ORIGINAL ARTICLE

A hybrid artificial bee colony with whale optimization algorithm for improved breast cancer diagnosis

Punitha Stephan[1] · Thompson Stephan[2] · Ramani Kannan[3] · Ajith Abraham[4]

Abstract
Breast cancer is the most common among women that leads to death if not diagnosed at early stages. Early diagnosis plays a vital role in decreasing the mortality rate globally. Manual methods for diagnosing breast cancers suffer from human errors and inaccuracy, and consume time. A computer-aided diagnosis (CAD) can overcome the disadvantages of manual methods and helps radiologists for accurate decision-making. A CAD system based on artificial neural network (ANN) optimized using a swarm-based approach can improve the accuracy of breast cancer diagnosis due to its strong prediction capabilities. Artificial bee colony (ABC) and whale optimization are metaheuristic search algorithms used to solve combinatorial optimization problems. This paper proposes a hybrid artificial bee colony with whale optimization algorithm (HAW) by integrating the exploitative employee bee phase of ABC with the bubble net attacking method of whale optimization to propose an employee bee attacking phase. In the employee bee attacking phase, employee bees use exploitation of humpback whales for finding better food source positions. The weak exploration of standard ABC is improved using the proposed mutative initialization phase that forms the explorative phase of the HAW algorithm. HAW algorithm is used in simultaneous feature selection (FS) and parameter optimization of an ANN model. HAW is implemented using backpropagation learning that includes resilient backpropagation (HAW-RP), Levenberg–Marquart (HAW-LM) and momentum-based gradient descent (HAW-GD). These hybrid variants are evaluated using various breast cancer datasets in terms of accuracy, complexity and computational time. HAW-RP variant achieved higher accuracy of 99.2%, 98.5%, 96.3%, 98.8%, 98.7% and 99.1% with low-complexity ANN model when compared to HAW-LM and HAW-GD for WBCD, WDBC, WPBC, DDSM, MIAS and INbreast, respectively.

1 Introduction

Early-stage diagnosis plays a major role in increasing the chance of recovery from breast cancer. World Health Organization (WHO) has reported that WHO estimates that there may be an increase in the cancer incidence of 27.5 million in the year of 2040, with 16.3 million deaths due to cancer [1]. Currently, the average risk of a woman in the USA developing breast cancer sometime in her life is about 13%. According to the American Cancer Society, in the USA in the year 2021, invasive breast cancer expected is 281,550 of new cases, and about 43,600 women are estimated to die because of breast cancer [2]. In metropolitan cities in India such as Mumbai, Chennai, Delhi, Bangalore, Ahmadabad and Bhopal, noninvasive breast cancer has

✉ Thompson Stephan
thompsoncse@gmail.com

1 Department of Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India

2 Department of Computer Science and Engineering, Faculty of Engineering and Technology, M. S. Ramaiah University of Applied Sciences, Bangalore, Karnataka, India

3 Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS, Seri Iskandar, Malaysia

4 Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, Washington, USA

affected 28% to 35% of the women population [3]. Thus, breast cancer has become a serious health issue around the globe and early detection is essential in reducing life fatalities [4]. Early detection can be done using various scanning methods such as magnetic resource imaging, ultrasound imaging, self-check-up, mammography and biopsies [5]. Traditionally followed breast cancer methods consume more time for diagnosis and they fail because of inaccurate diagnosis caused by human errors. Automated computer-based diagnosis schemes overcome the demerits of manual diagnosis, and hence, unnecessary surgeries and biopsies can be avoided [6]. Expert systems based on ANN have strong predictive capabilities which makes it suitable for building medical diagnosis systems [7]. ANN-based decision-making systems have outperformed the traditional technique used for classifying patterns.

Metaheuristic-based swarm intelligence approach is used for real-time optimization problem-solving [8–10]. Commonly used swarm intelligence approaches are the ant colony optimization (ACO) [11] and the particle swarm optimization (PSO) [12] inspired by the foraging behavior of ants and social behavior of birds, respectively. The echolocation capability of the microbats available in nature forms the basis of the bat algorithm (BA) [13]. A population-based swarm technique introduced based on the foraging behavior of honey bees [14]. The dynamic and static behavior of dragonflies forms the basis of a new metaheuristic algorithm called dragonfly algorithm (DA) [15]. Based on the herding behavior of krill, another swarm technique is proposed called krill herd (KH) algorithm [16].

A technique based on migration behavior, called the monarch butterfly optimization (MBO), is introduced [17]. The foraging behavior for the survival of E. coli bacteria forms the basis of the bacterial foraging optimization (BFO) [18]. Another swarm technique called the artificial immune system (AIS) is inspired by the biological immune system of the human body [19]. An algorithm for global optimization based on interior design and decoration [20]. A salp swarm algorithm (SSA) based on the swarming behavior of the salp in the ocean is introduced that can be used to solve multidimensional optimization problems [21]. Based on the Brownian movements and Levy movements of the predators during their foraging process, another swarm technique called marine predictor algorithm (MPA) was proposed [22]. This paper focused on hybridizing artificial bee colony optimization with the whale optimization algorithm to introduce the HAW algorithm. The proposed HAW algorithm integrates the employee bee phase of the ABC with the encircling prey/bubble net attacking method to have an enhanced exploitative phase called the employee attacking phase. In the employee attacking phase, the bees follow the bubble

net attacking method of the whales to find out better food sources. The explorative phase of the HAW is driven by a mutative initialization phase of the standard ABC algorithm.

Appropriate selection of ANN topology design parameters such as the number of hidden layers, numbers of hidden nodes, initial weight values between the connections, learning rate and algorithm plays a vital role in building a successful ANN model [23]. The convergence of the backpropagation learning process can be affected by the improper selection of weights making the learning process to be trapped in the local optimal locations [24, 25]. Improper selection of the hidden nodes may make the ANN classifier to deal with the problems of under fitting and over fitting. If the usage of hidden nodes in an ANN model is not appropriate to the amount of learning required for accurate diagnosis, then the ANN classifier may be either overtrained where the ANN model can give accurate results in case of training and fails with inaccurate results in the case of testing or undertrained where the prediction rate decreases. Based on the above discussion, this paper focuses on optimal selection of the value of initial weights and the optimal selection of the hidden node sizes of an ANN model using HAW algorithm with the help of a wrapper architecture such that work aims at improving the learning performance of an ANN avoiding the problems of overfitting and underfitting with increased predictive capabilities.

FS deals with the deletion of irrelevant, redundant and noisy features present in the input dataset of a classifier. FS improves the generalization of an ANN classifier system with reduced computational time, as demonstrated in [26, 27]. Hence, simultaneously optimizing the input features and design parameters of ANN such as the initial weights and hidden node size can increase the predictability of the ANN classifier. Swarm-based intelligent systems are used for coupled optimization of input features and ANN design parameters [28]. Due to the importance of simultaneous optimization of ANN design parameters and FS process, the ANN topology optimization can be coupled with ABC optimization due to its powerful local and global search capabilities in finding out global optimal solutions.

This paper focuses on the following objectives:

(i) A hybrid ABC-WOA optimization (HAW) that integrates the encircling prey and the bubble net attacking method of WOA with the employee bee phase of standard ABC to form an employee attacking phase.

(ii) To make HAW to escape from local optimum locations, the proposed employee attaching phase uses the simulated annealing technique.

(iii) To get a diversified set of solutions, exploration of the HAW is enhanced using the proposed mutative exploration phase of ABC.

(iv) HAW is implemented for optimal feature subset selection and ANN parameter optimization using Wisconsin breast cancer dataset. HAW-optimized ANN model is evaluated in terms of accuracy, complexity and computational time.

1.1 Artificial bee colony (ABC) optimization

A metaheuristic swarm-based search mechanism called ABC is introduced by Karaboga in 2005. It is a population-based approach, inspired by the foraging nature of honey bees that solves multidimensional and multimodal real-time optimization problems for different applications, as demonstrated in [29]. ABC is based on a stochastic process that is robust and highly flexible with a lesser number of control parameters that make it simple. The algorithmic steps of the ABC optimization process are described in Algorithm (1):

Algorithm 1: Artificial bee colony algorithm
Step 1: Initialization:

Food sources are randomly produced using Equation (1).

$$A_k^l = A_k^l + random(0, 1) * \left(A_{max}^l - A_{min}^l\right) \tag{1}$$

A_k^l represents k^{th} food source with lth parameter and $j = 1, 2.........N$, in which N represents maximum food sources. $l = 1, 2......dim$, in which 'dim' represents the dimension representing the number of parameters in the optimization problem. A_{max}^l and A_{min}^l are the minimum and maximum bound of the lth parameter of the optimization problem, respectively.

Step 2: **Quality Evaluation of food source:**

The fitness values are identified for each food source A_k.

Step 3: Employed bee Phase:

Food sources are assigned to employee bees or worker bees. The employee bees use Eq. (2) to search neighborhood food sources surrounding the current food sources A_k^l.

$$E_k^l = A_k^l + random[-1, 1] * \left(A_k^l - A_d^l\right) \tag{2}$$

A_d is a random food source where $d \in \{1, 2..., N\}$. 'l' is a random integer, and $i = \{1, 2..., dim\}$ and 'd' should not be equal to 'l' for proper exploitation. If the quality

of E_k^l is greater than A_k^l, then bee discards A_k^l saving E_k^l or vice versa.

Step 4: **Onlooker Bee Phase:**

Information regarding the selected food sources is shared with the onlooker bees. The probability value Z_k of each food source received from the employee bee is calculated using Eq. (3).

$$Z_k = \frac{fitness(A_k)}{\sum_{k=1}^{N} fitness(A_k)} \tag{3}$$

The quality of the food source A_k is represented as $fitness(A_k)$. The value Z_k of food source is compared with a $random(0, 1)$. Food sources with a Z_k value greater than $random(0, 1)$ are selected by the onlooker bees.

Step 5: **Food source memorization:**

The food source with the highest $fitness(A_k)$ is selected and memorized.

Step 6: **Scout bee phase:**

In the scout bee phase, unimproved food sources are identified based on a counter value and they are replaced by a randomly generated food source according to Equation (1).

1.2 Whale optimization algorithm (WOA)

WOA is a population-based swarm intelligence meta-heuristic algorithm introduced by Mirjalili and Lewis [30] which is inspired by the foraging behavior of humpback whales. The humpback whales' hunts group of krill or fishes using shrinking circle and producing bubbles in a circle '9'-shaped path. The exploitation phase is carried out using encircling prey and bubble net attacking based on the spiral. A random search of prey is used for exploration. The exploitation phase of WOA is explained in Algorithm (2).

Algorithm 2: Encircling prey/bubble net attacking of WOA

To hunt the prey for survival, humpback whales encircle around the prey which can be mathematically represented using Eqs. (4) and (5).

$$Y = \left|\vec{L}.A^*(t) - A(t)\right| \tag{4}$$

$$\vec{A}(t+1) = \overrightarrow{A^*}(t) - \vec{M} * Y \qquad (5)$$

where t represents the current iteration, A^* represents the best solution found so far, A gives the position vector, $| \ |$ represents the absolute value, and L and M represents the coefficient vectors that can be obtained using Eqs. (6 and 7).

$$\vec{M} = 2\vec{m} * \vec{r} - \vec{m} \qquad (6)$$

$$\vec{L} = 2 * \vec{r} \qquad (7)$$

where m is reduced linearly starting from 2 till 0 as the iteration proceeds. r represents a random vector from a uniform distribution between [0,1]. Each whale that represents a solution updates its position using Eq. (5) where the updated new position of the whale depends on the best position (prey) found so far. The position of the whales can be controlled by the adjustment of vectors L and M. The value of m is decreased to achieve the shrinking encircling behavior using Eq. (8).

$$m = 2 - t\frac{2}{MaxIterat} \qquad (8)$$

where t represents the current iteration and *MaxIterat* represents maximum iterations. The new position of the whale on the spiral path can be calculated using Eq. (9);

$$\vec{A}(t+1) = Y' * e^{ws} * \cos(2\pi l) + \vec{A} * (t) \qquad (9)$$

where $Y' = \left| \vec{A}^*(t) - \vec{A}(t) \right|$ which indicates the distance of a whale and the best solution (prey). w is a constant that represents the shape of the logarithmic spiral. s is the random number generated between $[-1,1]$. Hence, the updated new position of the whale is calculated using 50% probability using a random number P_i generated between [0,1] as represented by Eq. (10).

$$\vec{A}(t+1) = \begin{cases} \vec{A}(t+1) = \overrightarrow{A^*}(t) - \vec{M} * Y & if(P_i < 0.5) \\ \vec{A}(t+1) = Y' * e^{ws} * \cos(2\pi l) + \vec{A} * (t) & if(P_i \geq 0.5) \end{cases}$$
$$(10)$$

1.3 Comparative Investigation of ABC and WOA in terms of exploration and exploitation

In the context of exploration, WOA uses the search of prey phase for exploration that completely depends on a random search agent which is a stochastic strategy. In the same way, ABC incorporates scout bees for exploration with the help of a random search. This makes both the algorithms to produce solutions concentrated in a local area at the initialization phase, losing its diversification. Hence, the search process prematurely converges returning sub-

optimal solutions in both of the algorithms. Hence, both ABC and WOA are weak at exploration.

In the context of exploitation, the local search process is incorporated using the encircling prey and the bubble net attacking method. The WOA exploitative phase guarantees convergence since positions of the whale are updated using the best solution (prey) obtained so far. Hence, proper exploitation is guaranteed by the encircling prey and the bubble net attacking method in the direction toward the prey since the search process is always guided by the best solution found so far. Comparatively, ABC exploitation is carried out using the employee bee phase and the onlooker bee phase where the positions of the food sources are updated by changing the single parameter of the old solution (food source) that causes the existence of similar food sources that converge at the same optimum locations. Also, the local search of ABC cyclically revisits the same solutions that create the problem of looping making the search converge prematurely. Hence, WOA is better at exploitation as compared to ABC.

