Enhancing Arrhythmia Prediction using the Naked Mole Rat Algorithm and Machine Learning

Nitesh Sureja
Department of Computer Science & Engineering, KSET, KPGU University, Vadodara, India
nmsureja@gmail.com (corresponding author)

Rocky Upadhyay
Department of Computer Science & Engineering, KSET, KPGU University, Vadodara, India
rko75316060@gmail.com

Nandini Chaudhari
Department of Information Technology, KSET, KPGU University, Vadodara, India
nandini113@gmail.com

Shivam Upadhyay
Department of Computer Science & Engineering, KSET, KPGU University, Vadodara, India
shivam91092@gmail.com

Sonia Panesar
Department of Computer Science & Engineering, KSET, KPGU University, Vadodara, India
soniafpanesar@gmail.com

Hemant Patel
Department of Computer Science & Engineering & IT, Dr. Subhash University, Junagadh, India
hemant.patel@dsuni.ac.in

Heli Sureja
Department of Computer Science & Engineering, Gujarat Technological University, Ahmedabad, India
helisureja17@gmail.com

Received: 1 July 2024 | Revised: 28 July 2024 | Accepted: 3 August 2024

 DOI: https://doi.org/10.48084/etasr.8274

ABSTRACT

An irregular heartbeat is known as cardiac arrhythmia. Cardiac arrhythmia occurs when electrical impulses that instruct the heart to beat malfunction. To save lives, arrhythmia must be detected early. Machine Learning (ML) is one of the fastest, cheapest, and easiest disease detection approaches. This study introduces an improved Naked Mole Rat Algorithm (RLNMRA) to select highly influential features in diagnosis and then uses various ML classifiers to train the model. The experimental results showed accuracies of 95.0%, 99.0%, 97.0%, 98.0%, 98.0%, 92.0%, and 99.0% with KNN, RF, GB, SVM, LR, DT, and GNB, classifiers, respectively. These results were obtained after removing invalid features from the UCI Arrhythmia dataset. The proposed method was also evaluated using recall, precision, and F score. The performance of the proposed model indicates that it can be used in various domains.

Keywords-swarm intelligence; machine learning; arrhythmia; naked mole rat algorithm; reinforcement learning

I. INTRODUCTION

Despite significant advances in detection and treatment, cardiac arrhythmias kill one-third of people worldwide, according to the WHO. Cardiac arrhythmia, or abnormal heartbeat, is a medical ailment [1]. Cardiac arrhythmia occurs when electrical heartbeat impulses are misfired. Tachycardia, bradycardia, angina, and palpitations indicate arrhythmia. Normally, early diagnosis and patient survival depend on biomarkers [2].

In recent years, advances in automatic detection using Machine Learning (ML) techniques in diagnosis have been very rapid. With a set of information and features, ML techniques can accurately identify cardiac arrhythmias at a low cost, without several clinical trials. With growing datasets and improved ML algorithms, ML applications have a large impact on automated disease prediction. Various studies have attempted to identify cardiac arrhythmias using various datasets and ML algorithms. This study employs a swarm intelligence algorithm, namely the Naked Mole-Rat Algorithm (NMRA), to select the best features from the UCI Arrhythmia dataset and use it in ML to identify healthy and arrhythmia cases. The advantages of this algorithm are its high convergence speed and very good accuracy in feature extraction.

II. LITERATURE REVIEW

In [5], HHO was employed to reduce dimensionality in detecting arrhythmia, and Random Forest (RF) achieved an accuracy of 96%. In [6], a hybrid of the Harmony Search Algorithm (HSA) and Decision Tree (DT) was proposed to select the best features, achieving 95.25% accuracy with SVM, NB, J48, and MLP models. In [7], a jellyfish model was proposed to reduce the Cleveland dataset's features, achieving 98.4% and 94.4% accuracy with two different SVM models. In [8], the GWO algorithm was used to select the most significant features of a combined dataset, and the VGG-16 model achieved considerable results. In [9], PSO and GA were used to identify and choose the most important features from the Statlog and Cleveland datasets. The GAPSO-RF approach achieved 91.4% and 95.6% accuracy in the Statlog and Cleveland datasets, respectively. In [10], Moth Flame Optimization (MFO) was used to extract important features. Then KNN, SVM, DT, and ANN were used to predict heart disease, achieving 99.2% accuracy. In [11], an improved reptile search algorithm was used, employing opposition-based learning and simulated annealing for feature selection, achieving 70.42% accuracy. In [12], a marine predator algorithm was presented for feature selection problems, achieving excellent results in the COVID-19 dataset. In [13], the marine predator algorithm was combined with a KNN classifier, achieving excellent results in terms of the features selected for each dataset used. In [14], a variant of the Dragonfly Algorithm (DA) was presented, embedding Simulated Annealing (SA) to reduce the dimensionality of the problem and select only relevant features from a dataset. This approach aimed to solve the local optima problem of DA and improve its feature subset selection. This model achieved 90% accuracy in the Cleveland Heart dataset. In [15], a binary social spider algorithm was introduced for feature selection problems. This approach was tested against 21 UCI datasets using KNN and SVM classifiers, offering qualitative and stable solutions with superior performance. In [16], a fuzzy-rough-based binary shuffled frog-leaping algorithm was introduced for feature selection, achieving excellent results.

