ **then**
(13)   Produce new solution x_{new} by using equation (2);
(14)   Calculate the fitness f_{new} for x_{new};
(15)   Replace x_{new} with x_i if better;
(16)   //Scout Bee Phase
(17)   If an abandoned solution is found, replace it with the solution produced by equation (2);
(18)   Put the best solution in x_{best};
(19)   Itr = Itr + 1;

ALGORITHM 1: Pseudocode of the ABC algorithm.

To date, limited work has been done on deep learning for the classification of imbalanced data. In Reference [44], an ensemble pruning technique for deciding subclassifiers that adopted reinforcement learning was proposed. However, the model underperformed when the amount of data was increased. This is because it is difficult to choose classifiers when there are too many subclassifications.

3. The Proposed Solution

The overall structure of the proposed model is shown in Figure 1. We considered two critical options for classification. In the first step, we formulated a vector that includes all the learnable weights in our model. We assumed an initial value for the weights with ABC and then applied the backpropagation in the rest of the path. As mentioned, another problem that most classifiers suffer from, including ours, is imbalanced data. To address this, we employed reinforcement learning [49]. These concepts are detailed in the following sections.

3.1. Pretraining Phase. Weight initialization of deep networks is an essential part of deep models. Sometimes, incorrect initial values can lead to a failure of convergence in the model. The proposed model has a deep network with weights θ that need to be optimized. In this section, we present an encoding strategy and fitness function for the ABC algorithm.

3.2. Encoding Strategy. In our work, the encoding strategy aims to arrange the CNN and feed-forward weights in a vector that will be considered the position of the bees in the ABC. Setting the specific weights is a challenge. Nevertheless, we have designed an encoding strategy that is as appropriate as possible after a few experiments. Figure 2 illustrates an example with encoding of a three-layer CNN network with three filters in each layer and a feed-

forward network with three hidden layers. Note that all weight matrices in the vector are stored in rows.

3.3. Fitness Function. The fitness function is defined as follows to measure the effectiveness of a solution in the ABC algorithm [12]:

$$\text{Fitness} = \frac{1}{1 + \sum_{i=1}^{N} (y_i - \widehat{y}_i)^2}, \tag{4}$$

where N is the total number of samples, and y_i and $\widehat{y}_i$ are the target and predicted labels for i-th data, respectively.

4. Classification

Due to the difference in the amount of data between our two classes, we face the problem of imbalanced classification. To address this, we used the imbalanced classification Markov decision process (ICMDP) to construct a sequential decision problem. In reinforcement learning, an agent tries to obtain an optimal policy by performing a series of actions in the environment while maximizing its score. In the case of our model, a sample of the dataset is provided to the agent at each time point and classified. The environment then transmits the immediate score to the agent. A positive score corresponds to a correct rating, whereas a wrong rating gives a negative one. By maximizing cumulative rewards, the agent can arrive at the optimal policy. Let $D = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots, (x_N, y_N)\}$ be the imbalanced set of existing images with N samples, where x_i corresponds to the i-th image, and y_i is its corresponding label. The following explains the intended settings:

(i) Policy π_θ: policy π means a mapping function $S \longrightarrow A$, where S and A are a set of states and actions, respectively. In other words, every $\pi_\theta(s_t)$
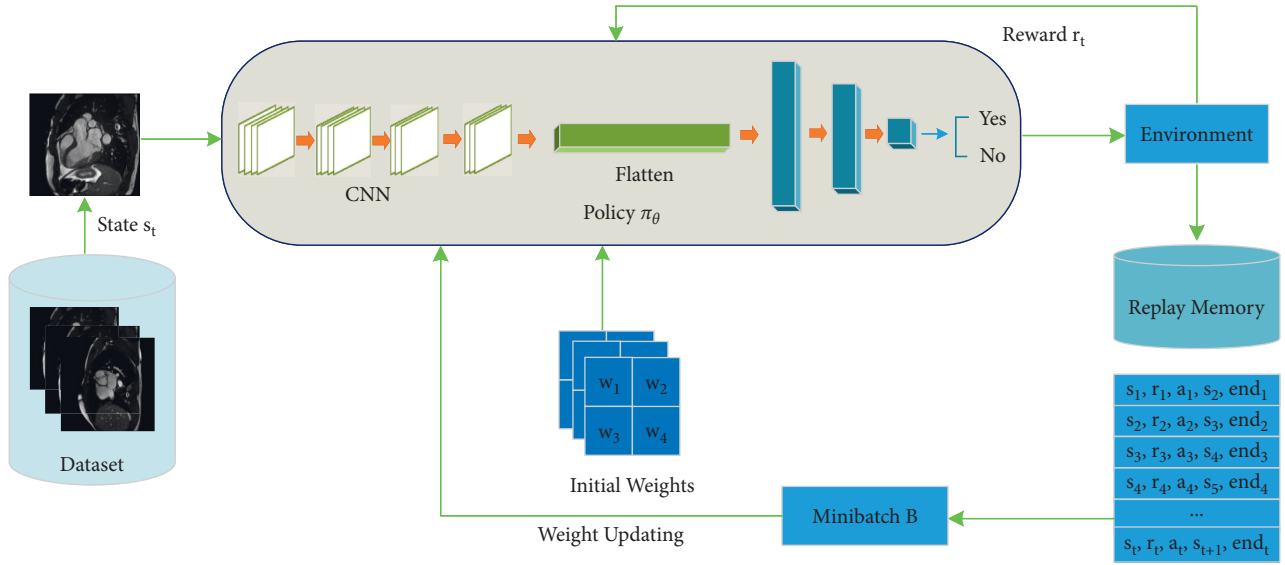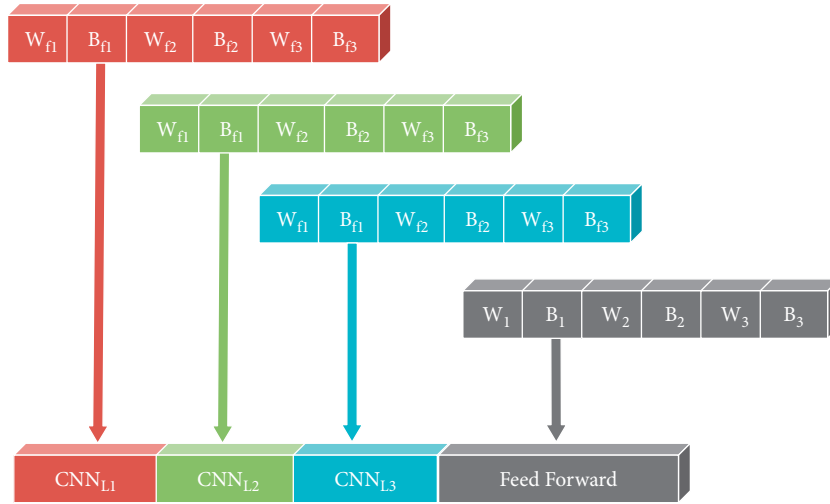
Figure 1: Overall process of RLMD-PA.

Figure 2: Placement of weights in a vector.

means performing the action a_t in the state s_t. π_θ is acknowledged as the classifier model with weights θ.

(ii) State $\mathbf{s_t}$: each state s_t is mapped with sample x_t from the dataset D. The first data x_1 are deemed the initial state of s_1. For the model not to learn a particular order, the D is shuffled in each episode.

(iii) Action $\mathbf{a_t}$: action a_t is performed to predict the label x_t. Since the offered classification is binary, $a_t \in \{0, 1\}$, zero represents the minority class and one represents the majority class.

