JOIV : Int. J. Inform. Visualization, 9(2) - March 2025 658-673

INTERNATIONAL JOURNAL
ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Recent Advances in Meta-heuristic Algorithms for Training Multilayer Perceptron Neural Networks

Maher Talal Al-Asaady [a,b], Teh Noranis Mohd Aris [a,*], Nurfadhlina Mohd Sharef [a], Hazlina Hamdan [a]

[a] *Department of Computer Science, Faculty of Computer Sciences and Information Technology, Universiti Putra Malaysia, Selangor, Malaysia*
[b] *Department of Network and Computer Software Techniques, Technical Institute Mosul, Northern Technical University, Mosul, Iraq*
*Corresponding author: *nuranis@upm.edu.my*

***Abstract*—Artificial Neural Networks (ANNs) have demonstrated applicability and effectiveness in several domains, including classification tasks. Researchers have emphasized the training techniques of ANNs to identify appropriate weights and biases. However, conventional training techniques such as Gradient Descent (GD) and Backpropagation (BP) often suffer from early convergence, dependence on initial parameters, and susceptibility to local optima, limiting their efficiency in complex, high-dimensional problems. Meta-heuristic algorithms (MHAs) offer a promising alternative as practical approaches for training ANNs, providing global search capabilities, robustness, and improved computational efficiency. Despite the growing use of MHAs, existing studies often focus on specific subsets of algorithms or narrow application domains, leaving a gap in understanding their comprehensive potential and comparative performance across diverse classification tasks. This paper addresses this gap by presenting a systematic review of advancements in training Multilayer Perceptron (MLP) neural networks using MHAs, analyzing 53 publications from 2014 to 2024. The research papers were chosen explicitly from four widely used databases: ScienceDirect, Scopus, Springer, and IEEE Xplore. Key contributions include a comparative analysis of evolutionary, swarm intelligence, physics-based, human-inspired algorithms, and hybrid approaches benchmarked on classification datasets. The study also highlights bibliometric trends, identifies underexplored areas such as adaptive and hybrid algorithms, and emphasizes the practical application of MHAs in optimizing ANN performance. This work is a significant resource for researchers, facilitating the identification of effective optimization methodologies and bridging the gap between theoretical advancements and real-world applications.**

***Keywords*—Artificial neural network training; classification; multilayer perceptron; meta-heuristic algorithms; optimization.**

Manuscript received 11 Sep. 2024; revised 20 Nov. 2024; accepted 22 Dec. 2024. Date of publication 31 Mar. 2025.

I. INTRODUCTION

Artificial Neural Networks (ANNs) are computational models inspired by the structure and functionality of the human neural system, designed to address complex tasks such as classification, regression, and prediction across diverse domains [1]. Among the many types of ANNs, Multi-Layer Perceptrons (MLPs) are particularly popular due to their flexibility in handling different tasks [2]. However, despite their superiority, the performance of ANNs, including MLPs, is highly dependent on the success of the training process, where the weights and biases of the network are adjusted to minimize the difference between predicted and actual outputs [3], [4], [5].

Traditional training methods, such as Backpropagation (BP), are the standard approaches for training ANNs that rely on gradient-based optimization to minimize error. However, these methods have limitations, including slow convergence, susceptibility to local optima, and high dependence on initial parameters [6], [7]. These issues, particularly evident in non-convex loss functions, often result in unstable training or suboptimal performance [8], [9], [10]. These limitations highlight the need for advanced optimization techniques to enhance performance in high-dimensional, complex problems.

Meta-heuristic Algorithms (MHAs) have emerged as a robust alternative to traditional gradient-based methods for training ANNs, overcoming many limitations, such as sensitivity to initial conditions and reliance on derivative information [11]. Inspired by natural processes like evolution, swarm intelligence, and social dynamics, MHAs employ global optimization strategies to explore solution spaces efficiently, avoid local minima, and handle complex, non-convex optimization problems [12]. Their flexibility allows them to adapt to diverse ANN training scenarios without extensive customization [13], while their balance of exploration (searching new regions) and exploitation (refining existing solutions) ensures faster and more reliable convergence [14]. These strengths make MHAs a powerful tool for addressing the challenges of ANN training.