The contributions of this study are as follows:

- Introduces a swarm intelligence ML approach to predict cardiac arrhythmias.
- Introduces RLNMRA, an improved NMRA algorithm, to select significant features and produce a subset of the arrhythmia dataset.
- Introduces reinforcement learning to enhance solution diversity and prevent the local optima problem of NMRA.
- Uses seven ML classifiers, including KNN, RF, GB, DT, LR, GNB, and SVM, to predict (classify) arrhythmia in a subset of features generated by RLNMRA.

III. METHOD

A. Basics and Background

NMRA is a swarm-inspired optimization algorithm that draws inspiration from the social behavior and foraging strategies of naked mole rats [17]. It aims to solve optimization problems by mimicking cooperative and competitive interactions between these rodents. NMRA can be structured into three main stages, each reflecting different aspects of the social and foraging behavior of naked mole rats. The basic idea for selecting NMRA is based on that it uses a very small number of parameters, is easy to implement, is not much used, and can be easily integrated with any enhancing techniques.

1) Initialization

Initially, a uniform random population of n naked mole rats (solutions) is generated. Each naked mole rat in the population is a D-dimensional vector in the range [1, 2 … n]. Here, D is the number of parameters or variables to be tested in the problem. Each NMR (solution) is represented as

$$SL_{i,j} = SL_{min,j} + UR(0,1) \times (SL_{min,j} - SL_{max,j}) \quad (1)$$

where $i \in [1, 2, \ldots n]$, $j \in [1,2,\ldots, D]$, and $SL_{i,j}$ represents the i^{th} solution in the j^{th} dimension. $SL_{max,j}$ and $SL_{min,j}$ represent the upper and lower bounds of the problem function, respectively. $UR(0,1)$ is a uniformly distributed random number. The fitness of NMRs is computed by evaluating the objective function after initialization. Based on the fitness of all NMRs, the classes of breeders (*BR*) and workers (*WR*) are defined. An overall initial fittest solution d is also calculated. After initialization of the NMR population in the worker and breeder classes, the search process is performed through repeated iterations in both the worker and breeder phases.

2) Worker Phase

After initialization, the second step is the exploration operation, which is governed by the worker phase of the NMRA algorithm. In this phase, workers try to improve their fitness to get a chance to become breeders and finally mate with the queen. During this phase, NMRs create a new solution

based on local information and their own experiences. The new solution is evaluated and compared with the old one to determine its fitness. The better-performing solution is retained and stored. Once all workers have successfully completed the search process, their fitness is stored. Then, the NMRA uses (2) to generate a new solution from the old one.

$$WR_i^{t+1} = WR_i^{t+1} + \lambda\left(WR_j^t + WR_k^t\right) \quad (2)$$

where WR_i^t represents the i^{th} worker in the t^{th} iteration, WR_i^{t+1} represents the new worker or solution, λ represents the mating factor, and WR_k^t and WR_j^t are two random solutions selected from the workers. The value of λ ranges from 0 to 1.

3) Breeder Phase

This is the third phase of the NMRA algorithm and is mainly meant for exploitation operations. Solutions are generated with respect to the best current solution and are controlled by the mating factor (λ). The NMRs in the breeder class strive to enhance their abilities to increase their chances of being selected for mating and to remain in the class. The NMRs in the breeder class update themselves based on a probability (*brp*) pertaining to the overall fittest distribution, which is a random number between 0 and 1. Breeders with less fitness are re-assigned to worker classes. Breeders update their positions using

$$BR_i^{t+1} = (1 - \lambda)BR_i^t + \lambda(d - BR_i^t) \quad (3)$$

where BR_i^t represents the i^{th} breeder in the t^{th} iteration. The mating frequency of the breeders is controlled by the λ factor, which supports recognizing a new breeder BR_i^{t+1} in the next iteration. Initially, the value of *brp* is set to 0.5. For simplicity, only one queen is kept in the colony. The task is to identify the most suitable breeder to mate with the queen. NMRA identifies the workers and breeders among the NMR colony. The initial evaluation identifies the fittest worker and breeder. In the end, workers' fitness is updated so that they have a chance to become breeders. Based on *brp*, breeders also update their fitness to survive as breeders. All sterile breeders are assigned to the worker class. The most suitable breeder from the pool serves as a potential solution to the problem under investigation.