(iv) Reward $\mathbf{r_t}$: reward considers the performance of an action. An agent with the correct classification gets a positive reward; otherwise, it gets a negative reward. The amount of this bonus should not be the same for both classes. Rewards can significantly improve model performance because the level of reward and action has been carefully calibrated. In this work,

the prize is defined for action according to the following equation [27]:

$$r_t(s_t, a_t, l_t) = \begin{cases} +1, a_t = y_t \text{ and } s_t \in D_H, \\ -1, a_t \neq y_t \text{ and } s_t \in D_H, \\ \lambda, a_t = y_t \text{ and } s_t \in D_S, \\ -\lambda, a_t \neq y_t \text{ and } s_t \in D_S. \end{cases} \quad (5)$$

where D_H and D_S represent the minority and majority classes, that is, healthy and sick, respectively, and λ is a value in the interval [0,1]. The reward λ is less than $1/-1$ as the minority class becomes more critical due to fewer data. In effect, we can ascribe more importance to the minority class in order for it to approximate the majority class. In the results section, we will see the importance of the value λ.

(v) Terminal **E**: the training process is completed at several terminal states, which occur in every training episode. An episode is the transition trajectory from an initial state to a final state, namely, $\{(s_1, a_1, y_1), (s_2, a_2, y_2), (s_3, a_3, y_3), \ldots, (s_t, a_t, y_t)\}$. In our case, an episode stops when all the training data have been classified or when a sample of the minority class is misclassified.

(vi) Transition probability **P**: the agent goes from state s_t to the next state s_{t+1} based on the order of the read data. The transition probability is determined as $p(s_{t+1}|s_t, a_t)$.

In ICMDP, the policy function reports the probability of all labels by receiving a sample:

$$\pi(a|s) = P(a_t = a|s_t = s). \tag{6}$$

In reinforcement learning, the intention is to maximize the discounted cumulative reward, or in mathematical terms, to attain a high limit for the following expression:

$$g_t = \sum_{k=0}^{\infty} \gamma^k. \tag{7}$$

Equation (7) is termed the return function, which contains all the accumulated return values of the agent searches in space. The discount factor $\gamma \in (0, 1]$ [50] is the coefficient of the effect of each reward. The function Q measures the quality of a state-action combination:

$$Q^{\pi}(s, a) = E_{\pi}[g_t|s_t = s, a_t = a]. \tag{8}$$

Equation (8) is expanded according to Bellman's formula [51]

$$Q^{\pi}(s, a) = E_{\pi}[r_t + \gamma Q^{\pi}(s_{t+1}, a_{t+1})|s_t = s, a_t = a]. \tag{9}$$

By maximizing the function Q supported by π, more cumulative rewards can be achieved. The optimal policy of π^* is assessed by considering the function Q^* as follows:

$$\pi^*(a|s) = \begin{cases} 1, & a = \text{argmax}_a Q^*(s, a), \\ 0, & \text{else.} \end{cases} \tag{10}$$

By combining the two equations (9) and (10), the function Q^* is expressed as follows [27]:

$$Q^*(s, a) = E_{\pi}[r_t + \gamma \text{max}_a Q^*(s_{t+1}, a_{t+1})|s_t = s, a_t = a]. \tag{11}$$

In a low-dimensional space state, the function Q can be easily solved by a table. However, the table technique is inadequate when space is joined. To solve this problem, Q-learning algorithms are used. In these algorithms, the tuple (s, a, r, s_0) received from equation (11) is saved as experience replay memory M. The agent gets a mini-batch B from M and executes the gradient descent on these data according to the following equation:

$$L(\theta_k) = \sum_{(s,a,r,s') \in B} (y - Q(s, a; \theta_k))^2, \tag{12}$$

where y is an estimate of the function Q expressed as follows [27]:

$$y = \begin{cases} r, & \text{end = True,} \\ r + \gamma \text{max}_{a'} Q(s', a'; \theta_{k-1}), & \text{else,} \end{cases} \tag{13}$$

where s' is the following state s, and a' is the action performed in s'; end means whether the agent makes a wrong classification for the minority class or not. Finally, the policy weights π can be updated as follows:

$$\theta = \theta + l \frac{\nabla L(\theta_k)}{\nabla(\theta_k)},$$

$$\frac{\nabla L(\theta_k)}{\nabla(\theta_k)} = -2 \sum_{(s,a,r,s') \in B} (y - Q(s, a; \theta_k)) \frac{\nabla Q(s, a; \theta_k)}{\nabla(\theta_k)}. \tag{14}$$

In conclusion, the optimal function Q^* can be achieved by minimizing the loss function presented in equation (12). Notably, the optimal policy of π^* is taken using Q^*, which is the optimal model for the proposed classifier.

4.1. Overall Algorithm. We devised the simulation environment according to the above. The structure of the policy network depends on the complexity and number of training samples. According to the structure of the training samples and the output, the network input equals to the number of data classes, which is equivalent to 2. The general training algorithm of the RLMD-PA model is displayed in Algorithm 2. In this algorithm, the policy weights are first initialized using the ABC algorithm, and then, the agent continues the training process until an optimal policy is reached. Action is based on a greedy policy, which is also evaluated by Algorithm 3. The algorithm is repeated for E times, which is taken as 18,000 in this paper. At each step, the policy network weights are stored.

5. Empirical Evaluation

5.1. Dataset. Cardiac magnetic resonance imaging (CMR) [52] allows for comprehensive anatomical and functional evaluation of the heart as well as detailed tissue characterization [53]. It is the preeminent imaging modality for noninvasive diagnosis myocarditis without biopsy. The Lake Louise criterion (LLC) [54] introduced benchmark criteria for diagnosing myocarditis using CMR [55] based on the presence of myocardial necrosis, edema, and hyperemia. The presence of late gadolinium enhancement confirms myocardial necrotic damage. T2-weighted images uncover areas of interstitial edema, which indicates inflammatory response. T1-weighted images before and after contrast can depict hyperemia in the myocardial tissue. Fulfilling two of three LLC criteria confers 80% accuracy for diagnosing myocarditis [56]. This article presents a model for identifying myocarditis by considering the three LLC criteria.

A one-year CMR research project on myocarditis was conducted from September 2016 at Omid Hospital in Tehran, Iran, where we performed CMR on patients who were clinically suspected to have myocarditis (e.g., chest pain, elevated troponin, negative functional imaging and/or coronary angiographic findings, and suspected viral etiology)

Data: $D = \{(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots, (x_N, y_N)\}$
(1) Initialize the weights of policy π using Algorithm 1;
(2) Initialize environment ε;
(3) Initialize replay memory M;
(4) **for** *Episode e = 1 to E* **do**
(5)   Shuffle the data D;
(6) $s_1 = x_1$;
(7)   **for** $t = 1$ *to N* **do**
(8) $a_t = \pi_\theta(s_t)$; //select an action based on ε -greedy
(9) $[r_t, \text{end}_t] = \text{Reward }(x_t, a_t, y_t)$;
(10) $s_{t+1} = x_{t+1}$;
(11)   Save $(s_t, a_t, r_t, s_{t+1}, \text{end}_t)$ to M;
(12)   Sample randomly a mini-batch of transitions $(s_k, a_k, r_k, s_{k+1}, \text{end}_k)$ $(s_k, a_k, r_k, s_{k+1}, \text{end}_k)$ from M;
(13) $y_k = \begin{cases} r_k, \text{end}_k = \text{True} \\ r_k + \gamma \max_{a'} Q(s_{k+1}, a'; \theta), \text{end}_k = \text{False} \end{cases}$,   Accumulate   gradients   w.r.t θ: $\theta = \theta$ + $\theta = \theta + (\nabla L(\theta_k)/\nabla(\theta_k))$;
(14)   **if** $\text{end}_k == $ *True* **then**
(15)   break;

ALGORITHM 2: Pseudocode of the RLMD-PA algorithm.