1.4 Problems that are addressed by the proposed HAW

Many researchers have used ABC and WOA to develop optimal classifiers that can be used for medical diagnosis purposes, but still, the standard ABC suffers from the following issues which HAW addresses.

(i) The local search by the employee and onlooker bee cyclically revisits similar solutions inducing the problem of looping making the search process converge prematurely.

(ii) ABC optimization makes the solutions to be concentrated in local regions due to a lack of diversified solutions at initialization.

(iii) The food source positions are updated by changing the single parameter of the old solution (food source) which causes the existence of similar food sources that converge at the same optimum locations.

(iv) Exploitation is performed by two phases, namely the employee bee phase and the scout bee phase, whereas the exploration process is done only by scout bees, which leads to an imbalance in exploration and exploitation.

HAW that is capable of resolving the above issues can be used for generating an optimized ANN classifier that can accurately and efficiently be used for breast cancer diagnosis.

2 Related works

P. Shunmugapriya and S. Kanmani proposed an integrated algorithm of ABC and ACO for finding optimal feature subsets of medical datasets [31]. The global search followed by ABC is improved by using feature subsets generated by ACO to the ABC optimization process. The approach yielded an accuracy of 99.07% using Wisconsin breast cancer dataset (WBCD). The exploitation uses traditional greedy selection making algorithm to prematurely converge at the local optimal locations. The algorithm is only used for FS and no parameter optimization. Zorarpacı and Ozel introduced a hybrid algorithm of DE and ABC for optimal binary subsets selection [32]. The algorithm combines the high exploration property of DE with an improved onlooker bee phase of the ABC. The approach achieved F-measure of 92.2, 96.4 and 97.6 for decision tree classifier, naive Bayes classifier and RBF networks classifier, respectively, using WBCD. The algorithm is only used for FS and no parameter optimization.

Shanthi and Bhaskaran presented a modified ABC for FS [33]. The exploitation of the employee bees is improved where the neighborhood search process is improved using the global best solution. The modified ABC is used for FS using benchmark datasets called mammographic image analysis society (MIAS) and digital database for screening mammography (DDSM) for breast cancer diagnosis. Classification is carried out using self-adaptive resource allocation network. The accuracy was evaluated as 96.89% and 97.17% for MIAS and DDSM, respectively. The algorithm has not focused on the explorative phase and it has used only randomized initial solutions with loss of diversification. Rao et al. applied a FS algorithm using ABC and decision trees based on the gradient boosting model [34]. The features are selected from the Wisconsin breast cancer dataset and Haberman's survival dataset. A regression tree is used as the classifier where gradient descent finds the direction of the gradient of residuals. The classification accuracy is 74.3% for Haberman's cancer dataset and 92.8% for WBCD. It has not been evaluated in terms of complexity.

An efficient ABC is proposed by Badem et al., for optimal learning of deep neural networks (DNN) [35]. This algorithm used ABC and Broyden–Fletcher–Goldfarb–Shannon (BFGS) with limited memory. This proposed ABC tuned the parameters of DNN with cascaded autoencoder layers. The classification accuracy using WBCD is 73.03%. The step size of the neighborhood search is kept static throughout the entire search process affecting the convergence. Garro presented an optimized classification of DNA microarrays using ABC [36]. The optimal feature subsets from breast cancer datasets are

selected using ABC. Then the selected optimal feature subsets are given to MLP, radial basis function neural network and support vector machine (SVM). The accuracy attained is 94.7% for MLP, for SVM accuracy is 89.5% and for RBF it is 73.7%. The algorithm used standard ABC without any improvement. Palanisamy and Kanmani proposed ABC-based FS for UCI datasets [37]. The system chooses 2 features from 9 attributes from WBCD and yielded an accuracy of 96.69%. The system is simple but used only the standard ABC.

Optimal FS using ABC for UCI repository datasets is proposed [38]. The employee bee phase is modified using a modification rate where the feature is selected if the random number is greater than the modification rate. The classification accuracy is 75.87%. The algorithm has not focused on the explorative phase and it has used only randomized initial solutions with loss of diversification. Two hybrid algorithms are proposed based on ABC and PSO [39]. In the first algorithm, the employee bee phase is hybridized with PSO to find new velocity position updates. In the second algorithm, the onlooker and scout bee phase are improved using mutations of the genetic algorithm. Both of the algorithms have the highest accuracy of 99.14% with an optimal selection of 13 features using WBCD. The algorithm is only used for FS and no parameter optimization. A hybrid algorithm for FS using branch and bound approach and ABC is proposed [40]. The algorithm first applies the branch and bound and finds the first set of features. Then, it applies ABC to identify the second set of features. A union operation is done to form a new set of optimal features. The algorithm has not focused on classification.

Schiezaro and Pedrini used an optimal FS using whale bubble net hunting strategy for UCI repository datasets [41]. The algorithm handles exploitation using the bubble net attacking method phase. Further, a global search is carried out by the search for the prey phase. During the evaluation, the SVM classifier attained an accuracy of 98.77%, precision of 99.15%, recall of 98.64% and f-score of 98.9%. This algorithm is only used for FS and no parameter optimization. J. Jona and N. Nagaveni presented an optimal FS using the integration of ACO and cuckoo search [42]. Local search behavior of ACO is improved using the exploitation of cuckoo search. The algorithm selected feature set that is optimal from the set of 78 texture features derived using GLCM. The input is taken from the MIAS dataset. In this approach, 5 features were selected with 94% accuracy. The algorithm uses the SVM classifier for prediction. This algorithm showed increased performance of 4% and 2% when compared with PSO and ACO, respectively.

A novel hybrid whale–artificial bee colony optimizer framework is introduced by Siddavaatam and Sedaghat for

cross-layer optimization for Internet of Things (IoT) [93]. An efficient MAC for IoT has been designed to minimize energy consumption with extended network lifetime. The novel hybrid whale–artificial bee colony optimizer framework is used to obtain optimal nodes and the communication parameters in the IoT. It saves computation resources of the resource constrained IoT devices.

3 Materials and methodologies

A wrapper-based method that eliminates the use of statistical methods such as information gain or F-score is used for implementing the proposed HAW. The proposed architecture is depicted in Fig. 1. The input dataset is the breast cancer dataset where the total set is divided into three subsets. The first set that contains 50% of samples is used for training. The next 25% of samples are used for the validation, and the rest of 25% are used for testing. With the help of the optimal set of input features generated by HAW, the optimal features are selected from the three

subsets where the other features are rejected. The optimal selected features of the training set are used for training the underlying ANN classifier. The proposed wrapper architecture was implemented utilizing MATLAB 8.5 software. Backpropagation training is done using a neural network toolbox.

3.1 Breast cancer datasets used by the proposed Wrapper architecture

HAW is evaluated using breast cancer datasets such as the WBCD [87], Wisconsin diagnostic breast cancer dataset (WDBC) [88], Wisconsin prognostic breast cancer dataset (WPBC) [89], DDSM [90], MIAS [91] and INbreast database [92]. The description of the datasets used is given in Table 1.

The optimal initial weights and hidden node size generated by HAW are used as the initial parameter settings of ANN. The ANN error is calculated with the help of the validation set. If the validation error increases for six iterations continuously, the training of ANN is stopped.
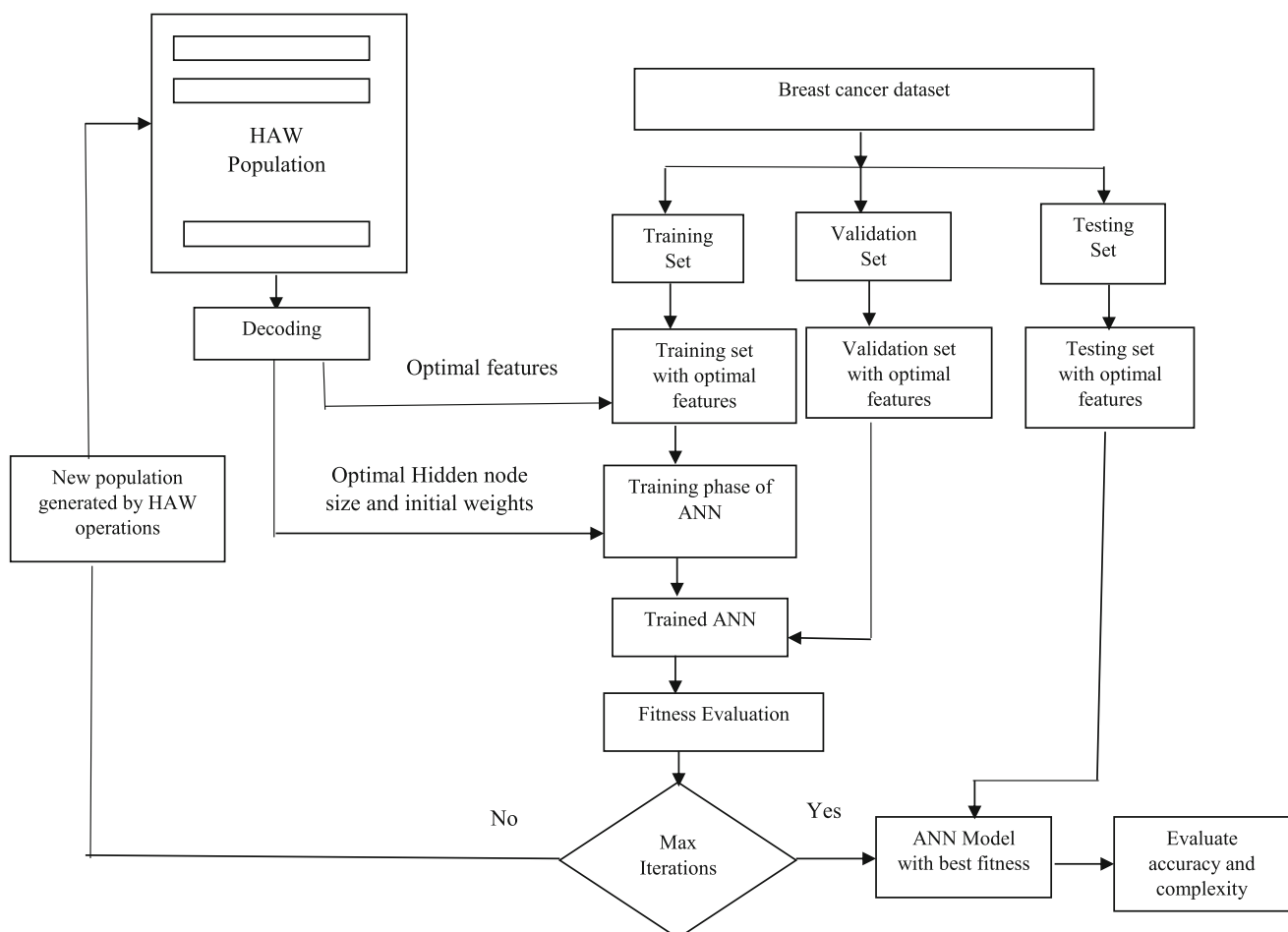

Fig. 1 Proposed wrapper architecture

Table 1 Breast cancer datasets used by the proposed wrapper architecture

Dataset	Cases	Features
WBCD	699—total	9
	241—malignant	
	458—benign	
WDBC	569—total	30
	357—malignant	
	212—benign	
WPBC	198—total	32
	151—nonrecurrent	
	47—recurrent	
DDSM	480—total	21
	170—malignant	
	310—benign	
MIAS	338—total	21
	138—malignant	
	200—benign	
INbreast database	275—total	21-
	95—malignant	
	180—benign	

The fitness of trained ANN is calculated using Eqs. (12 and 13). The ANN with high fitness (best) is selected and tested using the testing set with optimal feature subsets. The complexity (number of connections) of final ANN achieved is calculated using Eq. (11).

$$\text{Connect} = U * V + V * W + V + W \qquad (11)$$

'U' gives input features (size), 'V' indicates hidden node size, and 'W' indicates output nodes (size). The resulting ANN with the least connections guarantees less complexity. Fitness of ANN is calculated by Eq. (6). A higher value of $ANNErr$ indicates low-fitness ANN.

$$\text{ANN Err} = \left(\frac{P_{max} - P_{min}}{L * w}\right) \sum_{i=1}^{l} \sum_{j=1}^{w} \left(A_j^i - B_j^i\right)^2 \qquad (12)$$

$$\text{Fitness} = \frac{1}{ANNError} \qquad (13)$$

'l' and 'w' is the size of the output nodes and validation examples, respectively. P_{max} and P_{min} are maximum and minimum actual output, respectively. B_j^i and A_j^i is the target output and actual output, respectively.

The initial solution representation is given using Fig. 2. I bits give the random initial weights, in which 2^I different initial weights can be explored. J bits give the hidden node size so that 2^J hidden node size can be explored. K bits give the feature bits that represent the total features. If a feature is selected, then 'K' bit is one; otherwise, it is zero. The size of the K bits may vary based on the total number of features available in the dataset.