Reinforcement Learning (RL) has been employed in various fields to address challenges [18]. The core concept of RL entails an agent executing an action that modifies the state of the environment and subsequently receives a reward contingent upon that action. RL can be classified into two distinct categories: policy- and value-based models. *Q*-Learning (QL) is a type of value-based algorithm [19]. In the learning phase, the agent performs the action that has the highest anticipated Ql value. The most basic version of QL, known as one-step QL, updates the *Ql* value based on the state-action pair in a single step. This study uses the one-step QL approach. Equation (4) continuously updates the *Q* table based on the reward associated with each state-action pair. The symbols *Lr* and γ represent the learning rate and the discount factor, respectively, which are both between 0 and 1. $Ql(st_t, at_t)$ represents the Ql value obtained by taking action at_t in the current state st_t, while $max_{at}Q(st_{t+1}, a_{tt+1})$ represents the maximum expected Ql value in the Q table when executing the action at_{t+1} in state st_{t+1}. It is important to note that a higher learning rate *Lr* causes the algorithm to learn from the expected Ql value, while a lower learning rate causes the algorithm to exploit the prior Ql value. Thus, the learning rate serves as a means to achieve a trade-off between exploiting existing knowledge and exploring new possibilities.

$$Ql(st_t, at_t) = (1 - Lr)Ql(st_t, at_t) + Lr(rs_{t+1} + \gamma\, max_{at}Ql(st_{t+1}, at_{t+1})) \quad (4)$$

B. Proposed RLNMRA

1) Motivation for the Proposed Model

Basic NMRA employs two approaches to discover solutions: exploration and exploitation. NMRA's inability to alter course during repetitions is an inherent constraint, making it vulnerable to becoming stuck in local optima. The adaptive search operation is used, which is more advantageous in acquiring the global minimum, and RL is implemented to efficiently achieve this objective. Random opposition-based learning is used to enhance the search for alternative solutions, thus increasing the diversity within the population.

2) RLNMRA Structure

In RLNMRA, the search space is considered the environment, while each individual is seen as the training agent of RL. QL is employed to dynamically alternate between exploration and exploitation. QL updates the Q value associated with a state-action pair considering the highest fitness value observed so far and the average fitness value from previous iterations. A reward table is employed to provide incentives or penalties to the agent, based on its current action and status. The proposed NMRA with RL consists of three actions that are determined by the value of the exploration rate (Φ): increasing, decreasing, and maintaining the exploration rate. The Φ value in the subsequent iteration is modified according to the highest fitness in the current iteration and the cumulative average fitness value using (5). Here, Φ^{it+1} represents the exploration rate at the next iteration, Δ is the incremental value, and $f(X_{best}^{it})$ represents the fitness of the best position in the current iteration. M represents the mean fitness of the fit solutions discovered so far, calculated using (6). Here, n is the number of iterations that have been carried out until the current one, and wf_i is the weighted factor for the fittest individual X_{best}^{it} at iteration it, calculated using the $wf_i = e^{it/IT}$ formula.

$$\Phi^{it+1} = \begin{cases} \Phi^{it} * (1 + \Delta) & if\ f(X_{best)}^{it} > M \\ \Phi^{it} * (1 - \Delta) & if\ f(X_{best)}^{it} < M \\ \Phi^{it} & otherwise \end{cases} \quad (5)$$

$$M = \frac{1}{n}\sum_{it=1}^{n} w_i X_{best}^{it} \quad (6)$$

In this process, it represents the current iteration, and *IT* represents the total number of iterations. It is important to note that the most physically fit NMRs (solutions) have a greater impact on the calculation of the *M* value. Specifically, if the fitness achieved is higher than the average fitness, the algorithm should focus on a smaller search space and improve

the acquired solutions. Alternatively, the algorithm expands its search region to discover novel solutions and prevent getting stuck in local optima. In summary, the first scenario described in (7) occurs when the agent achieves a higher level of fitness than the average. In the second situation, the agent's fitness starts to decline compared to the prior agent's experience. The RLNMRA has three states, denoted as $st = \{1, -1, 0)$ which correspond to the activities described in (7). The reward table assigns a positive value of $+1$ to the state $st_t = 1$ and a negative value of -1 to all other states. If the fitness gained in iteration it, is better than the average fitness of the last $it - 1$ iterations, then the current state st_t is equal to 1. Equation (8) illustrates the reward approach, where st_t is the state obtained by the agent at iteration *it*.