Over the past two decades, a variety of MHAs, such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Grey Wolf Optimizer (GWO), have been employed for training ANNs, showcasing their effectiveness in enhancing classification accuracy, convergence speed, and robustness across diverse applications [15]However, while using MHAs for training ANNs has seen significant advancements, gaps still need to be found in the literature. Comprehensive studies are required to holistically evaluate all MHA categories and provide unbiased performance comparisons based on existing research. Such a review is essential to better understand the strengths and limitations of different algorithms and to guide future research in this field.

Several prior reviews have explored using MHAs in ANN training, focusing primarily on a subset of well-established algorithms—for instance, Si et al. [16] conducted an experimental study on 13 MHAs, concentrating mainly on medical data classification. Similarly, Mousavirad et al. [17] reviewed 15 algorithms for ANN training but limited their review to older methods such as PSO, GA, and Differential Evolution (DE). Another study by Emambocus et al. [18] focused exclusively on swarm intelligence algorithms, comprehensively reviewing how swarm-based methods have optimized ANN structures and training. While these reviews provide valuable insights into traditional MHAs, they often focus on specific application domains, such as medical datasets or benchmark tasks, limiting their generalizability to broader contexts.

Furthermore, many earlier studies overlooked the latest advancements in the field. For example, the growing number of emerging MHAs, which have demonstrated significant potential in recent years, are rarely discussed compared to older methods [19]. Additionally, while comparative studies exist, many reviews need a systematic methodology, making their findings difficult to generalize or reproduce.

This study addresses critical gaps in the literature by systematically evaluating major categories of MHAs, including hybrid, physics-based, and human-inspired algorithms, thereby offering a comprehensive and inclusive perspective. Applying bibliometric tools such as R-tool and VOS viewer identifies key trends, research gaps, and the interconnected nature of MHA research. Furthermore, the study emphasizes the practical implementation of MHAs for classification tasks across diverse datasets, focusing on essential evaluation metrics such as accuracy, Mean Square Error (MSE), and sensitivity.

This paper provides a comprehensive and up-to-date survey of MHAs for ANN training, focusing on established and newly developed algorithms. The review includes an analysis of recent advancements in MHA research, highlighting algorithms introduced over the past decade. Specifically, the study evaluates the performance of various MHAs, including evolutionary algorithms, swarm intelligence algorithms, physics-based algorithms, and human-inspired algorithms, in the context of ANN training for classification tasks across diverse domains. Additionally, a trend analysis (Figure 1) of research publications from 2004 to 2024, based on data from Scopus, illustrates the increasing interest in MHAs for ANN optimization, particularly in the areas of ANNs, optimization, and classification. The analysis reveals a steady rise in publications, with a significant surge in recent years, underscoring the growing recognition of MHAs' potential in enhancing ANN performance for classification tasks.

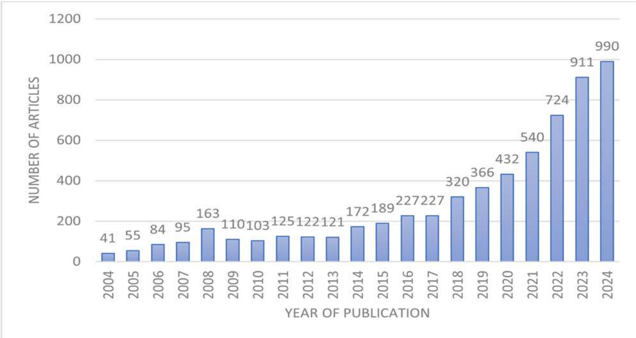

Fig. 1 Annual publication trend for ANN, optimization, and classification keywords (Dec 2024)

Additionally, this study employs a systematic review methodology to ensure rigor and reproducibility when comparing the performance of various MHAs. Conducting a comprehensive evaluation provides actionable insights into the most effective algorithms for ANN training. The findings aim to enhance researchers' and practitioners' understanding of the strengths and limitations of different MHAs, thereby establishing a foundation for future exploration of adaptive and hybrid approaches to optimize neural network performance.

The rest of this article is organized as follows: Section II overviews the materials and study method. Section III presents the results and discussion. Finally, Section IV concludes the article.

II. Materials and Methods

This study aims to present, identify, and evaluate different MHAs and their strategies for ANN training for classification tasks. We followed a structured methodology proposed by [20] and clearly defined guidelines to achieve this objective. Additionally, our research was conducted in alignment with the PRISMA guidelines for systematic reviews, as detailed in [21].