3.2 Detailed description of the proposed HAW algorithm

HAW optimization algorithm is framed by the integration of a mutative initialization phase of ABC optimization with the exploitation phase of the whale optimization technique. The standard ABC is weak in exploration because of localized initial food sources due to the poor random search process. Hence, the HAW algorithm has used a mutative exploration phase at its initialization such that the algorithm can explore the entire problem space and finds out new promising regions. This employee bee phase of the ABC optimization process is integrated with the exploitative of WOA such that the employee bees follow the encircling prey/bubble attacking method of whales to update the positions of the food sources. The best food source found at each iteration is considered as the target prey of WOA. The HAW involves two stages: In the first stage, HAW uses a mutative initialization phase is proposed using different mutations and it derives a possible set of diversified solutions. In the second stage, an employee bee attacking phase is proposed such that the optimum set of solutions derived by the mutative initialization phase forms the initial food source positions of the employee attacking phase that follows the attacking method of whales for the prey. The simulated annealing technique is used in the employee bee attacking phase to make the algorithm to escape from the local optimum locations and avoid looping problems. A flowchart representation of the HAW algorithm is shown in Fig. 3. HAW optimization is summarized as follows:

(i) A mutative initialization phase is proposed to derive a set of diversified solutions to expedite the search speed at the exploration phase.

(ii) An employee attacking phase is proposed so that the employee bees adapt the encircling prey/ bubble net attacking method of whales for updating the current food source positions during their

Fig. 2 Initial solution representation

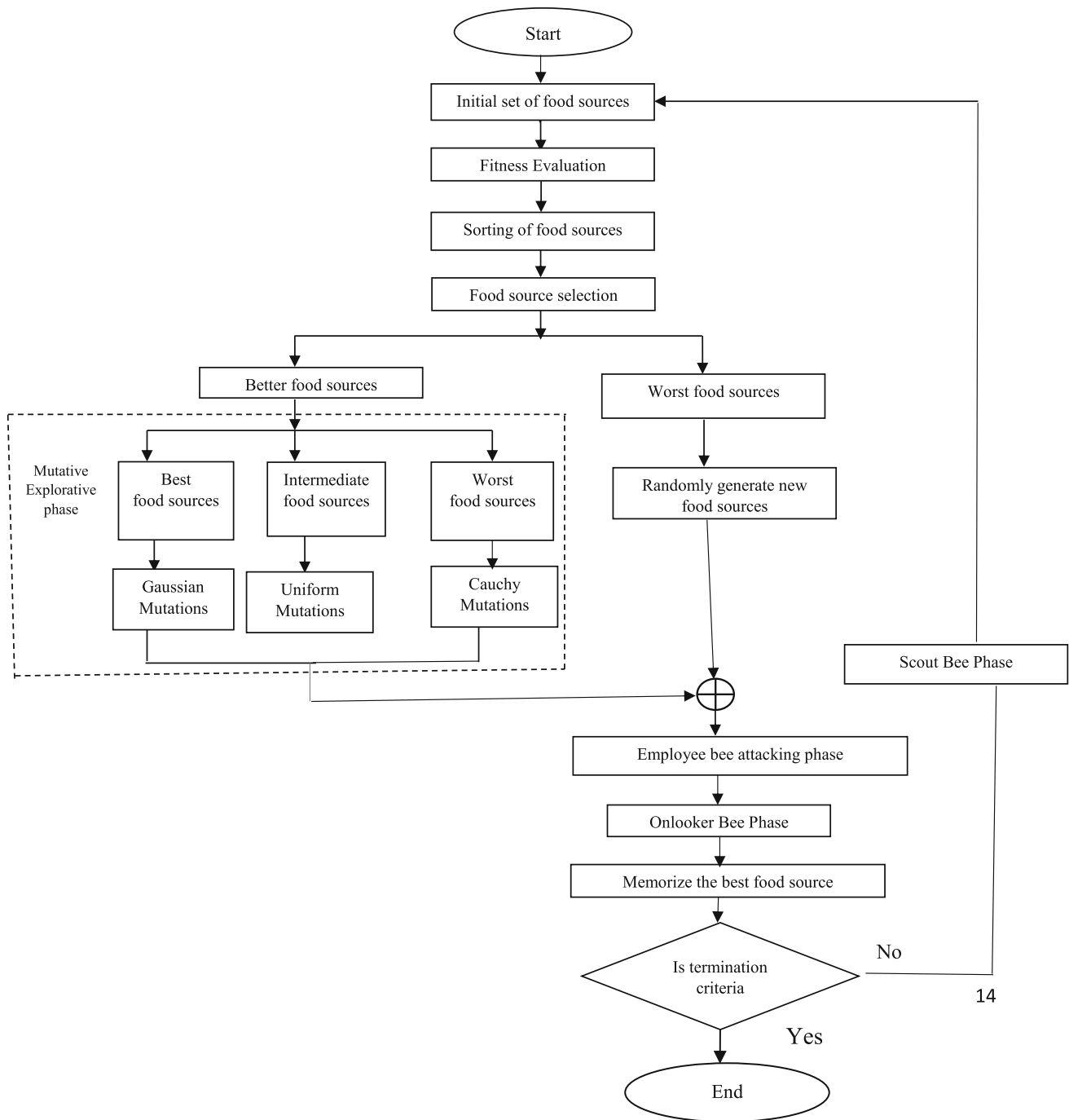| I bits	J bits	K bits
000001010111111.....	101000011110101.......	1000110........
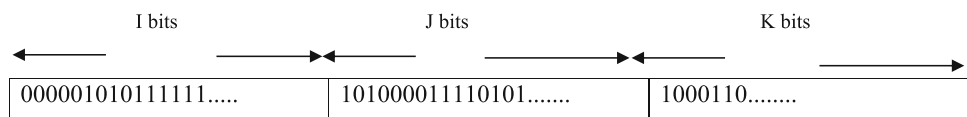
Fig. 3 Flowchart representation of the proposed HAW algorithm

foraging process. The exploitation of the employee bee attacking phase is guided by the best food source (prey of the whales) found so far.

(iii) To escape from suboptimal location and to avoid looping problems, simulated annealing (SA)-based employee attacking phase is proposed.

(iv) The onlooker bee phase and scout bee phase are followed in the same way as that of the standard ABC optimization.

3.2.1 Initialization and fitness calculation

A food source indicates a possible solution of the underlying optimization problem. Each food source is generated using the 'dim' number of variables that represent the dimension of problem space considered. The generation of the initial population is done through the random distribution of food sources using Eq. (14).

$$A_k^l = A_{min}^l + random(0,1) * \left(A_{max}^l - A_{min}^l\right) \qquad (14)$$

A_k^l represents the kth variable of food source k and $k = 1, 2 \ldots N$ where N represents the maximum size of the food sources, where $N = 1, 2 \ldots dim$, and 'dim' represents the dimension based on the number of parameters of the underlying optimization problem. $random(0,1)$ is the random number generated between 0 to 1. A_{max}^l represents the maximum bound of the lth variable of the optimization problem and A_{min}^l gives the minimum bound of the lth variable of an optimization problem. The estimated tight bound of Algorithm (3) is $\theta(n^2)$ where n will be the number of food sources. The algorithm for the initial generation of food sources is given in Algorithm (3).

Algorithm 3: Initialization of HAW

> **Generation of Initial population**
>> For $m = 1$ to i do
>>> For $n = 1$ to d do
>>>> $Y_m^n = Y_{min}^n + random(0,1) * (Y_{max}^n - Y_{min}^n)$
>>> End for n
>> End for m
>
> **Affinity Evaluation and sorting**
> For $m = 1$ to i do
>> Calculate fitness (Y_m)
>> Sort (Y_m) based on fitness
> End for

3.2.2 Proposed mutative exploration phase

The mutative exploration phase detects multiple food sources based on its quality. Better food sources are selected from the total population. Further, they are divided into three subpopulations based on the fitness difference between each food source and the best food source in the population with the help of three different threshold values such as limit1, limit2 and limit3. The three different subpopulations of food sources are subjected to different mutations where higher-fitness food sources are mutated less and low-fitness food sources are mutated high. Thus, the amount of mutation is inversely proportional to the fitness value of food sources. Better food sources with high fitness values are grouped as A_{k1} food sources whose fitness

is close to the fitness of the best food source of the total population. The A_{k1} food sources are best; hence, a local search process is facilitated around the best food sources, which are mutated using Gaussian mutations where Eq. (15) is utilized to make small random changes to A_{k1} food sources.

$$A_{k1}' = A_{k1} + \mu_g.G(0,1) \qquad (15)$$

A_{k1}' is the mutated food source, which is generated after the Gaussian mutation; μ_g is the mutation rate indicating the strength of Gaussian noise added; A_{k1} is the original food source; and $G(0,1)$ is the random number of Gaussian distribution whose mean is zero and variance is one.

Intermediate food sources with intermediate fitness values are grouped as A_{k1} food sources whose fitness shares an intermediate fitness difference from the best food source of the total population. The A_{k2} food sources are intermediate; hence, a uniform search process is facilitated around the intermediate food sources and is mutated using uniform mutations where Eq. (16) is utilized to generated uniformly mutated food sources of the A_{k2} represented as A_{k2}' in which a random value is chosen from a solution and replaced with a uniform random value between the user-defined upper *(Ub)* and lower *(Lb)* limits.

$$A_{k2}' = A_{k2} \qquad (16)$$

The worst food sources with low fitness values are grouped as A_{k3} food sources whose fitness values are at a larger difference from the fitness of the best food source in the population. The A_{k3} food sources are worse than A_{k1} and A_{k2}. Hence, a global search is facilitated around the worst food sources, in which A_{k3} food sources are mutated using Levy mutations (LM) using Eq. (17) that are more probable to escape from the local optimum.

$$A_{h3}' = A_{h3} + \mu_c.C(0,1) \qquad (17)$$

A_{h3}' is the mutated food source generated after Cauchy mutation and μ_c is the Cauchy mutation rate that indicates the strength of the mutation. The mutated food sources along with the replaced worst food sources A_x forms the new set of food source A_{new} eligible for exploitation. The estimated tight bound of Algorithm 4 is $\theta(n^2)$ where n are the number of food sources mutated in explorative phase. The mutative exploration phase is described by algorithm (4).

Algorithm 4: Mutative exploration phase

Replace the worst $'x'$ food source by randomly generated food source A_x

For each better food source (A_k)

Select A_{k1}, A_{k2}, and A_{k3} food sources from better food sources.

A_{k1} if $\{|\text{Fitness}(Ak) - \text{Fitness}(A_{best})|\} \leq \text{limit1}$

A_{k2} if $\{|\text{Fitness}(A_k) - \text{Fitness}(A_{best})|\} \leq \text{limit2}$ and if $\{|\text{Fitness}(A_k) - \text{Fitness}(A_{best})|\} \geq \text{limit1}$

A_{k3} if $\{|\text{Fitness}(A_k) - \text{Fitness}(A_{best})|\} \geq \text{limit2}$

If A_{k1} food sources

Mutate using equation (15)

Else If A_{k2} food sources

Mutate using equation (16)

Else

Maturate using equation (17)

Set of mutated food sources generated are $\overline{A_k}$

Set $A_{new} = \overline{A_k}$

End for

Generate $A_{new} = A_{new} \cup A_x$

3.2.3 Proposed employee bee attacking phase

The food sources generated from the mutative exploration phase along with the replaced randomly generated food sources form the initial food source positions of the exploitation phase. Each food source is assigned with an employee bee where a bee searches a better food source around the current food source following the same way a whale search for the prey. An employee bee follows a circular path during searching for a neighborhood food source where the best food source found so far is considered as the center of the circle. This exploitative mechanism makes the search process to be always guided by the best optimal locations. The step size β of the search process is kept high at the initial stages of the iteration on the circular path to facilitate the process of exploration, whereas in the later iterations the step size is gradually decreased to facilitate the process of exploitation. This dynamic step size β of the search process makes the employee bees to search the entire problem space such that the bees can reach remote locations that were not reached in the exploration phase. The problem of oscillations and local optima can be eliminated with the dynamic step size in the circular path. A random number ran is generated using uniform distribution and checked against the control variables C_1 and C_2. Then, if a randomly generated number ran is less than C_1 then the food source position is found using Eq. (18).

$$N_k^l = A_k^l + \beta * \left(A_k^l - A_d^l\right) \tag{18}$$

where $\beta = \text{random}(-1,1)$ is the step size that is dynamically varied across the iterations. N_k^l is the new food source. A_k^l is the current food source, A_d^l is the randomly selected food source.

Then if the randomly produced number is less than C_2 and a probability check is done using P_i where if $P_i < 0.5$ then new neighborhood position is found searching around a circular path keeping the best food source found so far A_k^* as the center of the circle using Eq. (19).

$$N_k^l = \vec{A}_k^* - \vec{M} * Y \tag{19}$$

M represents the coefficient vectors obtained using Eq. (6 &8). Y is the distance between the current food source and the center of the circle (best food source) represented in Eq. (20).

$$Y = \left|A_k^* - A_k^l\right| \tag{20}$$

Another probability check, if $P_i \geq 0.5$ then new neighborhood position is found searching around a circular path keeping the best food source A_k^* found so far as the center of the circle using Eq. (21)

$$N_k^l = Y * e^{aw} * \cos(2\pi l) + A_k^* \tag{21}$$

a is a constant that represents the shape of the logarithmic spiral. w is the random number generated between $[-1,1]$.