The proposed RLNMRA model adjusts the learning rate according to the accumulated performance, as this factor greatly influences the attainment of the solution. When the learning rate is near one, the newly acquired information significantly influences the future reward. At a low learning rate, the value of existing information surpasses that of newly acquired information. To optimize the outcome, the learning rate is dynamically decreased at each iteration using (9). Here, Lr_{init} and Lr_{final} represent the starting and final values of the learning rate, respectively.

$$st_t = sign(f(x^{it}) - M), sign(x) = \begin{cases} 1 & if\ x > 1 \\ -1 & if\ x < 1 \\ 0 & otherwise \end{cases} \quad (7)$$

$$Reward = \begin{cases} +1 & if\ st_t = 1 \\ -1 & otherwise \end{cases} \quad (8)$$

$$Lr = \frac{Lr_{init} + Lr_{final}}{2} - \frac{Lr_{init} + Lr_{final}}{2} \cdot \cos\left(\pi\left(1 - \frac{it}{IT}\right)\right)) \quad (9)$$

The Random Opposition-Based Learning (ROBL) technique is integrated into the RLNMRA algorithm to assist in avoiding the problem of local optima [20]. ROBL uses randomization to enhance the performance of OBL models, defined as:

$$x_{ij}^{l} = lb_{ij} + ub_{ij} - rand \times x_{ij},$$
$$ij = 1,2, \ldots\ldots\ldots, n \quad (10)$$

where x_{ij}^{l} and x_{ij} denote the antithetical and initial solutions, and lb_j and ub_j represent the minimum and maximum limits of the variables.

IV. EXPERIMENTS AND RESULTS

A. Dataset

The UCI Arrhythmia dataset [21] was used, containing 279 attributes, 206 of which are linearly valued and the rest are nominal. The aim is to distinguish between the presence and absence of cardiac arrhythmia and to classify it in one of the 16 classes. Class 01 refers to normal, ECG classes 02 to 15 refer to different classes of arrhythmia, and class 16 refers to unclassified.

B. Evaluation Measures

According to [22], F-measure (*FM*), recall (*RC*), accuracy (*ACC*), and precision (*PR*) metrics are used for evaluation, calculated using (11)-(14). True Positive (*TP*) represents patients correctly classified with arrhythmia. True Negative (*TN*) presents patients accurately recognized without arrhythmia. False Positive (*FP*) presents misclassified healthy individuals as patients, and False Negative (*FN*) presents the number of individuals misdiagnosed as healthy.

Algorithm: Proposed NMRA
Inputs: Initialize population of NMRs: nr, Breeders (BR): nr/5, Workers (WR): nr−BR, probability breeding: brp, objective function, f(x), Set the state st= (st1, st2, st3} and action at = (at1, at2, at3). Initialize Q table and reward table, select present state randomly
Output: Find the overall fittest solution (fx)
Find the current best solution
Calculate the exploration rate ∅
Do Until iteration < maximum number of iterations
For i=1: N
For j=1: WR
If rand < ∅
Perform worker phase using (2)
Evaluate the identified worker
Else
For j=1:BR
If rand < 0.5
If UR (0, 1) > brp
Perform the breeder phase using (3)
Evaluate the identified breeder
End If
End If
End for
Compute $x_{best}^{l}(it+1)$ using ROBL
End For
Combine the newly found breeder and worker population
Find the best $x_{best}(it+1)$ by evaluating the new population
If $f(x_{best}^{l}(it+1))$ is better than $f(x_{best}^{l}(it+1))$
$x_{best}(it+1) = x_{best}^{l}(it+1)$
Update the current state using (7)
Calculate the reward using (8)
Update Q-table using (4)
End for

```
Update iteration count
Update the learning rate using (9)
End until
Save the final fittest solution (fx)
End
```