(1) **Function** Reward (x_t, a_t, y_t):
(2) $\text{end}_t = \text{False}$;
(3)   **if** $s_t \in D_H$ **then**
(4)   **if** $y_t == a_t$ **then**
(5) $r_t = 1$;
(6)   **else**
(7) $r_t = -1$;
(8) $\text{end}_t = \text{True}$;
(9)   **end**
(10)   **else**
(11)   **if** $y_t == a_t$ **then**
(12) $r_t = \lambda$;
(13)   **else**
(14) $r_t = -\lambda$;
(15)   **end**
(16)   **end**
(17)   **return** $[r_t, \text{end}_t]$
(18) End Function

ALGORITHM 3: Pseudocode of Reward function.

and the treating physician assessed that CMR would likely affect clinical management (e.g., ongoing symptoms, ongoing myocardial injury evidenced by persistent ECG abnormalities, and presence of ventricular dysfunction). The protocol had been approved by the local ethics committee. CMR examination was performed on a 1.5-Tesla system [57]. All cases were scanned with body coils in standard supine position. T1-weighted images were acquired in the axial views. Shortly after gadolinium injection, the T1-weighted sequences were repeated. After approximately 10–15 minutes, late gadolinium enhancement [58] sequences were performed in standard left ventricular short- and long-axis views. Table 1 summarizes the CMR sequence parameters [3].

A total of 586 patients were identified who had positive evidence of myocarditis on the CMR images, which might show one or more areas of disease. A total of 307 healthy subjects were included as controls. We chose eight CMR images from each patient or control subject for the analysis, which were one long-axis image and one short-axis image acquired using each of the following four CMR sequences: late gadolinium enhancement, perfusion, T2-weighted, and steady-state free precession. The final CMR dataset comprises 4,686 and 2,449 samples from sick (i.e., myocarditis) and healthy subjects, respectively. Figure 3 shows example images obtained from this dataset. It may be noted that in this study, analysis is performed at the image level, and not at the patient level. In other words, prediction is based on a single image regardless of how many images are available for each patient.

Institutional approval was allowed to use the patient datasets in research studies for diagnostic and therapeutic

TABLE 1: Characteristics of the Z-Alizadeh Sani myocarditis dataset.

Protocols	TE (mm)	TR (mm)	NF	Slice thickness (mm)	Concatenation and slice number	NE	Breath-hold time (s)
CINE_segmented (true FISP) long axis (LAX)	1.15	33.60	15	7	3	1	8
CINE_segmented (true FISP) short axis (SAX)	1.11	31.92	15	7	15	1	8
T2-weighted (TIRM) LAX, precontrast	52	800	Noncine	10	3	1	9
T2-weighted (TIRM) SAX, precontrast	52	800	Noncine	10	5	1	10
T1 relative-weighted TSE (Trigger)-AXIA- dark blood pre- and postcontrast	24	525	Noncine	8	5	1	7
Late-GD enhancement LGE (high-resolution PSIR) SAX and LAX	3.16	666	Noncine	8	1	1	7

TE: time echo, TR: time repetition, NF: number of frames, NE: number of excitations.
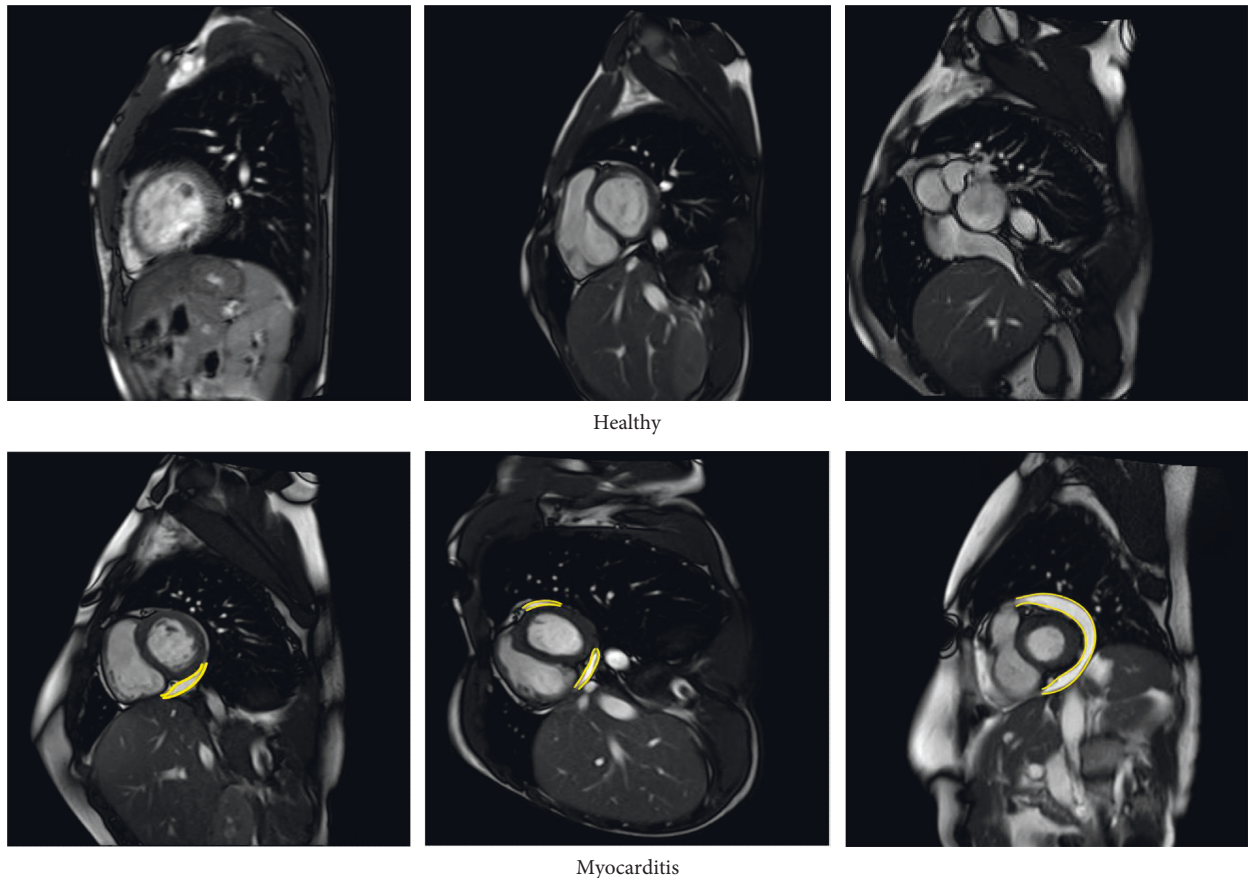

Healthy

Myocarditis

FIGURE 3: Typical healthy and myocarditis images obtained from the Z-Alizadeh Sani myocarditis dataset. The yellow lines indicate the location of myocarditis.

TABLE 2: 5-CV classification performances (accuracy, recall, and precision) obtained for automated myocarditis detection using various combinations of deep learning models with the Z-Alizadeh Sani myocarditis dataset.