If fitness of N_k^l (neighborhood food source) is lesser than fitness A_k^l (current food source), the employee bee attacking phase accepts N_k^l by accepting downhill movements to make the search process to escape from its local optimum in the fitness landscape. The acceptance of the worst food sources is based on the probability value $e^{\frac{-\Delta E}{t}}$ where ΔE represents the difference between the current temperature and the next randomly generated temperature. This is done using a simulated annealing-based selection mechanism where better and worst solutions are accepted in the employee bee attacking phase. Acceptance or rejection of worst solutions is based on the controlling parameter called the simulated annealing temperature where the probability of accepting the worst solutions decreases as the iteration proceeds where the temperature gets decreased. At the initial stages, the simulated annealing temperature 't' is set to have a high value where the value $\frac{-\Delta E}{t}$ tends toward 0 making the probability value $e^{\frac{-\Delta E}{t}}$ toward 1 allowing the acceptance of the worst solutions. As the iteration grows, the value $\frac{-\Delta E}{T}$ tends toward 1 making the probability value

$e^{\frac{-\Delta E}{t}}$ toward 0 allowing the acceptance of better solutions. This makes the HAW algorithm to allow both the uphill and downhill movements in the fitness landscape making the algorithm to eliminate the problem of striking at local optimum locations. When the iterations grow, the simulated annealing temperature 'T' is cooled down using Eq. (22).

$$T(t+1) = \emptyset * T(t) \tag{22}$$

where $T(t+1)$ is the new temperature and $T(t)$ is the temperature of the previous iteration. '$\emptyset$' is the simulated annealing constant which is set close to 1. The estimated tight bound of Algorithm 5 is $\theta(n^2)$ where n will be the number of the employee bees assigned to the food sources. The employee bee attacking phase is described by algorithm (5).

Algorithm 5: Proposed employee bee attacking phase

For $k= 1$ to n do

    For $l=1$ to dim do

      If $ran \geq C_1$

        Search for the neighbouring food source using Equation (18)

      Else If $ran < C_2$

        If ($P_i < 0.5$)

          Search for the neighbouring food source using Equation (19)

        Else if ($P_i \geq 0.5$)

        Search for the neighbouring food source using Equation (21)

        End if

      End if

    End for l

    If $fitness(N_k^l) > fitness(A_k^l)$

      Accept N_k^l

    Else

      Calculate $\Delta E = fitness(A_k^l) - fitness(N_k^l)$

      Accept N_k^l with probability $e^{\frac{-\Delta E}{t}}$

    End if

End for k

3.2.4 Onlooker bee phase

The information about the newly generated food sources N_k^l is shared with onlooker bees. Further, onlooker bees produce probability value Z_k for a food source, and it receives from the employee bee attacking phase using Eq. (3). The estimated tight bound of Algorithm 6 is $\theta(n)$ where n is the number of food sources shared by employee bees.

Algorithm 6: Onlooker bee phase

For $k= 1$ to N do

    Calculate probability using Equation (3)

      If $random(0,1) < Z_k$

        Accept N_k^l

      Else

        Reject N_k^l

   End for

   Memorize the best food source A_{best}

3.2.5 Scout bee phase

The abandoned food sources are replaced and new food sources are introduced by the scout bees. In each iteration, if the food source isn't improved then the limit value associated with the food source is incremented and if the food source doesn't improve for certain iterations and crosses the threshold limit value, then those food sources are replaced by the scout bees by random generation process using Eq. (1). The estimated tight bound of Algorithm 7 is $\theta(n)$ where n is the number of unimproved solutions. The detailed description of the scout bee phase is given in Algorithm (7).

Algorithm 7: Scout bee phase

While (A_k does not improve)

    count $=$ count $+ 1$

    If count $>$ count$_{max}$

      Replace A_k with the newly generated food source

    Else

  count $= 0$

Iteration$=$ Iteration $+1$

4 Performance evaluation of the proposed HAW optimization

The HAW algorithm is used to generate the optimal feature subsets, initial weights and hidden node size of an ANN which is trained using three different learning mechanisms based on backpropagation such as resilient backpropagation (RP), Levenberg–Marquardet backpropagation (LM) and momentum-based gradient decent backpropagation (GD). The HAW performance is analyzed to find out the backpropagation variant that achieves the least ANN error with low complexity using WBCD. Further, the best backpropagation variant that is selected is used for training the ANN when tested for the rest of the datasets. The HAW-optimized ANN is analyzed for ten independent runs using various generation sizes, and the mean accuracy, mean connections and mean computational time are calculated.

4.1 Parameter settings and experimental setup using WBCD

The implementation of backpropagation is done using default training parameters. The winner takes all approach in output nodes is used for classification. The parameter settings of the ANN classifier are shown in Table 2. HAW utilized the cancer1 dataset [88] which contains preprocessed WBCD samples. The values between 0 and 1 are rescaled and attribute values that are missing are filled using mean values of non-missing attributes. The training dataset is formed using the first 349 samples which are 50% of the total dataset, the second 175 samples that lead to 25% of the total dataset are used as the validation dataset and the last 175 samples that are the remaining 25% are used as the testing dataset.

Table 2 Parameter settings of ANN using WBCD dataset

Parameter	Value
Training	Backpropagation
Input node size	9
Output node size	2
Initial weights (number of bits)	15
Hidden node size (number of bits)	2
Input features (number of bits)	Vary based on the feature size
Activation function (hidden node)	Hyperbolic tangent
Activation function (output node)	Pure linear
Training set (no. of samples)	349 (50%)
Validation set (no. of samples)	175 (25%)
Testing set (no, of samples)	175 (25%)

The HAW parameter settings are shown in Table 3. The values of thresholds limit1 and limit2 are set using trial-and-error method. The count limit is set as 10 to enhance global exploration. The employee bees size and the onlooker bees size are set according to the size of the solution space. The mutation rates of the Gaussian, uniform and Cauchy rates were set using the trial-and-error method to enhance the local, uniform and global search process. The updating probability and the attacking probability using trial-and-error method for different generation sizes and they are set accordingly to avoid convergence at local optimal locations.

4.2 Evaluation of Proposed HAW using WBCD with respect to accuracy, complexity and computational time

The HAW is evaluated for the complexity in terms of the number of connections and accuracy using ten runs for generation sizes 10, 20 and 30 as given in Table 4. HAW-RP achieved the highest mean accuracy of 99.25% at the 20th generation size. The mean connections were 10.40 for HAW-RP, which is low when compared to HAW-LM and HAW-GD. HAW gained the less complexity ANN trained with RP. The validation error convergence of the optimized ANN for RP, LM and GD for various generations is depicted in Fig. 4.

Followed by HAW-RP, HAW-GD is 98.57% accurate, with average connections of 14.21 in the 30th generation. Next to HAW-GD, HAW-LM achieved high accuracy of 98.22% in the 30th generation with average connections of 15.46. The classification accuracy of the optimized ANN for RP, LM and GD for various generations is depicted in

Table 3 Parameter settings of HAW using WBCD dataset

Parameter	Value
Employee bees (size)	30
Onlooker bees (size)	30
Scout bees (size)	1
Total colony (size)	60
Count limit ($Count_{max}$)	10
Better food source selection (A_k)	80%
Worst food source selection (A_x)	20%
Control threshold (limit1)	0.0001
Control threshold (limit2)	0.5
Mutation rate (μ_g)	0.4
Mutation rate (μ_c)	0.6
Updating probability by the bees in the spiral	0.5
Attacking probability	0.5

Table 4 Performance of the proposed HAW algorithm for various backpropagations

Max generation size	Proposed HAW-RP						Proposed HAW-LM						Proposed HAW-GD					
	Classification accuracy (%)			Number of connections			Classification accuracy (%)			Number of connections			Classification accuracy (%)			Number of connections		
	Best	Mean	Q1 Q3	Best	Mean	Q1 Q3	Best	Mean	Q1 Q3	Best	Mean	Q1 Q3	Best	Mean	Q1 Q3	Best	Mean	Q1 Q3
10	98.42	97.55	97.05 98.22	11	11.56	10.75 10.95	97.75	96.73	97.05 98.22	11	16.42	97.05 98.22	97.83	96.91	96.65 97.21	11	14.65	97.05 98.22
20	99.55	99.25	97.05 98.22	10	10.40	9.75 9.89	97.95	96.71	97.05 98.22	12	15.34	97.05 98.22	98.12	97.38	97.55 98.22	11	14.82	97.05 98.22
30	99.50	98.48	97.05 98.22	11	11.23	10.75 10.95	98.45	98.22	97.05 98.22	13	14.46	97.05 98.22	98.91	98.57	97.34 98.08	12	14.21	97.05 98.22

Fig. 5. The accuracy of HAW-RP was 1.05% more than HAW-LM and 0.69% more than HAW-GD.

The confusion matrix based on the true positive, true negative, false positive and false negative of HAW-RP, HAW-LM and HAW-GD for the best network achieved is shown in Table 5. Figure 6 gives the complexity of the ANN achieved in terms of the number of connections for various generation sizes. HAW-RP achieved lower complexity with lesser connections, followed by HAW-GD and HAW-LM. The mean hidden node count of HAW-RP was 22.5% less than HAW-LM and 20.78% less than HAW-GD. HAW-RP produced ANN complexity of having 28.08% fewer connections than HAW-LM and 26.81% fewer connections than HAW-GD.

The performance metrics of the optimized ANN generated using HAW-RP, HAW-LM and HAW-GD are shown in Table 6. HAW-RP showed higher sensitivity and specificity, followed by HAW-GD and HAW-LM. Figure 7 shows the hidden node count for HAW-RP, HAW-LM and HAW-GD over generation sizes.

Table 7 gives the importance of FS on the performance of the HAW-optimized ANN in terms of classification. Table 7 shows that HAW-optimized ANN with FS improved prediction accuracy with less complexity when compared with the performance without the FS.

Table 8 shows the confusion matrix for best network achieved using HAW-RP, with selected features such as uniformity of cell size, single epithelial cell size and bare nuclei.

The evaluation of HAW-optimized ANN with respect to computational time is given in Table 9. The mean computational time for 10 runs was estimated for various sizes of 10, 20 and 30. The computational time of ANN optimized using HAW-LM was low in comparison with the computational time of HAW-RP and HAW-GD.

Figure 8 compares HAW with PSO, DE, ABC, BA, ACO, BFO, DA, GA, AIS, MBO, WOA and SSA using the WBCD. HAW attained highest accuracy, which was 9.67%, 7.53%, 10.89%, 12.27%, 12.78%, 8%, 10.03%, 0.97%, 8.83%, 9.91%, 5.03% and 3.39%, respectively, higher than when compared to above-mentioned algorithms. This time complexity of the proposed ANN architecture is calculated to be $O(n * t * (ij + jk))$ where n is the number of epochs, t is the number of training examples and i, j and k are the number of nodes in the first, second and third layers of ANN architecture.

Figure 9 compares HAW and existing hybrid algorithms that are ABC-based using WBCD, which includes ABC-ACO [31], ABC-DE [32], ABC-DA [97] and ABC-gradient decision tree [34]. The accuracy of HAW was 0.07%, 1.58%, 2.36% and 2.02%, respectively, higher when compared to the above-mentioned algorithms.
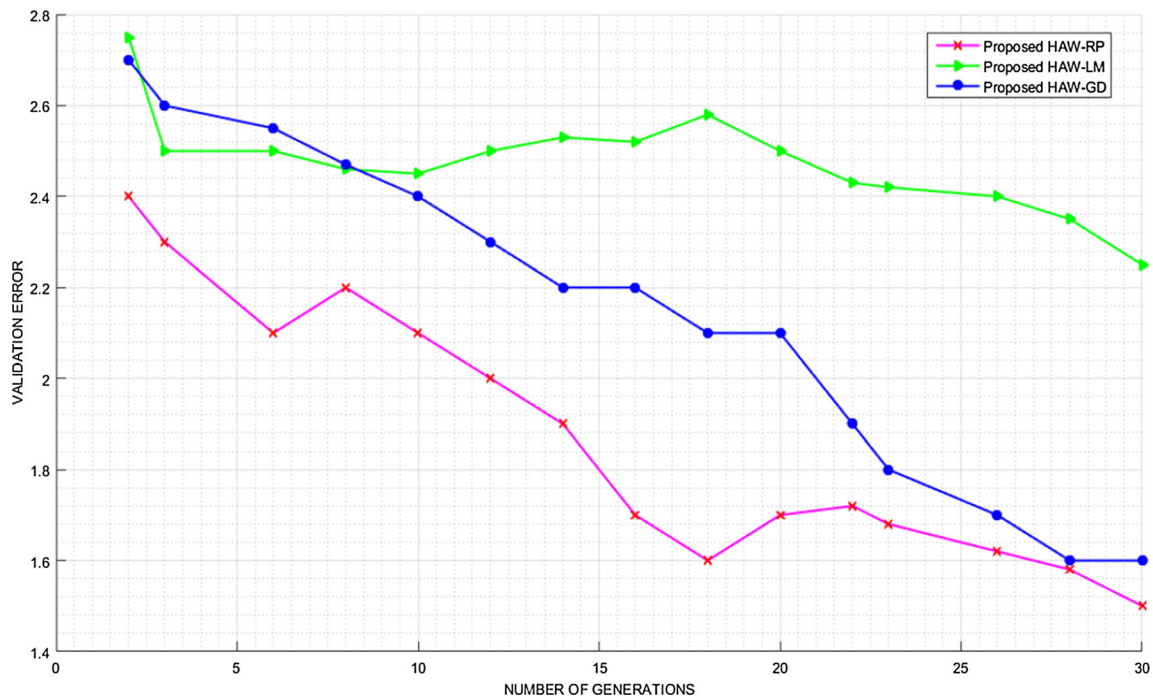
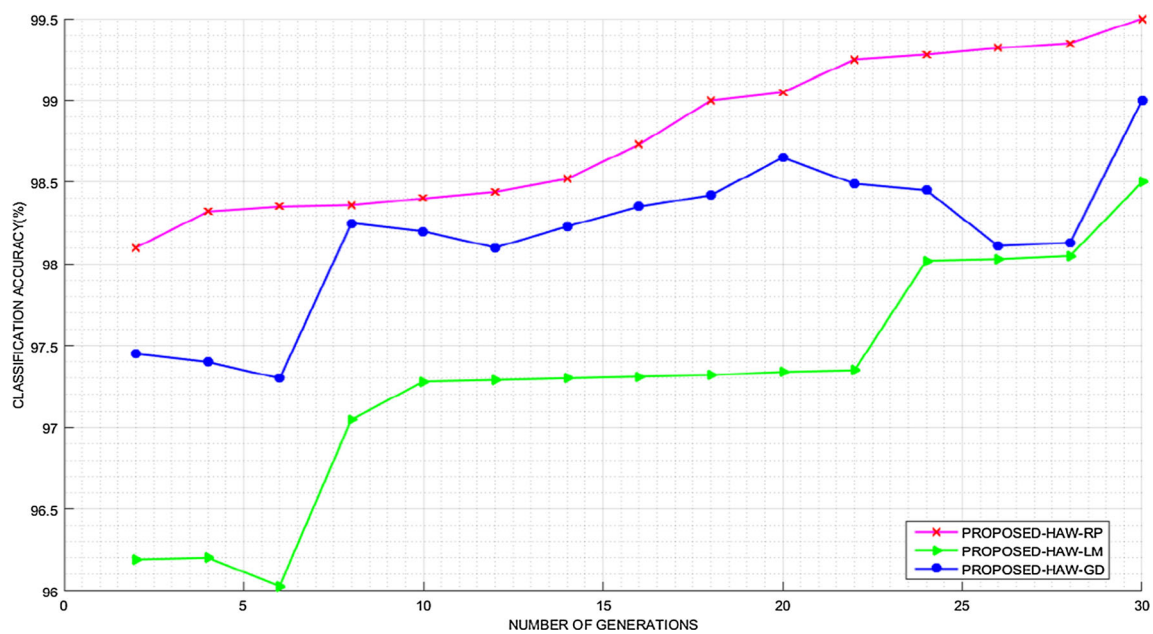**Fig. 4** Convergence of validation error for HAW