$$FM = \frac{(2*PR*RC)}{(PR+RC)} * 100 \quad (11)$$

$$RC = \frac{TP}{TP+FN} * 100 \quad (12)$$

$$ACC = \frac{TP+TN}{TP+FN+FP+TN} \times 100 \quad (13)$$

$$PR = \frac{TP}{TP+FP} * 100 \quad (14)$$

A PC with an Intel i3 processor and 8 GB of RAM running Windows 10 64-bit was used for the experiments. Python was used to implement the models. Table I shows a comparative assessment of traditional classifiers without any improvements for predicting arrhythmia. The experimental values indicate that the RF classifier performed well with an accuracy of 92.64%. The RF classifier outperformed other classifiers with values of 92.87%, 92.55%, and 92.64% for precision, recall, and F-measure, respectively. Table II shows a comparison of the classifiers using iHHO [5] to predict arrhythmia. The RF classifier achieved an accuracy of 97.0%, again outperforming other classifiers with 99.07%, 95.13%, and 96.99% for precision, recall, and F-measure, respectively. Table III shows a comparative evaluation of classifiers using the proposed model for predicting arrhythmia. The experimental values indicated that the RF and GNB classifiers both performed well, with an accuracy of 99.00% with RLNMRA. SVM and LR followed, with accuracies of 98.00% each. GB, KNN, and DT achieved accuracies of 97.00%, 96.00%, and 92.00%, respectively. The RF classifier also outperformed the others with 99.71%, 98.61%, and 99.15% precision, recall, and F-measure, respectively. Figures 1-7 show the confusion matrices of different classifiers. Figures 8-10 show the results graphically. In addition, the accuracy results of RLNMRA were compared with the RSA [11], IRSA [11], PSO [11], GA [11], GOA [11], SMO [11], BDA [23], IBDA [24], BGWO [25], BPSO, IBPSO [26], and BBA [26], based on the same dataset. Figure 11 shows a graphical comparison of their results. These results, along with Tables I-IV, demonstrate that the proposed algorithm outperformed the other algorithms by a good margin within the same dataset.

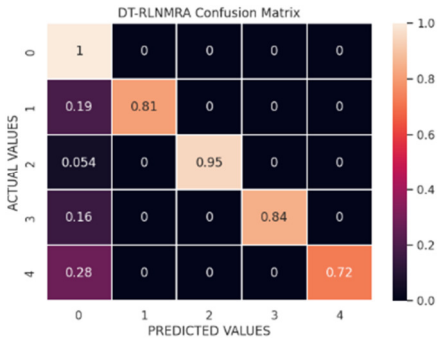

Fig. 1. DT-RLNMRA confusion matrix.

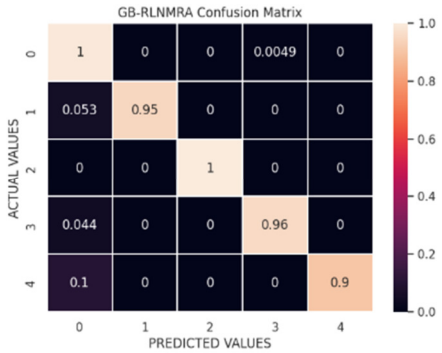

Fig. 2. GB-RLNMRA confusion matrix.

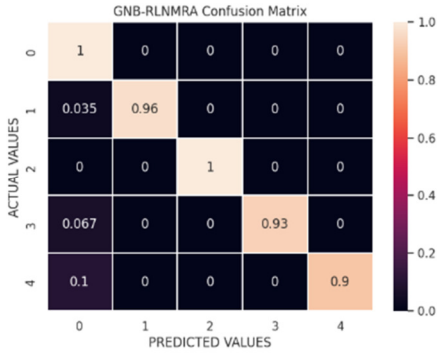

Fig. 3. GNB-RLNMRA confusion matrix.

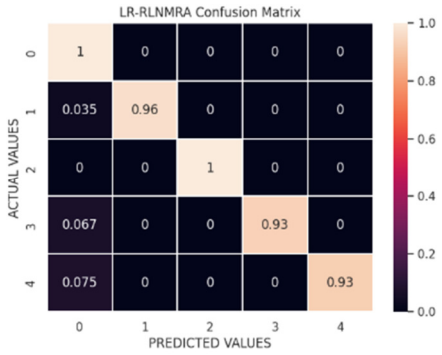

Fig. 4. LR-RLNMRA confusion matrix.

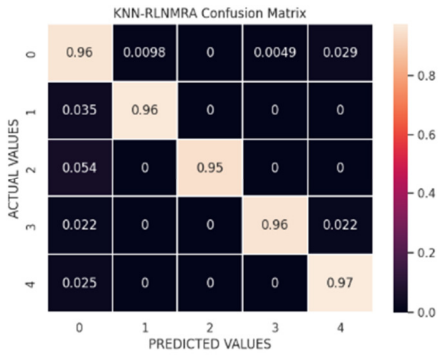

Fig. 5. KNN-RLNMRA confusion matrix.