	Accuracy					Recall					Precision				
Method	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.
CNN-KCL [3]	0.783	0.811	0.846	0.810	0.024	0.732	0.738	0.807	0.751	0.032	0.704	0.752	0.789	0.745	0.032
CNN + random weight	0.755	0.770	0.807	0.772	0.021	0.695	0.713	0.755	0.717	0.213	0.666	0.685	0.737	0.691	0.029
CNN + ABC	0.799	0.803	0.845	0.815	0.020	0.741	0.766	0.814	0.771	0.027	0.726	0.729	0.783	0.746	0.027
CNN + RL	0.821	0.829	0.869	0.840	0.021	0.762	0.798	0.835	0.801	0.028	0.745	0.772	0.819	0.779	0.029
RLMD-PA (CNN + ABC + RL)	0.862	0.884	0.912	0.886	0.020	0.837	0.869	0.879	0.863	0.017	0.804	0.837	0.886	0.840	0.034

TABLE 3: 5-CV classification performances (F-measure, specificity, and G-means) obtained for automated myocarditis detection using various combinations of methods with the Z-Alizadeh Sani myocarditis dataset.

	F-measure					Specificity					G-means				
Method	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.
CNN-KCL [3]	0.718	0.746	0.798	0.748	0.031	0.814	0.852	0.870	0.845	0.022	0.772	0.795	0.838	0.797	0.025
CNN + random weight	0.681	0.702	0.746	0.704	0.026	0.788	0.800	0.838	0.806	0.020	0.742	0.759	0.795	0.760	0.021
CNN + ABC	0.735	0.745	0.798	0.758	0.026	0.826	0.835	0.864	0.842	0.018	0.787	0.795	0.839	0.806	0.021
CNN + RL	0.767	0.777	0.827	0.790	0.026	0.836	0.864	0.889	0.863	0.020	0.811	0.821	0.862	0.831	0.022
RLMD-PA (CNN + ABC + RL)	0.820	0.847	0.882	0.851	0.024	0.877	0.900	0.932	0.901	0.024	0.857	0.879	0.905	0.882	0.019

purposes. Approval was granted on the grounds of existing datasets. Informed consent was received from all of the patients in this study. All methods were carried out in accordance with relevant guidelines and regulations. Ethical approval for using these data was obtained from the Tehran Omid Hospital.

5.2. Metrics.
To evaluate the classification performance of the proposed model, we used six standard performance metrics, namely, accuracy, recall, precision, F-measure, specificity, and G-means [59], and they are defined as follows:

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

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

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

$$\text{F} - \text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}},$$

$$\text{Specificity} = \frac{TN}{TN + FP},$$

$$\text{G} - \text{means} = \sqrt{\text{Recall} \times \text{Specificity}},$$

(15)

where TP, TN, FN, and FP are true positive, true negative, false negative, and false positive, respectively. The F-measure and G-means are commonly applied to evaluate imbalanced classification [27], which aligns nicely with our dataset sample distribution and the reason for existing our proposed method. In addition, it is noteworthy that our prediction is per image. In this way, the intelligent myocarditis classification system can effectively screen entire CMR studies and flag individual images for scrutiny by physician readers. For this purpose, low FP and high recall metrics would be desired.

5.3. Details of Model.
This work used Python and the PyTorch framework. The codes are written in Jupyter notebook. We used five layers of two-dimensional convolution for the CNN network with 128, 64, 32, 16, and 8 filters. The size of the kernel, stride, and padding in each layer are 3, 2, and 1 for
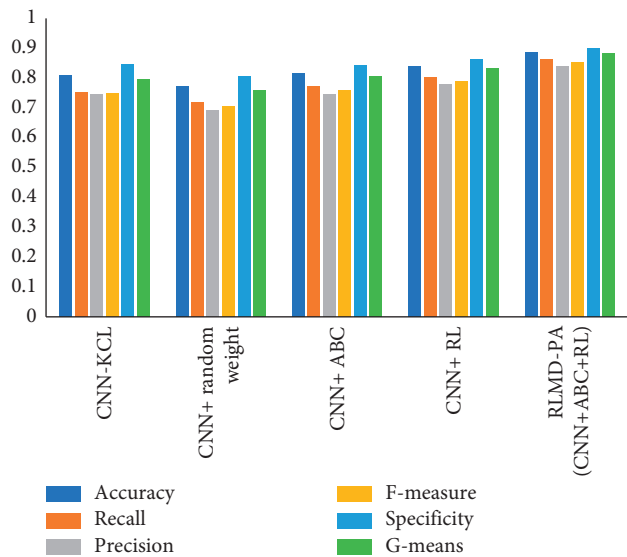

FIGURE 4: Performance of deep learning models on the mean.

both dimensions, respectively. Each convolution layer involves a max-pooling layer with dimensions of 2×2. The three fully connected layers have 128, 64, and 32 hidden layers, respectively. To prevent overfitting, dropout with a probability of 0.4 and early stopping are employed. In every experiment, the batch size is set to 64. The images in the dataset are in gray-scale and light intensities of image pixels are mapped to the range [0, 1]. The images in the dataset come in different sizes and are all resized to 100×100 for analysis.

5.4. Experimental Results.
While standard techniques like data augmentation and weighted loss function [60] can sometimes be used to correct the imbalanced data distributions, they are not applicable in all situations. In our experiments, data augmentation and weighted loss function do not enrich our model, which is not unexpected.

We used k-fold cross-validation ($k = 5$ or 5-CV) in all our implementations. The entire dataset is divided into k subsets. $k - 1$ subsets are applied for training and the remaining one k for test. This procedure is iterated k times until all data subsets are utilized exactly four times for training and once for testing. All parameters are expressed as

TABLE 4: 5-CV classification performances (accuracy, recall, and precision) obtained for automated myocarditis detection using various machine learning algorithms with the Z-Alizadeh Sani myocarditis dataset.

	Accuracy					Recall					Precision				
Method	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.
SVM	0.568	0.691	0.754	0.683	0.070	0.674	0.745	0.778	0.737	0.042	0.450	0.565	0.651	0.565	0.074
KNN	0.480	0.614	0.635	0.588	0.064	0.399	0.637	0.683	0.589	0.111	0.337	0.490	0.511	0.460	0.072
Naïve Bayes	0.547	0.632	0.676	0.615	0.051	0.388	0.534	0.713	0.565	0.134	0.395	0.510	0.553	0.484	0.062
Logistic regression	0.627	0.662	0.720	0.661	0.038	0.583	0.658	0.741	0.657	0.057	0.503	0.542	0.603	0.541	0.041
Random forests	0.415	0.550	0.590	0.530	0.070	0.537	0.683	0.711	0.648	0.071	0.329	0.437	0.469	0.420	0.056

TABLE 5: 5-CV classification performance (F-measure, specificity, and G-means) obtained for automated myocarditis detection using various machine learning algorithms with the Z-Alizadeh Sani myocarditis dataset.