Fig. 5 Performance of HAW with respect to classification accuracy

Table 10 gives a comparison of the HAW-optimized classifier with existing breast cancer diagnosis schemes using WBCD datasets. Table 10 shows that optimized ANN classifier using HAW outperformed existing breast cancer diagnosis schemes.

4.3 Performance evaluation of HAW using different breast cancer datasets

The HAW-RP yielded the best ANNs with high accuracy and low complexity. Hence, HAW-RP was tested for the rest of the breast cancer datasets taken for investigation. Table 11 shows the performance of the. Table 11 shows the performance of the HAW-RP-optimized ANN for different

Table 5 Confusion Matrix of the HAW-optimized ANN across ten runs

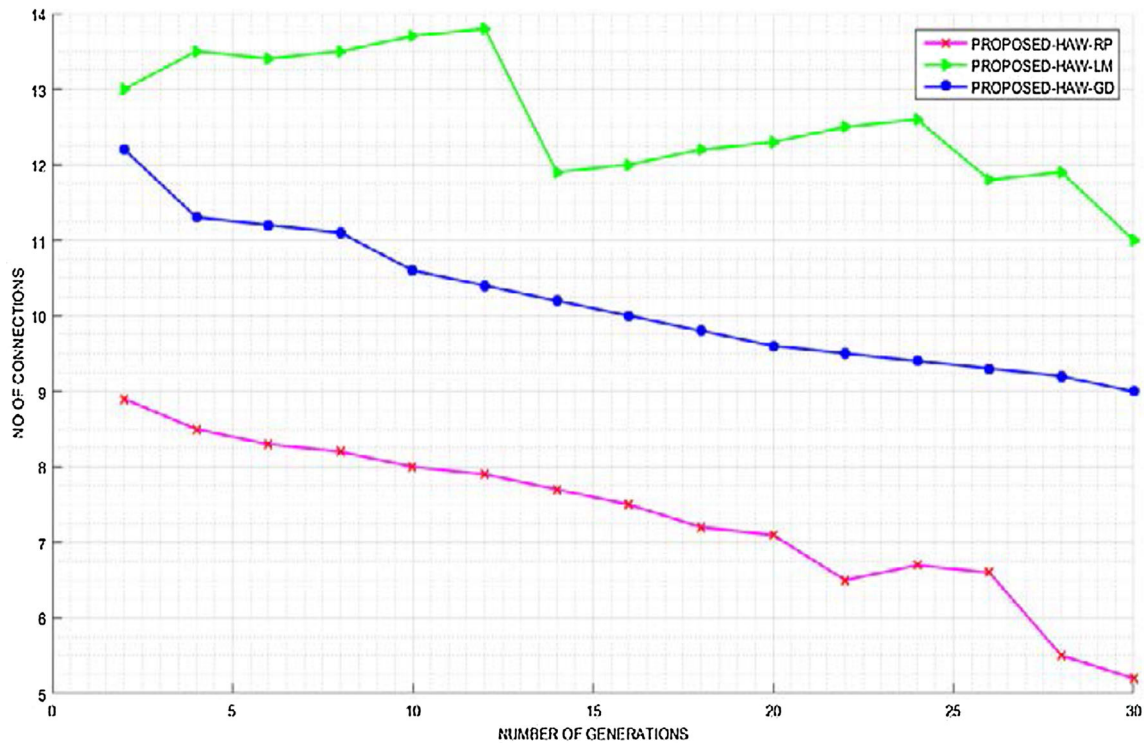| Methods of comparison	Number of actual cases	Test outcome-predicted	
		Malignant	Benign
HAW-RP	650 (malignant)	647 (TP)	10 (FN)
	1100 (benign)	3 (FP)	1090 (TN)
HAW-LM	650 (malignant)	640 (TP)	21 (FN)
	1100 (benign)	10 (FP)	1079 (TN)
HAW-GD	650 (malignant)	642 (TP)	17 (FN)
	1100 (benign)	8 (FP)	1083 (TN)

Fig. 6 Performance of HAW with respect to the number of connections

breast cancer datasets. The average classification accuracy, F-measure, number of connections and computational time have been calculated using ten different independent runs

Table 6 Performance of HAW-optimized ANN based on different metrics

Metrics	Proposed HAW-RP	Proposed HAW-LM	Proposed HAW-GD
Sensitivity (%)	98.47	96.82	97.42
Specificity (%)	99.72	99.08	99.26
Accuracy (%)	99.25	98.22	98.57
Precision (%)	99.53	98.46	98.76
Negative predictive value (NPV) (%)	99.09	98.09	98.45
F-measure	0.99	0.97	0.98

for different generation sizes, and the best of the average values has been recorded. The HAW algorithm-optimized ANN model; namely, HAW-RP has shown better performance with a smaller number of connections in comparison with standard ABC and WOA for the datasets taken for investigation.

Figures 9 and 10 show the graphical representation of the performance of the standard ABC and WOA-optimized ANN model and the HAW-optimized ANN model for various breast cancer datasets taken for investigation with respect to classification accuracy and an average number of connections. The HAW-optimized ANN model has shown higher accuracy and low complexity in comparison with the standard ABC and WOA-optimized ANN model. Table 12 shows the confusion matrix with a frequently selected feature set for the various datasets used for
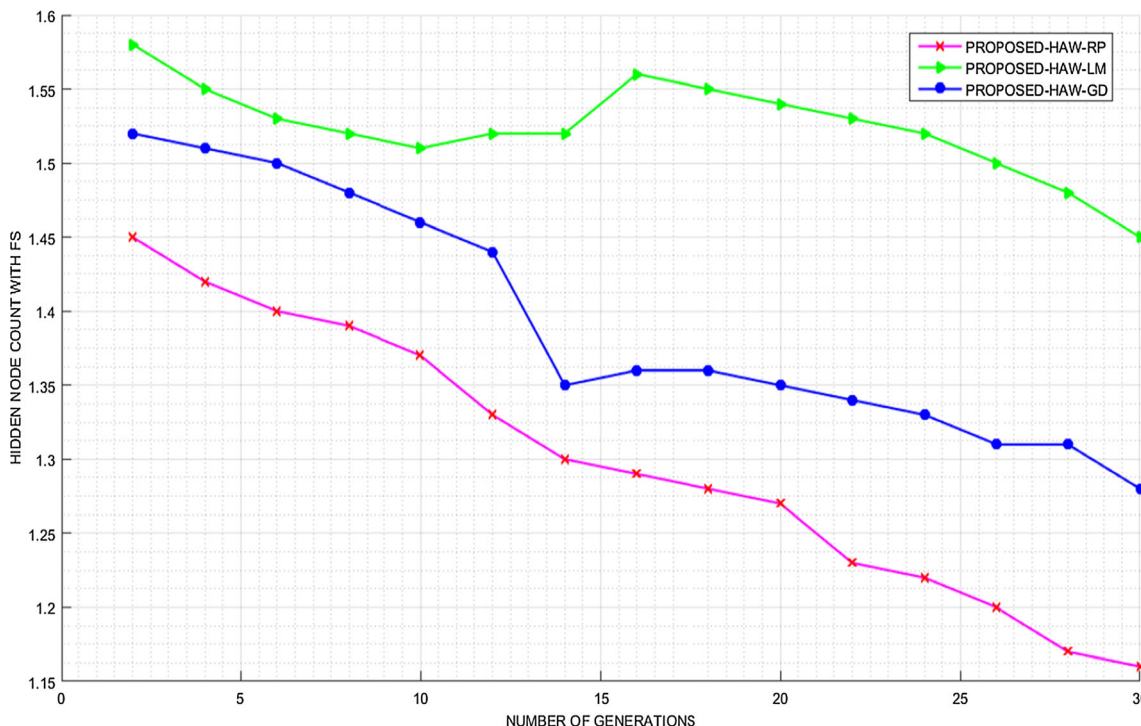
13682

Neural Computing and Applications (2021) 33:13667–13691

Fig. 7 Evolution of hidden node count with FS for HAW across different generations

Table 7 Performance of HAW-RP based on FS

Max generation size	Feature selection (FS)	Average hidden node count	Quartiles Q1 Q3	Average number of selected features	Quartiles Q1 Q3	Average number of connections	Quartiles Q1 Q3	Average accuracy (%)	Quartiles Q1 Q3
10	With FS	1.4	0.9	5.8	4.5	12.56	10.75	97.55	97.05
			1.1		5.2		10.95		98.22
	Without FS	2.5	1.9	9	8.2	27.5	23.87	88.34	84.13
			2.2		8.9		26.54		86.23
20	With FS	1.2	0.8	5	4.4	10.40	9.75	99.25	97.05
			0.9		5.2		9.89		98.22
	Without FS	2.2	1.3	9	7.9	26.6	25.02	90.42	88.97
			2.0		8.7		26.17		89.59
30	With FS	1.3	0.7	5.1	4.2	12.23	10.75	98.48	97.05
			1.0		4.9		10.95		98.22
	Without FS	2.2	1.5	9	8.2	24.8	22.31	89.05	87.55
			1.7		8.7		24.09		88.07

investigation of HAW-RP-based ANN after the execution of ten independent runs.

Table 13 shows the performance comparison of the HAW-optimized ANN model in terms of average classification accuracy for ten independent runs with various classifiers for different datasets. The HAW-ANN model has shown higher accuracy compared with existing classifiers for all the datasets taken for investigation.

4.4 Comparison with existing approaches using WDBC and WPBC

Figure 10 compares HAW with PSO, DE, ABC, BA, ACO, BFO, DA, GA, AIS, MBO, WOA and SSA using the WDBC. Each algorithm is executed for 10 runs various generation sizes 10, 20 and 30 and the best is taken for comparison. HAW-based ANN attained highest accuracy, which was 7.65, 6.83, 12.96, 17.12, 8.84, 5.69, 9.2, 7.42,

Table 8 Confusion Matrix of HAW-RP-optimized best ANN with features selected

FS	Actual cases		Predicted cases		Selected feature set
			Benign	Malignant	
With FS	Benign	110	110	1	Uniformity of cell size, single epithelial cell size, bare nuclei
	Malignant	65	0	64	
Without FS	Benign	110	102	7	Clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses
	Malignant	65	8	58	

Table 9 Performance-based on average computational time

Maximum generation size	Average CPU time(s)		
	Proposed HAW-RP	Proposed HAW-LM	Proposed HAW-GD
10	427.5	301.5	502.8
20	820.2	789.6	845.9
s30	1105.9	1036.5	1254.1

12.96, 11.3, 7.65 and 5.35%, respectively, higher than when compared to above-mentioned algorithms.

Table 14 gives a comparison of the HAW-optimized classifier with existing breast cancer diagnosis schemes using WDBC and WPBC datasets. Table (6.10) shows that the proposed optimized ANN classifier using HAW outperformed existing breast cancer diagnosis schemes.