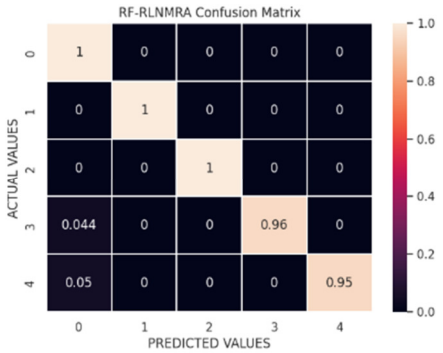

Fig. 6. RF-RLNMRA confusion matrix.

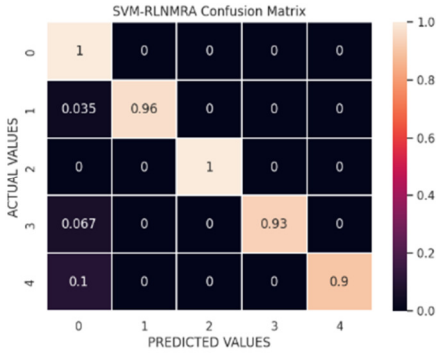

Fig. 7. SVM-RLNMRA confusion matrix.

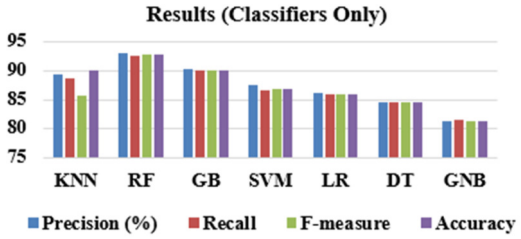

Fig. 8. Results attained with only classifiers.

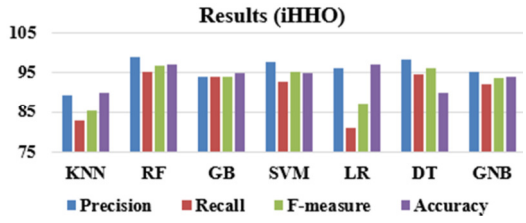

Fig. 9. Results attained with iHHO and classifiers.

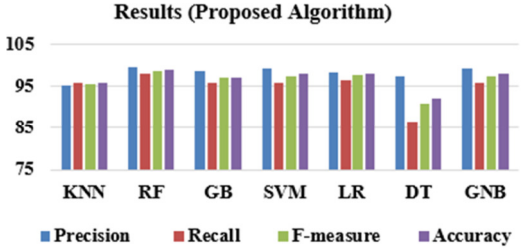

Fig. 10. Results attained with RLNMRA and classifiers.

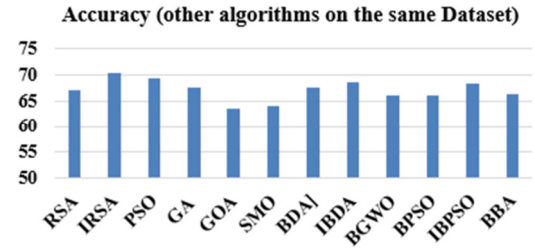

Fig. 11. Accuracies achieved by other algorithms on the same dataset.

TABLE I. RESULTS ATTAINED WITH CLASSIFIERS ONLY

Model	PR (%)	RC (%)	FM (%)	ACC (%)
KNN	89.27	88.64	85.72	89.92
RF	92.87	92.55	92.64	92.64
GB	90.27	89.99	90.07	90.07
SVM	87.54	86.67	86.81	86.81
LR	86.22	85.95	86.02	86.02
DT	84.63	84.56	84.59	84.59
GNB	81.27	81.32	81.21	81.21

TABLE II. RESULTS ATTAINED WITH IHHO

Model	PR (%)	RC (%)	FM (%)	ACC (%)
KNN	89.49	82.74	85.35	90.00
RF	99.07	95.13	96.99	97.00
GB	94.02	94.06	93.99	95.00
SVM	97.80	92.80	95.40	95.00
LR	96.16	81.12	87.15	96.00
DT	98.45	94.52	96.30	89.87
GNB	95.2	92.2	93.6	94.00

TABLE III. RESULTS ATTAINED WITH THE PROPOSED ALGORITHM

Model	PR (%)	RC (%)	FM (%)	ACC (%)
KNN	95.21	95.95	95.47	96.00
RF	99.61	98.11	98.83	99.00
GB	98.7	95.96	97.25	97.00
SVM	99.15	95.96	97.47	98.00
LR	98.24	96.46	97.79	98.00
DT	97.37	86.44	91.02	92.00
GNB	99.15	95.96	97.47	99.00