	F-measure					Specificity					G-means				
Method	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.
SVM	0.540	0.652	0.695	0.639	0.060	0.505	0.662	0.760	0.651	0.093	0.583	0.704	0.752	0.692	0.065
KNN	0.365	0.554	0.585	0.516	0.089	0.528	0.601	0.629	0.587	0.039	0.459	0.619	0.643	0.587	0.075
Naïve Bayes	0.391	0.522	0.623	0.520	0.092	0.610	0.642	0.692	0.645	0.031	0.499	0.608	0.682	0.600	0.072
Logistic regression	0.565	0.571	0.665	0.593	0.042	0.606	0.665	0.716	0.663	0.049	0.631	0.646	0.724	0.659	0.038
Random forests	0.408	0.533	0.559	0.509	0.063	0.342	0.471	0.529	0.459	0.071	0.429	0.567	0.605	0.545	0.071

means, standard deviations, medians, minimums, and maximums. First, we compared our proposed method with the only published work in this field, CNN-KCL [3]. Next, to investigate the contributions of the two distinct components ABC and RL in our model, we compared the performance of a basic model without ABC and RL, that is, CNN + random weight, versus the models CNN + ABC and CNN + RL, which used ABC and RL for training, respectively. The evaluation results of our RLMD-PA model performance as well as the aforementioned comparisons on the Z-Alizadeh Sani myocarditis dataset are presented in Tables 2 and 3. In general, the RLMD-PA model reduces the error by more than 43%. From the means of all the performance metrics, the RLMD-PA model outperforms the CNN-KCL method as well as CNN + random weight, CNN + ABC, and CNN + RL combinations of its components. Both ABC and RL individually improve on the basic CNN network across all assessed performance metrics, which supports the use of combined approaches of initial weight and reinforcement learning. For better visualization, the results are illustrated in Figure 4. In terms of time, the best model was obtained after 100 iterations in 2 hours, while CNN-KCL got the best after 350 iterations in 5 hours.

Standard machine learning classifiers have not been successful in classifying medical images, because they typically assume images as one-dimensional vectors, which cause the neighboring pixels of a specific pixel to be spaced apart. In order to compare with our deep model, we used five algorithms: support vector machine (SVM) [61], k-nearest neighbor [62], naïve Bayes [63], logistic regression [64], and random forests [65] to classify the CMR images of the study dataset. SVM performed the best among these methods but is still inferior to deep models. The results are summarized in Tables 4 and 5, and the mean performance metrics is shown in Figure 5.
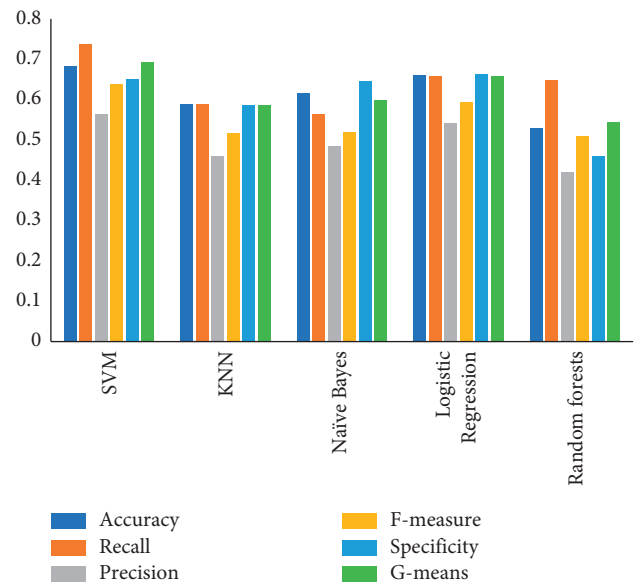

FIGURE 5: Performance of traditional methods on the mean.

5.5. Investigation of Other Metaheuristic Algorithms on the Algorithm.
The proposed model employs ABC algorithm in conjunction with backpropagation for the initial value. To compare the performance of ABC versus alternative instructors, we employed ABC in our model with five conventional algorithms, namely, gradient descent with momentum backpropagation (GDM) [66], gradient descent with adaptive learning rate backpropagation (GDA) [67], gradient descent with momentum and adaptive learning rate backpropagation (GDMA) [68], one-step secant backpropagation (OSS) [69], and Bayesian regularization backpropagation (BR) [70], and four metaheuristic

TABLE 6: Parameter setting for the experiments.

Algorithm	Parameter	Value
	Limit	$n_e \times$ dimensionality of problem
ABC	n_o	50% of the colony
	n_e	50% of the colony
	n_s	1
GWO	No parameters	
	Constant for loudness update	0.50
BAT	Constant for an emission rate update	0.50
	Initial pulse emission rate	0.001
COA	Discovery rate of alien solutions	0.25
WOA	B	1

TABLE 7: Results of 5-CV classification performances (accuracy, recall, and precision) obtained for automated myocarditis detection using various conventional and metaheuristic algorithms with the Z-Alizadeh Sani myocarditis dataset.

	Accuracy					Recall					Precision				
Method	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.
CNN + GDM + RL	0.811	0.857	0.868	0.849	0.022	0.784	0.801	0.830	0.806	0.018	0.732	0.806	0.825	0.796	0.038
CNN + GDA + RL	0.817	0.846	0.857	0.840	0.017	0.784	0.812	0.837	0.808	0.022	0.742	0.786	0.828	0.778	0.035
CNN + GDMA + RL	0.829	0.855	0.887	0.854	0.025	0.764	0.816	0.855	0.817	0.037	0.752	0.809	0.849	0.800	0.037
CNN + OSS + RL	0.823	0.849	0.867	0.846	0.016	0.741	0.814	0.837	0.804	0.037	0.778	0.787	0.814	0.791	0.015
CNN + BR + RL	0.826	0.833	0.855	0.837	0.012	0.745	0.796	0.812	0.785	0.027	0.752	0.761	0.850	0.784	0.041
CNN + GWO + RL	0.833	0.848	0.869	0.850	0.016	0.771	0.796	0.842	0.804	0.027	0.769	0.800	0.816	0.797	0.020
CNN + BAT + RL	0.837	0.847	0.865	0.851	0.013	0.778	0.782	0.833	0.796	0.024	0.787	0.805	0.830	0.807	0.016
CNN + COA + RL	0.815	0.843	0.882	0.844	0.028	0.750	0.826	0.856	0.813	0.046	0.748	0.757	0.838	0.781	0.039
CNN + WOA + RL	0.820	0.845	0.847	0.837	0.012	0.750	0.826	0.814	0.789	0.021	0.742	0.783	0.807	0.781	0.024

TABLE 8: Results of 5-CV classification performances (F-measure, specificity, and G-means) obtained for automated myocarditis detection using various conventional and metaheuristic algorithms with the Z-Alizadeh Sani myocarditis dataset.

	F-measure					Specificity					G-means				
Method	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.	Min	Median	Max	Mean	Std.dev.
CNN + GDM + RL	0.757	0.811	0.825	0.801	0.026	0.827	0.882	0.898	0.875	0.028	0.805	0.848	0.860	0.840	0.021
CNN + GDA + RL	0.765	0.799	0.811	0.792	0.019	0.834	0.863	0.902	0.860	0.028	0.812	0.839	0.850	0.834	0.015
CNN + GDMA + RL	0.771	0.806	0.849	0.808	0.033	0.838	0.880	0.909	0.877	0.026	0.815	0.843	0.878	0.846	0.026
CNN + OSS + RL	0.759	0.799	0.825	0.797	0.024	0.859	0.873	0.885	0.872	0.010	0.804	0.839	0.861	0.837	0.021
CNN + BR + RL	0.776	0.784	0.794	0.784	0.007	0.841	0.850	0.921	0.868	0.034	0.821	0.825	0.829	0.825	0.003
CNN + GWO + RL	0.779	0.797	0.828	0.801	0.021	0.856	0.880	0.889	0.877	0.013	0.821	0.836	0.863	0.840	0.018
CNN + BAT + RL	0.782	0.793	0.823	0.801	0.018	0.873	0.885	0.901	0.885	0.010	0.824	0.832	0.859	0.839	0.016
CNN + COA + RL	0.752	0.803	0.844	0.796	0.038	0.835	0.854	0.901	0.862	0.028	0.800	0.845	0.876	0.837	0.031
CNN + WOA + RL	0.768	0.793	0.798	0.785	0.014	0.832	0.869	0.888	0.866	0.021	0.812	0.832	0.839	0.827	0.012

algorithms, namely, gray wolf optimization (GWO) [71], the Bat algorithm (BA) [72], Cuckoo optimization algorithm (COA) [73], and whale optimization algorithm (WOA) [74]. The population size and number of function evaluations are 100 and 25,000 for all metaheuristic algorithms, respectively. Other parameter settings can be seen in Table 6. The performance metrics of these comparisons are summarized in Tables 7 and 8 and illustrated in Figure 6. In general, metaheuristic algorithms are better than conventional algorithms with the exception of GDMA in terms of accuracy, recall, and F-measure scores. Importantly, the ABC algorithm outperformed all conventional and metaheuristic algorithms to improve the error in the recall and F-measure criteria by more than 25% and 22%, respectively.