Figure 10 compares HAW with PSO, DE, ABC, BA, ACO, BFO, DA, GA, AIS, MBO, WOA and SSA using the WPBC. Each algorithm is executed for 10 runs various generation sizes 10, 20 and 30 and the best is taken for
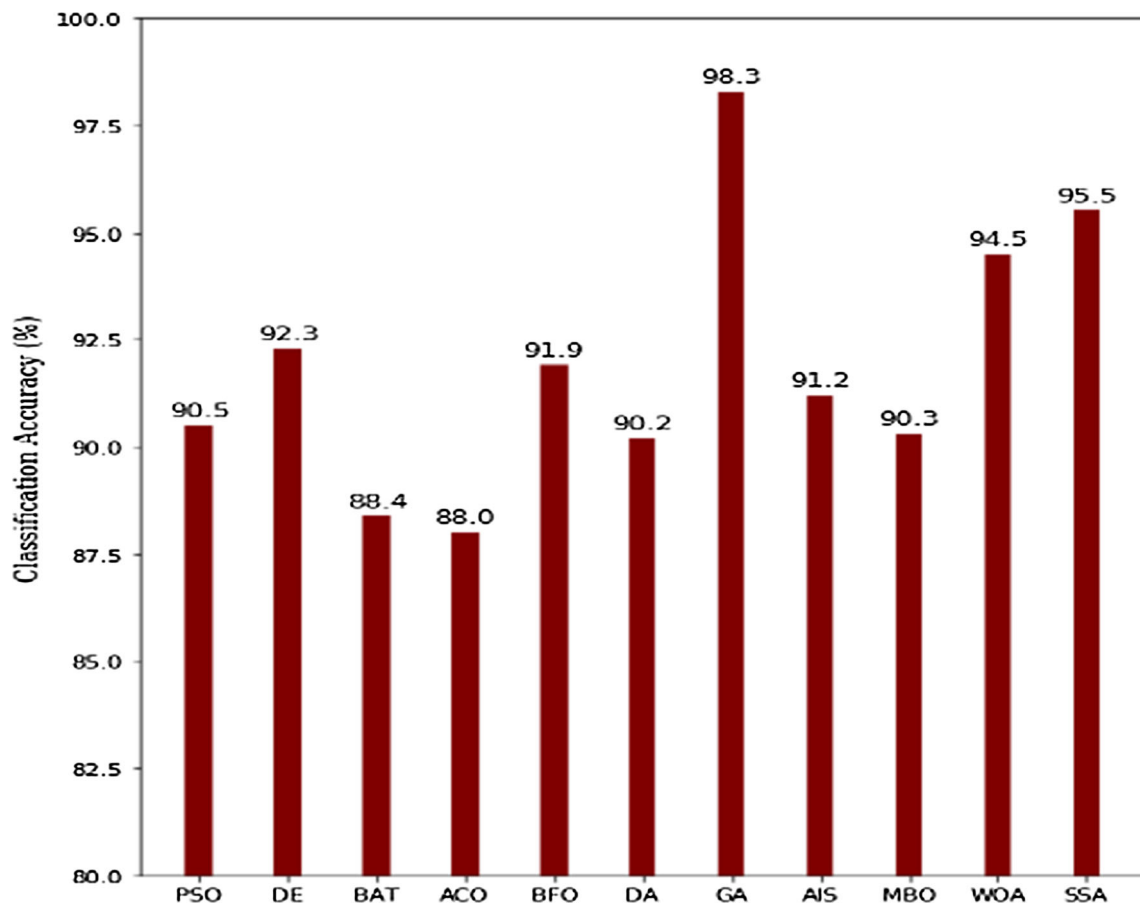

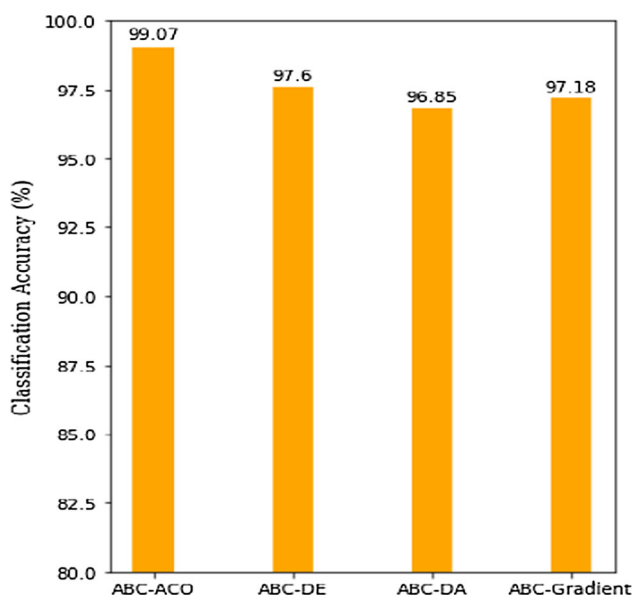**Fig. 8** Comparison between evolutionary methods using WBCD dataset

Fig. 9 Comparison of ABC-based hybrid algorithms using WBCD datasets

comparison. HAW-based ANN attained highest accuracy, which was 7.6%, 6.06%, 11.33%, 13.03%, 6.41%, 9.31%, 10.44%, 5.02%, 10.44%, 7.6%, 2.99% and 2.67%,

respectively, higher when compared to the above-mentioned.

4.5 Comparison with existing approaches using DDSM, MIAS and INbreast databases

Figure 11 shows the comparison of HAW with evolutionary algorithms such as PSO, DE, ABC, BA, ACO, BFO, DA, GA, AIS, MBO, WOA and SSA. Each algorithm is executed for 10 runs various generation sizes 10, 20 and 30 and the best is taken for comparison. Using DDSM database, HAW-based ANN achieved accuracy which is 7.98%, 7.16%, 13.3%, 17.48.1%, 9.17%, 6.01%, 9.53%, 7.74%, 13.3%, 11.64%, 6.81% and 6.35% more than above-mentioned algorithms, respectively. Using MIAS database, HAW-based ANN achieved highest accuracy which is 10.53%, 10.28%, 10.4%, 8.2%, 8.22%, 12.67%, 10.28%,7.17%, 9.42%, 10.28%, 8.22% and 7.87% more than above-mentioned algorithms, respectively. Using INbreast database, HAW-based ANN achieved highest accuracy which is 7.6%, 9.14%, 14.83%, 9.26%, 5.99%, 11.47%, 8.66%, 5.99%, 8.07%, 12.36%, 10.97% and 10.11% more than above-mentioned algorithms, respectively.

Table 10 Comparison with existing breast cancer diagnosis schemes

WBCD dataset		
First author, year	Method	Accuracy (%)
Quinlan,1996 [44]	C4.5 classifier	94.74
Hamilton, 1996 [45]	RAIC classifier	95.00
Nauck,1999 [46]	NEFCLASS	95.06
Pena-Reyes,1999 [47]	FUZZY-GA classifier	97.36
Setino, 2000 [48]	Neuro-rule classifier	98.10
Albrecht, 2002 [49]	LSA classifier Machine	98.80
Fogel, 1995 [50]	ENN classifier	98.05
Abonyi, 2003 [51]	SFC classifier	95.57
Polat, 2007 [52]	LS-SVM classifier	98.54
Gujaro-Berdinas, 2007 [53]	LIS classifier	96.00
Karabatak, 2009 [7]	AR + ANN classifier	97.40
Monirul, 2010 [62]	CAFS + ANN classifier	98.76
Stoean, 2013 [54]	SVM + EA classifier	97.07
Fadzil Ahmad, 2014 [55]	GANN-ANN classifier	98.29
Shunmugapriya, 2017 [31]	ABC-ACO-J48 classifier	99.04
Rao, 2018 [34]	ABC + GD classifier	92.8
Ghanem, 2018 [97]	ABC-DA + ANN classifier	98.29
Karthik, 2018 [56]	DNN classifier	98.62
Dalwinder, 2019 [94]	Ant lion + ANN	97.85
Derangula, 2021 [95]	PSO + firefly + naive Bayes	98.00
Nayak, 2020 [96]	EHO + NN	98.37
Proposed work	HAW-ANN classifier	99.25

Table 11 Performance of HAW-RP-optimized ANN for different datasets

| Datasets	Average classification accuracy (%)			Average F-measures			Average number of connections			Average computational time (secs)		
	ABC	WOA	HAW-RP	ABC	WOA	HAW-RP	ABC	WOA	HAW-RP	ABC	WOA	HAW-RP
WBCD	89.5	94.5	99.2	0.8812	0.9327	0.990	25.21	28.11	10.40	640.6	700.5	820.2
WDBC	87.5	91.5	98.5	0.8648	0.9005	0.975	24.67	22.17	19.82	590.3	600.2	750.2
WPBC	86.5	93.5	96.3	0.8532	0.9206	0.953	24.89	23.19	18.52	300.1	350.7	425.5
DDSM	87.2	92.5	98.8	0.8641	0.9187	0.981	36.23	37.87	16.8	535.6	600.2	700.6
MIAS	89.4	91.2	98.7	0.8894	0.9072	0.978	33.78	35.78	25.34	450.2	500.9	656.8
INbreast	86.3	89.3	99.1	0.8570	0.8831	0.986	30.82	31.12	25.12	400.1	455.7	600.4
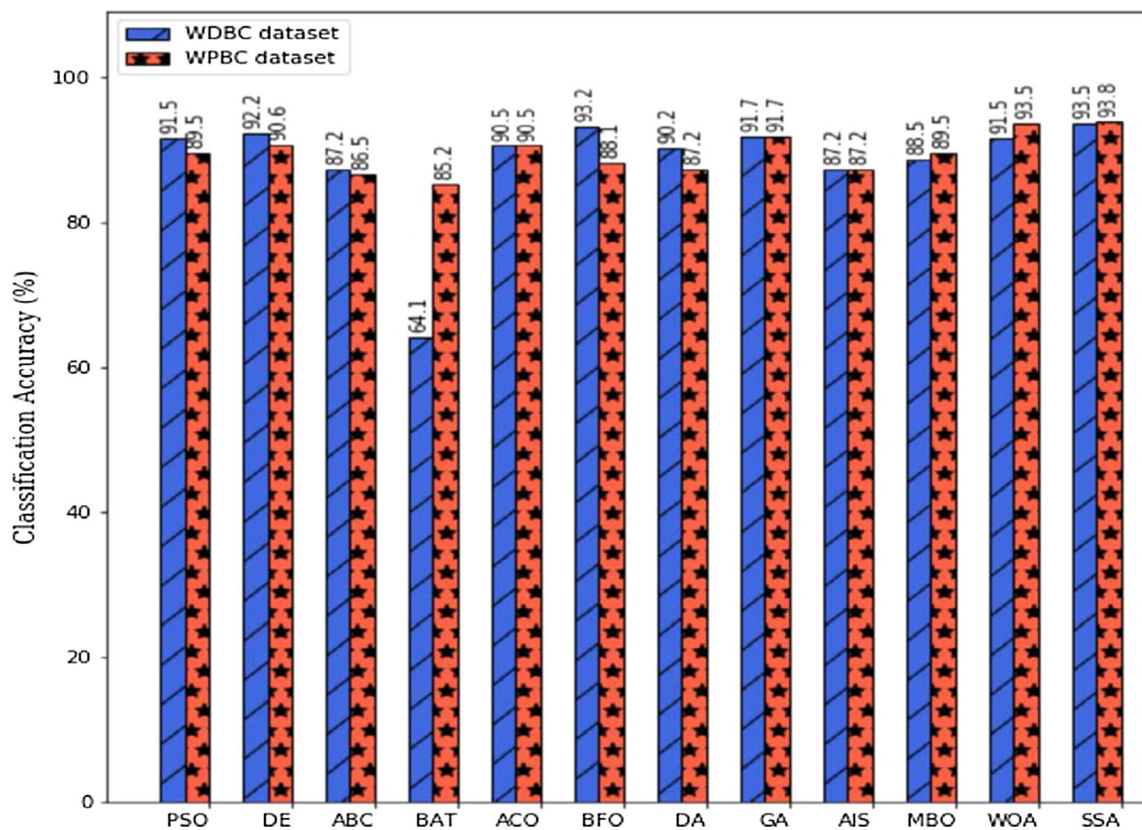
Fig. 10 Comparison with other evolutionary methods using WDBC and WPBC dataset

Table 15 gives a comparison of the HAW-optimized classifier with existing breast cancer diagnosis schemes using DDSM, MIAS and INbreast datasets. Table 16 shows that optimized ANN classifier using HAW outperformed existing breast cancer diagnosis schemes.

4.6 Comparison with existing approaches that doesn't involve complex tuning process for breast cancer diagnosis

Table 16 gives a comparison of the HAW-ANN scheme with existing breast cancer diagnosis schemes that involves either FS or parameter tuning process. Table 17 shows that having both FS and parameter optimization of ANN in parallel is vital and the HAW-ANN outperformed existing breast cancer diagnosis schemes that has used either FS or the parameter tuning of the classifier involved.

5 Discussions

The advantage of proposed HAW lies in deriving an optimal ANN by the optimal searching of hidden node size, the initial value of weights and input features using a simple wrapper approach. ANN uncertainty arises due to the improper selection of the best set of parameters of the

Table 12 Confusion matrix after ten runs for various datasets

HAW-RP

WDBC dataset	Test outcome-predicted		Frequently selected features
	Malignant (actual—420)	Benign (actual—1000)	Texture, radius, compactness concavity, fractal dimension
	415 (TP)	16 (FN)	
	5 (FP)	984 (TN)	
WPBC dataset	Test outcome-Predicted		Radius, texture, area, concavity, symmetry, fractal dimension
	Recurrent (actual—190)	Nonrecurrent (actual—300)	
	183 (TP)	11 (FN)	
	7 (FP)	289 (TN)	
DDSM dataset	Test outcome-predicted		Skewness, kurtosis, correlation, contrast, area, major axis length, skeleton
	Malignant (actual—400)	Benign (actual—900)	
	392(TP)	7 (FN)	
	8 (FP)	893 (TN)	
MIAS dataset	Test outcome-predicted		Solidity, perimeter, extent, skewness, entropy, mean, variance, eccentricity
	Malignant (actual—250)	Benign (actual—600)	
	247 (TP)	8 (FN)	
	3 (FP)	592 (TN)	
INbreast dataset	Test outcome-predicted		Skewness, kurtosis, correlation, area, major axis length, convex area, eccentricity
	Malignant (actual—290)	Benign (actual—400)	
	287 (TP)	3 (FN)	
	3 (FP)	397 (TN)	

Table 13 Average classification accuracy for various classifiers

Datasets	Average classification accuracy (%)				
	SVM	Naive Bayes	Random forest	MLP	HAW-ANN
WBCD	92.3	83.4	85.9	88.6	99.2
WDBC	89.7	88.4	82.4	87.5	98.5
WPBC	84.5	89.1	84.9	88.1	96.3
DDSM	92.2	80.9	82.5	92.2	98.8
MIAS	88.4	80.3	77.9	86.2	98.7
INbreast	88.3	82.6	88.2	91.1	99.1

model. These uncertainties also occur due to inherent variability present in the input of ANN. If the inputs and the other parameters of the model are uncertain, it causes uncertain and inaccurate output of the model. This problem of uncertainty in the ANN architecture used in this paper has been eliminated by choosing the best parameters of the model using proposed HAW algorithm. The algorithm has also selected the optimal input features from different datasets that acts as the input of the ANN used. Uncertainty in the input data has also been overcome by appropriate data division and data processing methods such as normalization and data filling procedures. However, detailed uncertainty analysis can be taken as the future work for this research.