TABLE IV. ACCURACIES ATTAINED WITH OTHER ALGORITHMS

Model	ACC (%)	Model	ACC (%)
RSA [11]	66.98	BDA [23]	67.61
IRSA [11]	70.42	IBDA [23]	68.57
PSO [11]	69.30	BGWO [23, 24]	66.07
GA [11]	67.50	BPSO [23]	66.10
GOA [11]	63.58	IBPSO [23, 25]	68.26
SMO [11]	64.10	BBA [23, 26]	66.22

V. CONCLUSION

This study introduced an algorithm to accurately predict arrhythmias, which works in two stages. In the first stage, an improved naked mole algorithm (RLNMRA) was used to select the significant features from the arrhythmia dataset. Following feature selection, seven different ML classifiers were applied to the newly generated dataset to predict cardiac arrhythmia. The NMRA algorithm was enhanced by integrating RL. RL was used to enhance population diversity and convergence speed

and avoid the local optima problem. The experimental evaluation demonstrated the superiority of the proposed algorithm in predicting arrhythmia in all metrics used for comparison. Furthermore, the results show that the NMRA with RL significantly improved the performance of the KNN, RF, GB, SVM, LR, DT, and GNB classifiers. Future research could investigate the performance of the proposed algorithm on more sophisticated science and engineering problems and further enhance its complexity without affecting its current performance.

REFERENCES

[1] J. Wardrope, "Clinical ECG interpretation--an introduction.," *Emergency Medicine Journal*, vol. 16, no. 3, pp. 201–201, May 1999, https://doi.org/10.1136/emj.16.3.201.

[2] E. Lee and A. Moon, "Identification of Biomarkers for Breast Cancer Using Databases," *Journal of Cancer Prevention*, vol. 21, no. 4, pp. 235–242, Dec. 2016, https://doi.org/10.15430/JCP.2016.21.4.235.

[3] M. Salehi, J. Razmara, and S. Lotfi, "A Novel Data Mining on Breast Cancer Survivability Using MLP Ensemble Learners," *The Computer Journal*, vol. 63, no. 3, pp. 435–447, Mar. 2020, https://doi.org/10.1093/comjnl/bxz051.

[4] O. O. Akinola, A. E. Ezugwu, J. O. Agushaka, R. A. Zitar, and L. Abualigah, "Multiclass feature selection with metaheuristic optimization algorithms: a review," *Neural Computing and Applications*, vol. 34, no. 22, pp. 19751–19790, Nov. 2022, https://doi.org/10.1007/s00521-022-07705-4.

[5] E. Al. Nitesh Sureja, "Predicting Arrhythmia Based on Machine Learning Using Improved Harris Hawk Algorithm," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 10, pp. 862–870, Nov. 2023, https://doi.org/10.17762/ijritcc.v11i10.8603.

[6] M. Z. Gashti, "Detection of Spam Email by Combining Harmony Search Algorithm and Decision Tree," *Engineering, Technology & Applied Science Research*, vol. 7, no. 3, pp. 1713–1718, Jun. 2017, https://doi.org/10.48084/etasr.1171.

[7] A. A. Ahmad and H. Polat, "Prediction of Heart Disease Based on Machine Learning Using Jellyfish Optimization Algorithm," *Diagnostics*, vol. 13, no. 14, Jan. 2023, Art. no. 2392, https://doi.org/10.3390/diagnostics13142392.

[8] D. Santoshi, S. Chaudhari, and N. Pulgam, "Advanced Approach for Heart Disease Diagnosis with Grey Wolf Optimization and Deep Learning Techniques," in *Data Intelligence and Cognitive Informatics*, Tirunelveli, India, 2023, pp. 631–647, https://doi.org/10.1007/978-981-19-6004-8_49.

[9] M. G. El-Shafiey, A. Hagag, E. S. A. El-Dahshan, and M. A. Ismail, "A hybrid GA and PSO optimized approach for heart-disease prediction based on random forest," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 18155–18179, May 2022, https://doi.org/10.1007/s11042-022-12425-x.

[10] S. Haseena, S. K. Priya, S. Saroja, R. Madavan, M. Muhibbullah, and U. Subramaniam, "Moth-Flame Optimization for Early Prediction of Heart Diseases," *Computational and Mathematical Methods in Medicine*, vol. 2022, no. 1, 2022, Art. no. 9178302, https://doi.org/10.1155/2022/9178302.