5.6. Explore the Reward Function. The reward function is a practical device that helps the agent to achieve the goal. In this work, the minority class reward is $+1/-1$, while the majority is $+\lambda/-\lambda$. To examine the effect of the value λ on the classification model, we test 10 values of $\lambda \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ on the model. Details of the results for all the criteria for these experiments are given in Table 9. For better visualization, we have plotted the trends in Figure 7. On examination, for the accuracy criterion, when λ takes the values from [0, 0.3], the chart has an ascending trend, and from [0.3, 1] has a descending move` This process is valid for all criteria. If $\lambda = 0$, the importance of the majority class is disregarded, and if $\lambda = 1$, the importance of both classes is the same. Although the minority class is more important to us, the majority class cannot be ignored.
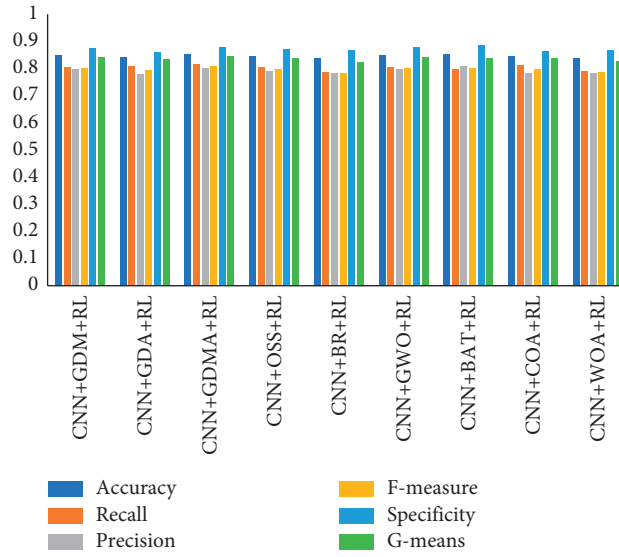
FIGURE 6: Performance of conventional and metaheuristic models on the mean.

TABLE 9: Performance evaluation obtained for various values of λ as the reward of the majority class.

λ	Accuracy	Recall	Precision	F-measure	Specificity	G-means
0	0.807	0.778	0.727	0.752	0.824	0.801
0.1	0.838	0.814	0.769	0.791	0.853	0.833
0.2	0.867	0.844	0.810	0.827	0.880	0.862
0.3	0.884	0.858	0.837	0.847	0.900	0.879
0.4	0.877	0.848	0.830	0.839	0.895	0.871
0.5	0.857	0.814	0.807	0.810	0.883	0.848
0.6	0.845	0.798	0.792	0.795	0.874	0.835
0.7	0.825	0.764	0.768	0.766	0.861	0.811
0.8	0.807	0.738	0.746	0.742	0.848	0.791
0.9	0.792	0.709	0.730	0.719	0.842	0.773
1	0.779	0.695	0.710	0.702	0.829	0.759
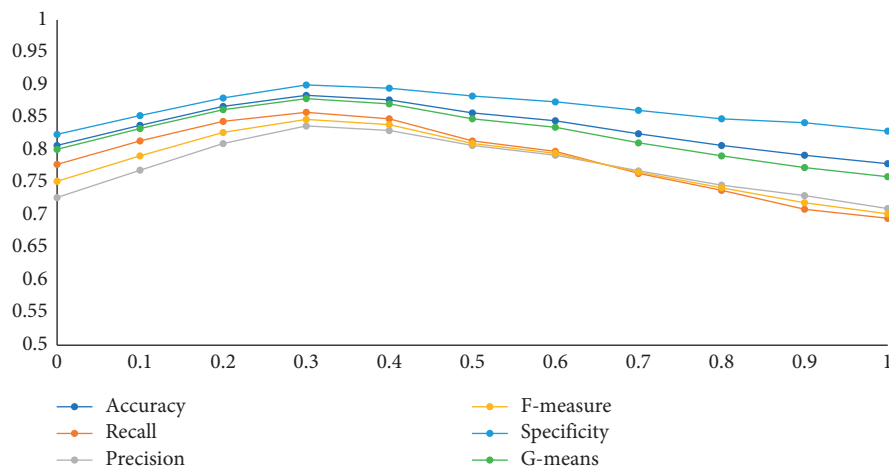
FIGURE 7: Graphical view of change in the performance parameters due to variation in λ.

6. Conclusion and Future Directions

This article presents a new model for classifying myocarditis images. The proposed model consists of two steps. First, the model weights are initialized using the ABC algorithm. Next, the model is considered an ICMDP problem. The environment assigns a high reward to the minority class and a low reward to the majority class. The algorithm terminates when the agent makes a wrong classification for the minority class, or the number of episodes runs out. We performed several experiments to examine various factors that affect the performance of the proposed model. The designed experiments confirmed that the RLMD-PA model with ABC and RL is an effective classifier for myocarditis images.

In the future, we will try to employ ensemble convolutional neural network (ECNN), as our model to use a set of CNN networks and connect them to yield higher performance. In addition, we can also work with the generative adversarial network (GAN), which is widely used in many applications. It may be worth exploring to employ the developed model for other medical applications such as stroke detection, cancer detection and plaque detection.

Data Availability

The dataset used to support the findings of this study is available on GitHub: https://github.com/vahid-moravvej/Z-Alizadeh-Sani-myocarditis-dataset.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Seyed Vahid Moravvej, Roohallah Alizadehsani, and Ru-San Tan contributed to prepare the first draft. Nahrizul Adib Kadri, Muhammad Mokhzaini Azizan, and U. Rajendra Acharya contributed to editing the final draft. Sadia Khanam and Zahra Sobhaninia contributed to all analysis of the data and produced the results accordingly. Afshin Shoeibi and Fahime Khozeimeh searched for papers and then extracted data. Zahra Alizadeh Sani, N. Arunkumar, Abbas Khosravi, and Saeid Nahavandi provided overall guidance and managed the project.

Acknowledgments

References

[1] L. T. Cooper Jr, "Myocarditis," *New England Journal Of Medicine*, vol. 360, no. 15, pp. 1526–1538, 2009.

[2] V. Kytö, P. Saukko, E. Lignitz et al., "Diagnosis and presentation of fatal myocarditis," *Human Pathology*, vol. 36, no. 9, pp. 1003–1007, 2005.

[3] D. Sharifrazi, R. Alizadehsani, J. Hassannataj Joloudari et al., "Cnn-kcl: automatic myocarditis diagnosis using convolutional neural network combined with k-means clustering," *Mathematical Biosciences and Engineering*, vol. 19, no. 3, pp. 2381–2402, 2020.

[4] A. Asher, "A review of endomyocardial biopsy and current practice in england: out of date or underutilised," *British Journal of Cardiology*, vol. 24, pp. 108–112, 2017.