The main objective of this research work is to propose a hybrid optimization algorithm to select the optimal feature set to be processed by the neural network in order to have a reduced computational time and complexity without sacrificing its performance. The optimal feature sets obtained were only 25% of the original set of features, and hence, this research work has the feed forward neural network considering the amount of processing data and to avoid the problem of over fitting of the data. The proposed ANN architecture has not been tested for high-dimensional datasets. This could be done as the future work where deep learning models or other decision-making models can be used when the set of features processed by the model is high. For the datasets taken for evaluation, the proposed

Table 14 Comparison with existing breast cancer schemes for WDBC and WPBC dataset

WDBC dataset		
First author, year	Method	Accuracy (%)
Bamakan, 2014 [57]	CFS-SVM classifier	87.84
	SVM-Filtered classifier	87.84
	Logistic regression-CFS classifier	95.95
	Logistic regression-filtered classifier	96.62
Xue, 2012 [58]	BPSO classifier	92.98
Xue, 2014 [59]	PSO classifier	93.98
Maldonado, 2011 [60]	KP-SVM classifier	97 ± 0.9
	REF-SVM classifier	95.25 ± 1.0
Miao, 2011 [61]	Training self-classifier	85.12
	Training random classifier	83.54
Luukka, 2006 [62]	LDA classifier	97.19
	C4.5 classifier	94.06
	DIMLP classifier	96.92
	SIM classifier	98.2
Razieh, 2015 [63]	PSO-KDE classifier	98
Djellali, 2018 [39]	ABC-GA classifier	91.67
Dalwinder, 2019 [94]	Ant lion + ANN	98.37%
Proposed Work	HAW-ANN classifier	98.5

WDBC dataset		
First author, year	Method	Accuracy (%)
Belciug, 2012 [64]	MLP-GA hybrid classifier	81.11
	MLP-BP classifier	60.21
	PNN classifier	74.43
	PCNN classifier	72.10
Salama, 2012 [65]	SMO classifier	76.28
	J48 classifier	76.28
Sridevi, 2014 [66]	Correlation + Rough Set classifier	85
Proposed work	HAW-ANN classifier	96.3

ANN architecture considered in this paper has achieved accurate results.

6 Conclusions

This paper proposed a hybrid algorithm by integrating the standard ABC with WOA for simultaneous feature subset selection and parameter optimization of ANN for breast cancer diagnosis. Performance evaluation of HAW using backpropagation variants such as RP, LM and GD is done for different breast cancer datasets. Based on the results achieved, RP achieved the best prediction accuracy with a low-complexity ANN. HAW-RP achieved the highest mean accuracy of 99.25%. The mean connections were 10.40 for HAW-RP, which is low when compared to HAW-LM and HAW-GD for WBCD dataset. For WDBC, WPBC, DDSM, MIAS and INbreast databases, HAW-

optimized architecture achieved 98.5, 96.3, 98.8, 98.7 and 99.1, respectively. From the results of this study, it is concluded that having both FS and parameter optimization of ANN is vital. In comparison with existing researches, the average accuracy attained by HAW is promising. In comparison with existing evolutionary algorithms, the HAW algorithm was more accurate and less complex.

Further, the limitations of the HAW-ANN classifier are based on the 'no free lunch theorem' which states that classifier trained on a particular feature set may not be used for other feature sets. Based on this, the HAW-optimized classifier that is trained on the breast cancer datasets has not been yet tested for other medical datasets and other high-dimensional datasets. The proposed wrapper approach of HAW-ANN scheme is complex and requires more computational time than the existing breast cancer scheme since it involves FS and parameter tuning process parallel.
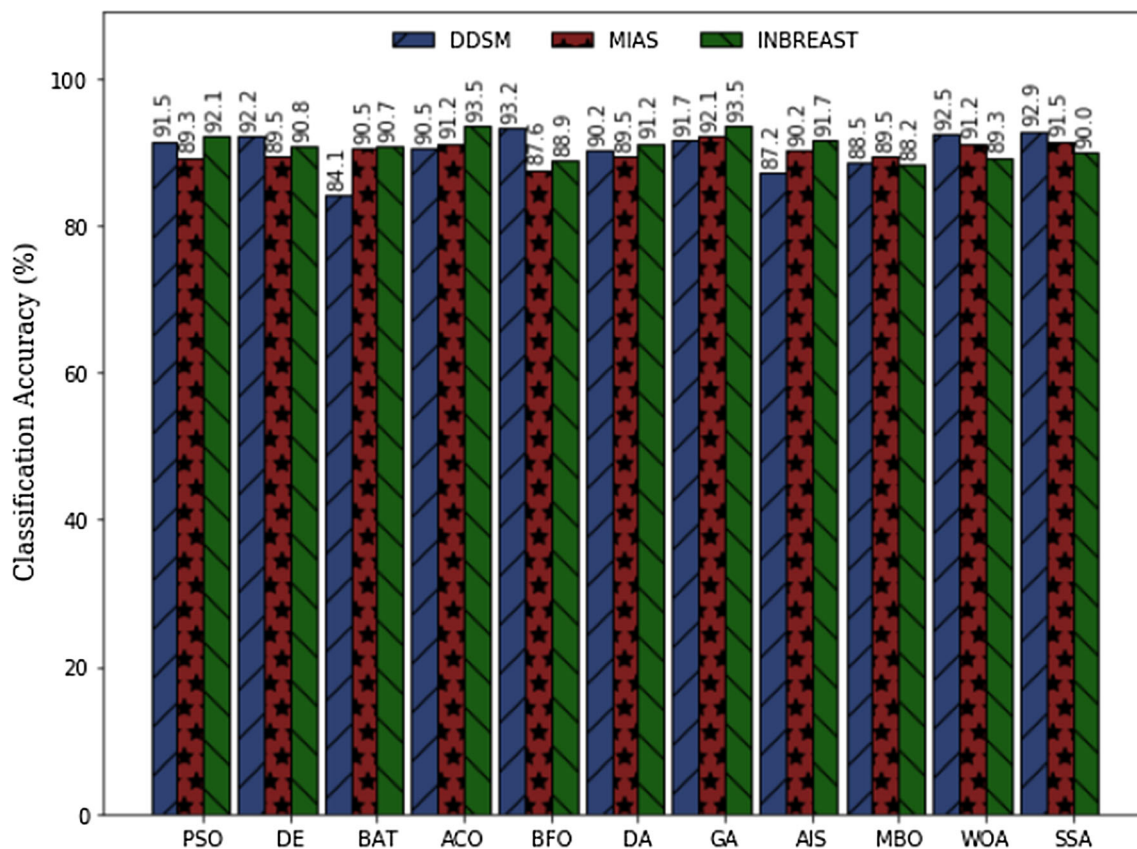
Fig. 11 Comparison with other evolutionary methods using DDSM, MIAS and INbreast

Table 15 Comparison with existing breast cancer diagnosis schemes for DDSM, MIAS and INbreast datasets	First author, year	Method	Database	Accuracy (%)
	Wang, 2014 [67]	SVM classifier	DDSM	92.74
	Liu, 2014 [68]	SVM classifier	DDSM	93.00
	Saki, 2013 [69]	OWBPE classifier	MIAS	89.28
	Buciu, 2011 [70]	PSVM classifier	MIAS	96.43
	Tahmasbi, 2011 [71]	ANN-MLP classifier	MIAS	96.43
	Tahmasbi, 2010 [72]	ANN-MLP classier	MIAS	92.80
	Zhang, 2012 [73]	SVM classifier	DDSM	72.00
	Verma, 2010 [74]	SCNN classifier	DDSM	94.28
	Verma, 2009 [75]	SCBDL classifier	DDSM	97.50
	Rojas, 2009 [76]	Bayesian classifier	DDSM, MIAS	81
	Dheeba, 2012 [77]	PSO-ANN classifier	MIAS	97.61
	Dheeba, 2012 [78]	DE-WNN classifier	MIAS	97.84
	Rahimeh, 2015 [79]	GA-CNN classifier	MIAS, DDSM	88.15, 94.99
	Andrik, 2017 [80]	CED classifier	INbreast	98.9 ± 0.4
	Ribl, 2018 [81]	DNN classifier	INbreast	95
	Proposed work	HAW-ANN classifier	DDSM	98.8
			MIAS	98.7
			INbreast	99.1

In the future, the proposed hybrid algorithm HAW can be used for segmentation of the breast regions and other abnormal regions such as masses and microcalcifications from the digital mammograms eliminating the background portions. This can be done by using HAW to generate optimal thresholds that can be used by various

Table 16 Comparison with existing breast cancer diagnosis schemes

First author, year	Method	Database	Accuracy (%)
Ghanem, 2018 [97]	ABC-DA-based FS + ANN	WBCD	98.29
Rao, 2018[34]	FS + decision tree	WBCD	97.18
Xue, 2014 [59]	PSO-based FS + KNN	WBCD	94.74
Kermani, 1995 [82]	GA-based FS + ANN	WBCD	94.7
Abbass, 2002 [84]	PSO-based parameter tuning of WNN	WBCD	98.1
Supriya, 2019 [86]	DA-based FS + gray wolf-based parameter tuning of ANN	WBCD	99
Dhanya, 2020 [85]	FS + KNN	WBCD	93.57
Liu, 2013 [68]	FS + SVM	DDSM	93.00
Verma, 2009 [83]	GA-based FS + ANN	DDSM	85
Buciu, 2011 [70]	Feature reduction + proximal SVM	MIAS	96.43
Dheeba, 2012 [77]	PSO-based parameter tuning of ANN	MIAS	97.61
Dheeba, 2012 [78]	PSO-based parameter tuning of WNN	MIAS	97.84
Rahimeh, 2015 [79]	GA-based FS + CNN	MIAS, DDSM	88.15
			94.99
Proposed work	HAW-based simultaneous FS and parameter tuning + ANN	WBCD	99.2
		WDBC	98.5
		WPBC	96.3
		DDSM	98.8
		MIAS	98.7
		INbreast	99.1

segmentation methods such as region growing and multi-level thresholding. HAW can also be used for segmentation of various abnormalities that occur in the different parts of the body such as the brain, liver and kidney. In future, HAW can also be used to select the optimal input feature subsets for various existing classifiers under different medical datasets for various disease diagnoses and it can be used to find out optimal parameters for different ANN architectures such as the CNN, RNN and deep belief neural networks. In future, the HAW can also be used in optimal parameter tuning for different classifiers such as the SVM, RF, decision tree and optimal selection of fuzzy rules.

References

1. American Cancer Society (2018) Global Cancer: Facts & Figures, 4th edition http://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/global-cancer-facts-and-figures/global-cancer-facts-and-figures-4th-edition.pdf.
2. Eltoukhy MM, Faye I, Samir BB (2010) A comparison of wavelet and curvelet for breast cancer diagnosis in digital mammogram. Comput Biol Med 40(4):384–391
3. Zhao M, Fu C, Ji L, Tang K, Zhou M (2011) FS and parameter optimization for support vector machines: A new approach based on genetic algorithm with feature chromosomes. Expert Syst Appl 38(5):5197–5204
4. Fu JC, Lee SK, Wong STC, Yeh JY, Wang AH, Wu HK (2005) Image segmentation, feature selection and pattern classification for mammographic microcalcifications. Comput Med Imaging Graph 29:419–429
5. Ghoncheh M, Pournamdar Z, Salehiniya H (2016) Incidence and mortality and epidemiology of breast cancer in the world. Asian Pac J Cancer Prev 17(S3):43–46
6. Übeyli ED (2007) Implementing automated diagnostic systems for breast cancer detection. Expert Syst Appl 33(4):1054–1062
7. Karabatak M, Ince MC (2009) An expert system for detection of breast cancer based on association rules and neural network. Expert Syst Appl 36(2):3465–3469
8. Punitha S, Amuthan A, Suresh Joseph K (2019) enhanced monarchy butterfly optimization Technique for effective breast cancer diagnosis. J Med Syst 43(7):1–14
9. Thompson K, Suresh Joseph K (2018) Particle swarm optimization-based energy efficient channel assignment technique for clustered cognitive radio sensor networks. Comput J Oxford Univ Press 61(6):926–936
10. Thompson S, Suresh Joseph K (2016) PSO assisted OLSR routing for cognitive radio vehicular sensor networks. In: ACM international conference on informatics and analytics (pp. 1–8)
11. Colorni A, Dorigo M, Maniezzo V (1991) Distributed optimization by ant colonies. In: Proceedings of the first European conference on artificial life (pp. 134–42)
12. Eberhart RC, Kennedy J (1995) A new optimizer using particle swarm theory. In: Proceedings of the sixth international symposium on micro machine and human science (pp. 39–43)
13. Yang XS (2010) A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization (NICSO 2010) (pp. 65–74). Springer; New York