[11] Z. Elgamal, A. Q. M. Sabri, M. Tubishat, D. Tbaishat, S. N. Makhadmeh, and O. A. Alomari, "Improved Reptile Search Optimization Algorithm Using Chaotic Map and Simulated Annealing for Feature Selection in Medical Field," *IEEE Access*, vol. 10, pp. 51428–51446, 2022, https://doi.org/10.1109/ACCESS.2022.3174854.

[12] Z. Beheshti, "BMPA-TVSinV: A Binary Marine Predators Algorithm using time-varying sine and V-shaped transfer functions for wrapper-based feature selection," *Knowledge-Based Systems*, vol. 252, Sep. 2022, Art. no. 109446, https://doi.org/10.1016/j.knosys.2022.109446.

[13] D. S. A. Elminaam, A. Nabil, S. A. Ibraheem, and E. H. Houssein, "An Efficient Marine Predators Algorithm for Feature Selection," *IEEE Access*, vol. 9, pp. 60136–60153, 2021, https://doi.org/10.1109/ACCESS.2021.3073261.

[14] H. Chantar, M. Tubishat, M. Essgaer, and S. Mirjalili, "Hybrid Binary Dragonfly Algorithm with Simulated Annealing for Feature Selection," *SN Computer Science*, vol. 2, no. 4, May 2021, Art. no. 295, https://doi.org/10.1007/s42979-021-00687-5.

[15] E. Baş and E. Ülker, "An efficient binary social spider algorithm for feature selection problem," *Expert Systems with Applications*, vol. 146, May 2020, Art. no. 113185, https://doi.org/10.1016/j.eswa.2020.113185.

[16] J. R. Anaraki, S. Samet, M. Eftekhari, and C. W. Ahn, "A Fuzzy-Rough Feature Selection Based On Binary Shuffled Frog Leaping Algorithm," *International Journal of Information, Control and Computer Sciences*, vol. 11, no. 9, Aug. 2018, https://doi.org/10.5281/ZENODO.1474575.

[17] R. Salgotra and U. Singh, "The naked mole-rat algorithm," *Neural Computing and Applications*, vol. 31, no. 12, pp. 8837–8857, Dec. 2019, https://doi.org/10.1007/s00521-019-04464-7.

[18] K. Sivamayil, E. Rajasekar, B. Aljafari, S. Nikolovski, S. Vairavasundaram, and I. Vairavasundaram, "A Systematic Study on Reinforcement Learning Based Applications," *Energies*, vol. 16, no. 3, Jan. 2023, Art. no. 1512, https://doi.org/10.3390/en16031512.

[19] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, May 1992, https://doi.org/10.1007/BF00992698.

[20] H. R. Tizhoosh, "Opposition-Based Learning: A New Scheme for Machine Intelligence," in International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria, Nov. 2005, vol. 1, pp. 695–701, https://doi.org/10.1109/CIMCA.2005.1631345.

[21] B. A. H. Guvenir, "Arrhythmia." UCI Machine Learning Repository, 1997, https://doi.org/10.24432/C5BS32.

[22] M. A. Karaolis, J. A. Moutiris, D. Hadjipanayi, and C. S. Pattichis, "Assessment of the Risk Factors of Coronary Heart Events Based on Data Mining With Decision Trees," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 3, pp. 559–566, Feb. 2010, https://doi.org/10.1109/TITB.2009.2038906.

[23] J. Li *et al.*, "IBDA: Improved Binary Dragonfly Algorithm With Evolutionary Population Dynamics and Adaptive Crossover for Feature Selection," *IEEE Access*, vol. 8, pp. 108032–108051, 2020, https://doi.org/10.1109/ACCESS.2020.3001204.

[24] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, Jan. 2016, https://doi.org/10.1016/j.neucom.2015.06.083.

[25] B. Ji, X. Lu, G. Sun, W. Zhang, J. Li, and Y. Xiao, "Bio-Inspired Feature Selection: An Improved Binary Particle Swarm Optimization Approach," *IEEE Access*, vol. 8, pp. 85989–86002, 2020, https://doi.org/10.1109/ACCESS.2020.2992752.

[26] J. Feng, H. Kuang, and L. Zhang, "EBBA: An Enhanced Binary Bat Algorithm Integrated with Chaos Theory and Lévy Flight for Feature Selection," *Future Internet*, vol. 14, no. 6, Jun. 2022, Art. no. 178, https://doi.org/10.3390/fi14060178.