[5] G. Katti, S. Arshiya Ara, and A. Shireen, "Magnetic resonance imaging (mri)–a review," *International Journal of Dental Clinics*, vol. 3, no. 1, pp. 65–70, 2011.

[6] M. Abdar, E. Nasarian, X. Zhou et al., "Performance improvement of decision trees for diagnosis of coronary artery disease using multi filtering approach," in *Proceedings of the 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp. 26–30, IEEE, Singapore, February 2019.

[7] S. Vahid Moravvej, A. Mirzaei, and M. Safayani, "Biomedical text summarization using conditional generative adversarial network (cgan)," 2021, http://arxiv.org/abs/2110.11870.

[8] M. Salimi Sartakhti, M. J. Maleki Kahaki, S. Vahid Moravvej, M. javadi Joortani, and A. Bagheri, "Persian language model based on bilstm model on covid-19 corpus," in *Proceedings of the 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, pp. 1–5, IEEE, Kashan, Iran, April 2021.

[9] S. V. Moravvej, M. J. M. Kahaki, M. S. Sartakhti, and M. Joodaki, "Efficient gan-based method for extractive summarization," *Journal of Electrical and Computer Engineering Innovations*, vol. 10, pp. 287–298, 2021.

[10] S. Zahra, H. Danesh, R. Kafieh, J. Jothi Balaji, and V. Lakshminarayanan, "Determination of foveal avascular zone parameters using a new location-aware deep-learning method," in *Applications of Machine Learning 2021* vol. 11843, San Diego, CA, USA, International Society for Optics and Photonics, Article ID 1184311, 2021.

[11] S. Zahra, N. Karimi, P. Khadivi, R. Roshandel, and S. Samavi, "Brain tumor classification using medial residual encoder layers," 2020, https://arxiv.org/abs/2011.00628.

[12] S. V. Moravvej, S. J. Mousavirad, M. H. Moghadam, and M. Saadatmand, T. Mantoro, M. Lee, M. A. Ayu, K. W. Wong, and A. N. Hidayanto, An lstm-based plagiarism detection via attention mechanism and a population-based approach for pre-training parameters with imbalanced classes," in *Neural Information Processing*, pp. 690–701, Springer International Publishing, Cham, 2021.

[13] M. Mavrovouniotis and S. Yang, "Training neural networks with ant colony optimization algorithms for pattern classification," *Soft Computing*, vol. 19, no. 6, pp. 1511–1522, 2015.

[14] B. Liu, L. Wang, Y. Liu, and S. Wang, "A unified framework for population-based metaheuristics," *Annals of Operations Research*, vol. 186, no. 1, pp. 231–262, 2011.

[15] S. Vakilian, S. Vahid Moravvej, and F. Ali, "Using the cuckoo algorithm to optimizing the response time and energy consumption cost of fog nodes by considering collaboration in the fog layer," in *Proceedings of the 2021 5th International Conference on Internet of Things and Applications (IoT)*, pp. 1–5, IEEE, Isfahan, Iran, May 2021.

[16] S. Jalaleddin Mousavirad, G. Schaefer, I. Korovin, and D. Oliva, "Rde-op: a region-based differential evolution algorithm incorporation opposition-based learning for optimising the learning process of multi-layer neural networks," in *Proceedings of the International Conference on the Applications of Evolutionary Computation (Part of EvoStar)*, pp. 407–420, Springer, April 2021.

[17] S. Vakilian, S. Vahid Moravvej, and F. Ali, "Using the artificial bee colony (abc) algorithm in collaboration with the fog nodes

in the internet of things three-layer architecture," in *Proceedings of the 2021 29th Iranian Conference on Electrical Engineering (ICEE)*, pp. 509–513, IEEE, Tehran, Iran, Islamic Republic, May 2021.

[18] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Pérez, "Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach," *Knowledge-Based Systems*, vol. 159, pp. 1–8, 2018.

[19] D. Karaboga, B. Akay, and C. Ozturk, "Artificial bee colony (abc) optimization algorithm for training feed-forward neural networks," in *Proceedings of the International conference on modeling decisions for artificial intelligence*, pp. 318–329, Springer, Kitakyushu, Japan, August 2007.

[20] S. Vahid Moravvej, M. Joodaki, and M. Salimi Sartakhti, "A method based on an attention mechanism to measure the similarity of two sentences," in *Proceedings of the 2021 7th International Conference on Web Research (ICWR)*, pp. 238–242, IEEE, Tehran, Iran, May 2021.

[21] S. Vahid Moravvej, M. J. Maleki Kahaki, M. Salimi Sartakhti, and A. Mirzaei, "A method based on attention mechanism using bidirectional long-short term memory (blstm) for question answering," in *Proceedings of the 2021 29th Iranian Conference on Electrical Engineering (ICEE)*, pp. 460–464, IEEE, Tehran, Iran, May 2021.

[22] H. Guo, Y. Li, and J. Shang, "Learning from class-imbalanced data: review of methods and applications," *Expert Systems with Applications*, vol. 73, pp. 220–239, 2017.

[23] C. DrummondR. C. Holte et al., "C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *Workshop on learning from imbalanced datasets II*vol. 11, , pp. 1–8, Citeseer, 2003.

[24] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-smote: a new over-sampling method in imbalanced data sets learning," in *Proceedings of the International Conference on Intelligent Computing*, pp. 878–887, Springer, Hefei, China, August 2005.

[25] I. Mani and I. Zhang, "Knn approach to unbalanced data distributions: a case study involving information extraction," in *Proceedings of the workshop on learning from imbalanced datasets*, vol. 126, ICML, United States, June 2003.

[26] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 20–29, 2004.

[27] E. Lin, Q. Chen, and X. Qi, "Deep reinforcement learning for imbalanced classification," *Applied Intelligence*, vol. 50, no. 8, pp. 2488–2502, 2020.

[28] K. Veropoulos, C. Campbell, N. Cristianini et al., "Controlling the sensitivity of support vector machines," in *Proceedings of the international joint conference on AI*, vol. 55, July 1999.

[29] G. Wu and E. Y. Chang, "Kba: kernel boundary alignment considering imbalanced data distribution," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 786–795, 2005.

[30] Y. Tang, Y.-Q. Zhang, V. Nitesh, and S. Krasser, "Svms modeling for highly imbalanced classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 1, pp. 281–288, 2008.

[31] B. Krawczyk and M. Woźniak, "Cost-sensitive neural network with roc-based moving threshold for imbalanced classification," in *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning*, pp. 45–52, Springer, Guimaraes, Portugal, January 2015.

[32] H. Yu, C. Sun, X. Yang, W. Yang, J. Shen, and Y. Qi, "Odocelm: optimal decision outputs compensation-based extreme learning machine for classifying imbalanced data," *Knowledge-Based Systems*, vol. 92, pp. 55–70, 2016.

[33] Y. Yan, M. Chen, M.-L. Shyu, and S.-C. Chen, "Deep learning for imbalanced multimedia data classification," in *Proceedings of the 2015 IEEE international symposium on multimedia (ISM)*, pp. 483–488, IEEE, Miami, FL, USA, December 2015.

[34] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3573–3587, 2017.

[35] D. Qi, S. Gong, and X. Zhu, "Imbalanced deep learning by minority class incremental rectification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 6, pp. 1367–1381, 2018.

[36] S. Wang, W. Liu, J. Wu et al., "Training deep neural networks on imbalanced data sets," in *Proceedings of the 2016 international joint conference on neural networks (IJCNN)*, pp. 4368–4374, IEEE, Vancouver, BC, Canada, July 2016.