A. Systematic Process

Fig. 2 shows the general approach to identifying the relevant literature and highlights the main steps underpinning the study's literature review process. These include study identification, title, abstract screening, inclusion and exclusion criteria application, and a full-text review. Compliance with these criteria guarantees the most rigid approach to selecting sources and their further analysis, using only the most relevant studies. The above methodology is explained in the following sub-sections.

1) Search Query: The search process starts in December 2023. Specific keywords, such as ‘artificial neural network,’ ‘feedforward,’ ‘multilayer perception,’ ‘training,’ meta-heuristic algorithms’, and ‘classification,’ were used to find appropriate articles. The Boolean operators fine-tuned the search queries "AND" and "OR."

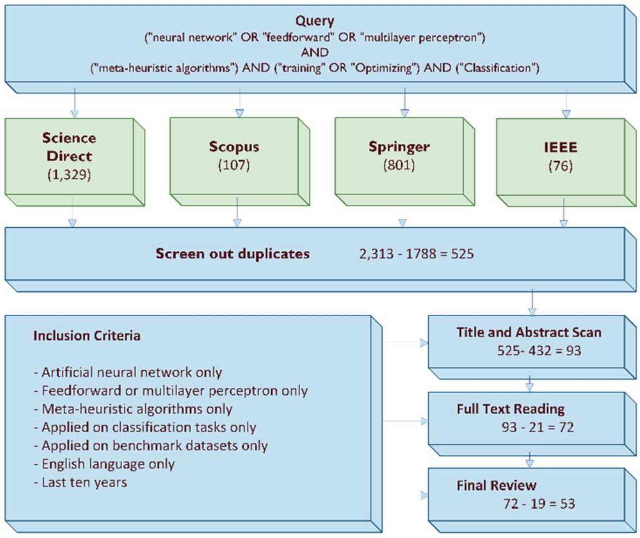

Fig. 2 A PRISMA flow diagram for study selection with the research query and inclusion criteria

2) Information Sources: A comprehensive literature search was conducted across four databases: Scopus, Springer, ScienceDirect, and IEEE Xplore. The search queries are guided entirely to ensure that all the chosen articles meet the required quality factors of uniqueness, a high citation index, and high impact.

3) Study Selection: The study selection process was conducted in two phases to ensure rigor and transparency.

- Initial screening: Articles were filtered based on their titles and abstracts to exclude irrelevant studies. Duplicate studies were also removed.
- Full-Text review: The remaining articles were assessed for relevance and adherence to the inclusion criteria.

Table I summarizes the reasons for exclusion at each stage, detailing the number of articles removed and ensuring adherence to the eligibility process.

TABLE I
SUMMARY OF REASONS FOR EXCLUSION AT EACH STAGE

Stage	#Articles	Reasons for Exclusion
Initial Screening	1,788	Irrelevant based on title/abstract.
Full-Text Review	93	Studies outside the scope, non-English, and other ANN types.
Final Inclusion	53	Met all inclusion criteria.

4) Eligibility Criteria: To ensure the inclusion of high-quality and relevant studies, specific eligibility criteria were established, and a systematic study selection process was followed. Table II illustrates the requirements that guided the inclusion and exclusion of studies.

5) Search Results: The first search returned 2,313 articles, with 1,329 from ScienceDirect, 107 from Scopus, 801 from Springer, and 76 from IEEE Xplore, spanning 2014 to 2024. 1788 articles were excluded from the databases utilized because they did not comply with the inclusion criteria. After scanning the titles and abstracts, 432 unrelated articles were eliminated, leaving 93 papers. 19 articles were eliminated after reading the whole text, leaving 53 papers in the final collection.

TABLE II
CRITERIA FOR INCLUSION AND EXCLUSION

Criteria	Description
Inclusion Criteria	Studies on ANNs (Feedforward Neural Networks and MLPs). Applied to classification tasks. Published between 2014–2024 in peer-reviewed journals or conferences. Applied on benchmark dataset.
Exclusion Criteria	Non-English articles, books, or reports. Studies focus on other neural network types (e.g., Convolution Neural Networks). Regression-focused research or datasets not related to classification.

6) Analytical Tools and Software: Specific analytical tools were employed to enhance the rigor and reproducibility of the review analysis. The bibliometric tools, R-tool and VOSviewer, were selected because they effectively visualize co-occurrence networks and scientific production patterns. These tools supported a systematic representation of the results, aligning with the PRISMA guidelines.