14. Karaboga D, Basturk B (2007) A powerful and efficient algorithm for numericalfunction optimization: artificial bee colony (ABC) algorithm. J Global Optim 39:459–471

15. Mirjalili S, Mirjalili S, Hatamlou A (2015) Multi-verse optimizer: a nature-inspiredalgorithm for global optimization. Neural Comput Appl 1–19 /03/172015

16. Gandomi AH, Alavi AH (2012) Krill herd: A new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simul 17(12):4831–4845

17. Wang G-G, Deb S, Cui Z (2015) Monarch butterfly optimization. Neural Comput Appl 28(3):1–20

18. P. K. M (2002) Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control Syst 22(3), 52–67

19. Castro L, Timmis JI (2003) Artificial immune systems as a novel soft computing paradigm. J Soft Comput 7: 526

20. Gandomi AH (2014) Interior search algorithm (ISA): A novel approach for global optimization. ISA Trans 53(4):1168–1183

21. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM (2017) Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191

22. Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH (2020) Marine Predators Algorithm: A Nature-inspired Metaheuristic. Expert Syst Appl 113377

23. Furundzic D, Djordjevic M, Bekic AJ (1998) Neural networks approach to early breast cancer detection. J Syst Architect 44(8):617–633

24. Kolen JF, Pollack JB (1991) Back propagation is sensitive to initial conditions. Adv Neural Inf Process Syst 3:860–867

25. Ferentinos KP (2005) Biological engineering applications of feedforward neural networks designed and parameterized by genetic algorithms. Neural Netw 18(7):934–950

26. Setiono R, Liu H (1997) Neural-network feature selector. IEEE Trans Neural Networks 8(3):654–662

27. Verikas A, Bacauskiene M (2002) FS with neural networks. Pattern Recogn Lett 23(11):1323–1335

28. Kabir MM, Islam MM, Murase K (2010) A new wrapper FS approach using neural network. Neurocomputing 73(16–18):3273–3283

29. Telikani A, Gandomi AH, Shahbahrami A, Dehkordi MN (2020) Privacy-preserving in association rule mining using an improved discrete binary artificial bee colony. Expert Syst Appl 144:113097

30. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67

31. Shunmugapriya P, Kanmani S (2017) A hybrid algorithm using ant and bee colony optimization for FS and classification (AC-ABC Hybrid). Swarm Evol Comput 36:27–36

32. Zorarpacı E, Özel SA (2016) A hybrid approach of differential evolution and artificial bee colony for FS. Expert Syst Appl 62:91–103

33. Shanthi S, Bhaskaran VM (2014) "Modified artificial bee colony-based FS: A new method in the application of mammogram image classification." Int J Sci Eng Tech Res (IJSETR) 3(6):1664–1667

34. Rao H, Shi X, Rodrigue AK, Feng J, Xia Y, Elhoseny M, Yuan X, Gu L (2019) FS based on artificial bee colony and gradient boosting decision tree. Appl Soft Comput 74:634–642

35. Badem H, Basturk A, Caliskan A, Yuksel ME (2017) A new efficient training strategy for deep neural networks by hybridization of artificial bee colony and limited–memory BFGS optimization algorithms. Neurocomputing 266:506–526

36. Garro BA, Rodríguez K, Vázquez RA (2016) Classification of DNA microarrays using ANNs and ABC algorithm. Appl Soft Comput 38:548–560

37. Palanisamy S, Kanmani S (2012) Artificial Bee Colony Approach for Optimizing FS. Int J Comput Sci Issue 9(3):432–438

38. Schiezaro M, Pedrini H (2013) Data FS based on Artificial Bee Colony algorithm. EURASIP J Image Video Process 1:2013

39. Djellali H, Djebbar A, Zine NG, Azizi N (2018) Hybrid Artificial Bees Colony and Particle Swarm on FS. In: Computational intelligence and its applications IFIP advances in information and communication technology (pp. 93–105)

40. Nagarajan G, Minu R, Muthukumar B, Vedanarayanan V, Sundarsingh S (2016) Hybrid genetic algorithm for medical image feature extraction and selection. Proc Comput Sci 85:455–462

41. Sayed GI, Darwish A, Hassanien AE, Pan JS (2016) Breast cancer diagnosis approach based on meta-heuristic optimization algorithm inspired by the bubble-net hunting strategy of Whales. In: Advances in intelligent systems and computing genetic and evolutionary computing (pp. 306–313)

42. Jona J, Nagaveni N (2014) Ant-cuckoo Colony Optimization for FS in Digital Mammogram. Pak J Biol Sci 17(2):266–271

43. Prechelt L (1994) Proben1: a set of neural network benchmark problems and benchmarking rules. Technical Report, University of Karlsruhe, Karlsruhe, Germany

44. Quinlan JR (1996) Improved Use of Continuous Attributes in C4.5. J Artific Intell Res 4:77–90

45. Hamilton HJ, Shan N, Cercone N (1996) RIAC: a rule induction algorithm based on approximate classification. In: International conference on engineering applications of neural networks, University of Regina

46. Nauck D, Kruse R (1999) Obtaining interpretable fuzzy classification rules from medical data. Artif Intell Med 16(2):149–169

47. Peña-Reyes CA, Sipper M (1999) A fuzzy-genetic approach to breast cancer diagnosis. Artif Intell Med 17(2):131–155

48. Setiono R (2000) Generating concise and accurate classification rules for breast cancer diagnosis. Artif Intell Med 18(3):205–219

49. Albrecht A, Lappas G, Vinterbo S, Wong C, Ohno-Machado L (2002) Two applications of the LSA machine. In: Proceedings of the 9th international conference on neural information processing (pp. 184–189). ICONIP 02

50. Fogel DB, Wasson EC, Boughton EM (1995) Evolving neural networks for detecting breast cancer. Cancer Lett 96(1):49–53

51. Abonyi J, Szeifert F (2003) Supervised fuzzy clustering for the identification of fuzzy classifiers. Pattern Recogn Lett 24(14):2195–2207

52. Polat K, Güneş S (2007) Breast cancer diagnosis using least square support vector machine. Dig Sign Process 17(4):694–701

53. Guijarro-Berdiñas B, Fontenla-Romero O, Pérez-Sánchez B, Fraguela P (2007) "A linear learning method for multilayer perceptrons using least-squares", intelligent data engineering and automated learning - ideaL. Lect Notes Comput Sci 4881:365–374

54. Stoean R, Stoean C (2013) Modeling medical decision making by support vector machines, explaining by rules of evolutionary algorithms with FS. Expert Syst Appl 40(7):2677–2686

55. Ahmad F, Isa NAM, Hussain Z, Osman MK, Sulaiman SN (2014) A GA-based FS and parameter optimization of an ANN in diagnosing breast cancer. Pattern Anal Appl 18(4):861–870

56. Karthik S, Perumal RS, Chandra Mouli PVSSR (2018) Breast cancer classification using deep neural networks. In: Knowledge Computing and Its Applications (pp. 227–241)

57. Bamakan SMH, Gholami P (2014) A novel FS method based on an integrated data envelopment analysis and entropy model. Proc Comput Sci 31:632–638

58. Xue B, Zhang M, Browne WN (2012) New fitness functions in binary particle swarm optimisation for FS. In: 2012 IEEE congress on evolutionary computation (pp. 1–8)

59. Xue B, Zhang M, Browne WN (2014) Particle swarm optimisation for FS in classification: Novel initialisation and updating mechanisms. Appl Soft Comput 18:261–276

60. Maldonado S, Weber R, Basak J (2011) Simultaneous FS and classification using kernel-penalized support vector machines. Inf Sci 181(1):115–128

61. Miao D, Gao C, Zhang N, Zhang Z (2011) Diverse reduct sub-spaces-based co-training for partially labeled data. Int J Approx Reason 52(8):1103–1117

62. Luukka P, Leppälampi T (2006) Similarity classifier with generalized mean applied to medical data. Comput Biol Med 36(9):1026–1040

63. Sheikhpour R, Sarram MA, Sheikhpour R (2016) Particle swarm optimization for bandwidth determination and FS of kernel density estimation-based classifiers in diagnosis of breast cancer. Appl Soft Comput 40:113–131

64. Belciug S, Gorunescu F (2012) A hybrid neural network/genetic algorithm applied to breast cancer detection and recurrence. Expert Syst 30(3):243–254

65. Salama GI, Abdelhalim MB, Zeid MAE (2012) Experimental comparison of classifiers for breast cancer diagnosis. In: 2012 Seventh International Conference on Computer Engineering & Systems (ICCES)

66. Sridevi T, Murugan A (2014) A novel FS method for effective breast cancer diagnosis and prognosis. Int J Comput Appl 88(11):28–33

67. Wang W, Yang L-J, Xie Y-T, An Y-W (2014) Edge detection of infrared image with CNN_DGA algorithm. Optik 125(1):464–467

68. Liu X, Tang J (2014) Mass Classification in Mammograms Using Selected Geometry and Texture Features, and a New SVM-Based FS Method. IEEE Syst J 8(3):910–920

69. Saki F, Tahmasbi A, Soltanian-Zadeh H, Shokouhi SB (2013) Fast opposite weight learning rules with application in breast cancer diagnosis. Comput Biol Med 43(1):32–41

70. Buciu I, Gacsadi A (2011) Directional features for automatic tumor classification of mammogram images. Biomed Signal Process Control 6(4):370–378

71. Tahmasbi A, Saki F, Shokouhi SB (2011) Classification of benign and malignant masses based on Zernike moments. Comput Biol Med 41(8):726–735

72. Tahmasbi A, Saki F, Shokouhi SB (2010) Mass diagnosis in mammographyimages using novel FTRD features. In: 2010 17th Iranian Conference of Biomedical Engineering (ICBME)

73. Zhang Y, Tomuro N, Furst J, Raicu DS (2011) Building an ensemble system for diagnosing masses in mammograms. Int J Comput Assist Radiol Surg 7(2):323–329

74. Verma B, Mcleod P, Klevansky A (2010) Classification of benign and malignant patterns in digital mammograms for the diagnosis of breast cancer. Expert Syst Appl 37(4):3344–3351

75. Verma B, Mcleod P, Klevansky A (2009) A novel soft cluster neural network for the classification of suspicious areas in digital mammograms. Pattern Recogn 42(9):1845–1852

76. Rojas-Dominguez A, Nandi AK (2009) Development of tolerant features for characterization of masses in mammograms. Comput Biol Med 39(8):678–688

77. Dheeba J, Selvi ST (2011) A swarm optimized neural network system for classification of microcalcification in mammograms. J Med Syst 36(5):3051–3061

78. Dheeba J, Singh NA, Selvi ST (2014) Computer-aided detection of breast cancer on mammograms: A swarm intelligence optimized wavelet neural network approach. J Biomed Inform 49:45–52

79. Rouhi R, Jafari M, Kasaei S, Keshavarzian P (2015) Benign and malignant breast tumors classification based on region growing and CNN segmentation. Expert Syst Appl 42(3):990–1002

80. Rampun A, Morrow PJ, Scotney BW, Winder J (2017) Fully automated breast boundary and pectoral muscle segmentation in mammograms. Artif Intell Med 79:28–41

81. Ribli D, Horváth A, Unger Z, Pollner P, Csabai I (2018) Detecting and classifying lesions in mammograms with Deep Learning, Sci Rep 8(1)

82. Kermani BG, White MW, Nagle HT (1995) Feature extraction by genetic algorithms for neural networks in breast cancer classification. In: Proceedings of the 17th annual conference on IEEE engineering in medicine and biology society (pp. 8311–832), vol 831

83. Verma B, Zhang P (2007) A novel neural-genetic algorithm to find the most significant combination of features in digital mammograms. Appl Soft Comput 7(2):612–625

84. Abbass HA (2002) An evolutionary ANNs approach for breast cancer diagnosis. Artif Intell Med 25(3):265–281

85. Dhanya R, Paul IR, Akula SS, Sivakumar M, Nair JJ (2020) F-test FS in Stacking ensemble model for breast cancer prediction. Proced Comput Sci 171:1561–1570

86. Supriya M, Deepa AJ (2019) A novel approach for breast cancer prediction using optimized ANN classifier based on big data environment. Health care Manage Sci, 1-13

87. https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(original)

88. https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

89. http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Prognostic)

90. Rebecca Sawyer Lee, Francisco Gimenez, Assaf Hoogi , Daniel Rubin (2016). Curated Breast Imaging Subset of DDSM . The Cancer Imaging Archive.

91. Suckling J, Parker J, Dance D, Astley S, Hutt I, Boggis C, Ricketts I et al (2015) MIAS (MIAS) database v1.21 . https://www.repository.cam.ac.uk/handle/1810/250394

92. http://medicalresearch.inescportopt/breastresearch/GetINbreastDatabase.html

93. Siddavaatam P, Sedaghat R (2020) A novel multi-objective optimizer framework for TDMA-based medium access control in IoT. CSIT 8:319–330. https://doi.org/10.1007/s40012-020-00283-7

94. Dalwinder S, Birmohan S, Manpreet K (2020) Simultaneous feature weighting and parameter determination of Neural Networks using Ant Lion Optimization for the classification of breast cancer. Biocyber Biomed Eng 40(1):337–351

95. Derangula A, Edara SR (2021) Identification of optimized features using nature-inspired meta-herustics based optimizations in breast cancer detection. In: Materials Today: Proceedings

96. Nayak M, Das S, Bhanja U, Senapati MR (2020) Elephant herding optimization technique based neural network for cancer prediction. Inf Med Unlock 21:100445

97. Ghanem WAHM, Jantan A (2018) A cognitively inspired hybridization of artificial bee colony and dragonfly algorithms for training multi-layer perceptrons. Cogn Comput 10(6):1096–1134