[37] C. Huang, Y. Li, C. Change Loy, and X. Tang, "Learning deep representation for imbalanced classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5375–5384, IEEE, Las Vegas, NV, USA, June 2016.

[38] K. Krishna and M. Narasimha Murty, "Genetic k-means algorithm," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 29, no. 3, pp. 433–439, 1999.

[39] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (abc) algorithm," *Applied Soft Computing*, vol. 8, no. 1, pp. 687–697, 2008.

[40] Y. Watanabe, M. Takaya, and A. Yamamura, "Fitness function in abc algorithm for uncapacitated facility location problem," in *Proceedings of the Information and Communication Technology-EurAsia Conference*, pp. 129–138, Springer, Daejeon, Korea, October 2015.

[41] L. Pack, M. L. Littman, and A. W. Moore, "Reinforcement learning: a survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.

[42] M. A. Wiering, H. Van Hasselt, A.-D. Pietersma, and S. Lambert, "Reinforcement learning algorithms for solving classification problems," in *Proceedings of the 2011 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pp. 91–96, IEEE, Paris, France, April 2011.

[43] J. Feng, M. Huang, Li Zhao, Y. Yang, and X. Zhu, "Reinforcement learning for relation classification from noisy data," in *Proceedings of the aaai conference on artificial intelligence*, vol. 32, AAAI Press, August 2018.

[44] C. Martinez, G. Perrin, E. Ramasso, and M. Rombaut, "A deep reinforcement learning approach for early classification of time series," in *Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), pages 2030–2034*, IEEE, Rome, Italy, September 2018.

[45] T. Zhang, M. Huang, and Li Zhao, "Learning structured representation for text classification via reinforcement learning," in *Proceedings of the -Second AAAI Conference on Artificial Intelligence*, AAAI Press, New Orleans, Louisiana, USA, February 2018.

[46] D. Liu and T. Jiang, "Deep reinforcement learning for surgical gesture segmentation and classification," in *Proceedings of the International conference on medical image computing and computer-assisted intervention*, pp. 247–255, Springer, Granada, Spain, September 2018.

[47] D. Zhao, Y. Chen, and L Le, "Deep reinforcement learning with visual attention for vehicle classification," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 9, no. 4, pp. 356–367, 2016.

[48] J. Janisch, T. Lisý, and V. Lisỳ, "Classification with costly features using deep reinforcement learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 3959–3966, 2019.

[49] M. A. Wiering and M. Van Otterlo, "Reinforcement learning," *Adaptation, learning, and optimization*, vol. 12, no. 3, 2012.

[50] R. Amit, R. Meir, and K. Ciosek, "Discount factor as a regularizer in reinforcement learning," in *Proceedings of the International conference on machine learning*, pp. 269–278, PMLR, Vienna, Austria, July 2020.

[51] K. D. AvinashJ. J. F. Sherrerd et al., *Optimization in Economic Theory*, Oxford University Press on Demand, Oxford, England, 1990.

[52] I. S. Syed, J. F. Glockner, D. Feng et al., "Role of cardiac magnetic resonance imaging in the detection of cardiac amyloidosis," *Journal of the American College of Cardiology: Cardiovascular Imaging*, vol. 3, no. 2, pp. 155–164, 2010.

[53] M. Chetrit and M. G. Friedrich, "The unique role of cardiovascular magnetic resonance imaging in acute myocarditis," *F1000Research*, vol. 7, 2018.

[54] M. D. Cornicelli, C. K. Rigsby, K. Rychlik, E. Pahl, and J. D. Robinson, "Diagnostic performance of cardiovascular magnetic resonance native t1 and t2 mapping in pediatric patients with acute myocarditis," *Journal of Cardiovascular Magnetic Resonance: Official Journal of the Society for Cardiovascular Magnetic Resonance*, vol. 21, no. 1, pp. 40–49, 2019.

[55] J. A. Pan, Y. J. Lee, and M. Salerno, "Diagnostic performance of extracellular volume, native t1, and t2 mapping versus lake louise criteria by cardiac magnetic resonance for detection of acute myocarditis: a meta-analysis," *Circulation: Cardiovascular Imaging*, vol. 11, no. 7, Article ID e007598, 2018.

[56] M. A. G. M. Olimulder, J. Van Es, and M. A. Galjee, "The importance of cardiac mri as a diagnostic tool in viral myocarditis-induced cardiomyopathy," *Netherlands Heart Journal*, vol. 17, no. 12, pp. 481–486, 2009.

[57] C. Moenninghoff, L. Umutlu, C. Kloeters et al., "Workflow efficiency of two 1.5 t mr scanners with and without an automated user interface for head examinations," *Academic Radiology*, vol. 20, no. 6, pp. 721–730, 2013.

[58] L. Lin, X. Li, J. Feng et al., "The prognostic value of t1 mapping and late gadolinium enhancement cardiovascular magnetic resonance imaging in patients with light chain amyloidosis," *Journal of Cardiovascular Magnetic Resonance: Official Journal of the Society for Cardiovascular Magnetic Resonance*, vol. 20, no. 1, pp. 2–11, 2018.

[59] X. Xiao, D. Lo, X. Xia, and T. Yuan, "Evaluating defect prediction approaches using a massive set of metrics: an empirical study," in *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pp. 1644–1647, ACM, April 2015.

[60] Q. Qiu and Z. Song, "A nonuniform weighted loss function for imbalanced image classification," in *Proceedings of the 2018 international conference on image and graphics processing*, pp. 78–82, ICIGP, Hong Kong, February 2018.

[61] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[62] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "Knn model-based approach in classification," in *Proceedings of the OTM Confederated International Conferences CoopIS, DOA, and ODBASE 2003 Catania*, pp. 986–996, Springer, Catania, Sicily, Italy, November 2003.

[63] I. Rish et al., "An empirical study of the naive bayes classifier," *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, pp. 41–46, 2001.

[64] J. Tolles and W. J. Meurer, "Logistic regression," *JAMA*, vol. 316, no. 5, pp. 533-534, 2016.

[65] A. Cutler, D. R. Cutler, and J. R. Stevens, "Random forests," in *Ensemble Machine Learning*, pp. 157–175, Springer, 2012.

[66] V. V. Phansalkar and P. S. Sastry, "Analysis of the backpropagation algorithm with momentum," *IEEE Transactions on Neural Networks*, vol. 5, no. 3, pp. 505-506, 1994.

[67] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design (Pws, boston, Ma)*, Google Scholar Google Scholar Digital Library Digital Library, 1996.

[68] C.-C. Yu and B.-Da Liu, "A backpropagation algorithm with adaptive learning rate and momentum coefficient,"vol. 2, pp. 1218–1223, in *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)*, vol. 2, pp. 1218–1223, IEEE, Honolulu, HI, USA, May 2002.

[69] R. Battiti, "First-and second-order methods for learning: between steepest descent and Newton's method," *Neural Computation*, vol. 4, no. 2, pp. 141–166, 1992.

[70] F. D. Foresee and M. T. Hagan, "Gauss-Newton approximation to bayesian learning,"vol. 3, pp. 1930–1935, in *Proceedings of the international conference on neural networks (ICNN'97)*, vol. 3, pp. 1930–1935, IEEE, Houston, TX, USA, June 1997.

[71] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[72] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pp. 65–74, Springer, New Yark, US, 2010.

[73] X.-S. Yang and S. Deb, "Cuckoo search via lévy flights," in *Proceedings of the 2009 World congress on nature & biologically inspired computing (NaBIC)*, pp. 210–214, IEEE, Coimbatore, India, December 2009.

[74] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.