B. Meta-heuristic Algorithms

Meta-heuristic Algorithms (MHAs) are powerful problem-solving methods that find near-optimal solutions across various problems, often at a reasonable computational cost. Unlike traditional algorithms, which aim for optimal solutions in predefined constraints, MHAs navigate complex solution spaces more dynamically [22]. These algorithms are population-based, each inspired by natural, physical, or social processes. The four categories explored in this study include Evolutionary Algorithms (EA), Swarm Intelligence Algorithms (SI), Physics-Based Algorithms (PBA), and Human-Based Algorithms (HBA). Fig. 3 represents these categories and the specific MHAs discussed in this study that are applied to training ANNs.

Fig. 3 Classification of MHAs used to train ANN in this study

1) Evolutionary Algorithms: EAs are practical for solving optimization problems with the help of the "survival of the fittest" principle. They start with a population that evolves across generations, improving through genetic operations like crossover and mutation [23]. Fig. 4 illustrates the general scheme of an EA [24]. The process starts with the initialization of a population of potential solutions. Of this population, the parents are chosen, and then, through crossover and mutation, the offspring come. These offspring are then assessed, and the survival returns to the population. It continues until a stopping condition is reached when the solution is near the optimal solution.

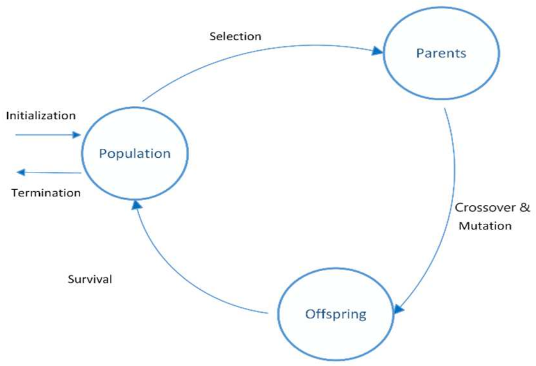

Fig. 4 The general scheme of the evolutionary algorithms

2) Swarm Intelligence Algorithms: SIs are nature-inspired optimization methods developed based on the interaction behavior of living organisms such as animals and insects. These algorithms operate on the principle of decentralization, wherein candidate solutions are updated through local interactions among agents and their environment. Fig. 5 depicts the general scheme of SIs [25]. The process begins with the initialization of a population of agents. Each agent's solution is evaluated using a fitness function. If the stop condition is not reached, the agents are updated and moved to explore the search space further. This loop continues until the stop condition is met; at this point, the algorithm returns the global best solution and terminates.

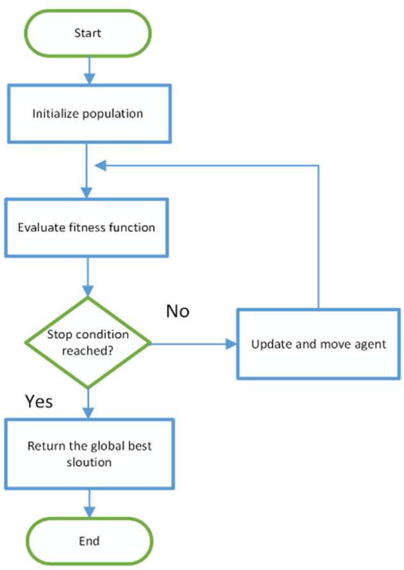

Fig. 5 The general scheme of swarm intelligent algorithms

3) Physics-based Algorithms: PBAs are naturally inspired by physical principles, and the interactions of search agents are modeled according to physical laws. These algorithms mimic gravity, electricity, heat, and waves to control the search process for the best solution [26].

4) Human-based Algorithms: HBAs are computational methods derived from human activities and behavior, including communication, problem-solving, and social relations. These methods copy various human thinking properties to help solve optimization problems. Many HBAs contain learning, cooperation, and adaptation features that make them suitable for solving cases of dynamic optimization problems [26].

C. ANN Training Using MHAs

Given the limitations of traditional training methods, MHAs are seen as promising alternatives to train the weights and biases of ANNs to improve their performance. The first step in training ANN using MHA is problem representation. ANN variables can be a vector, binary string, or a real array whose length equals the sum of all the weights and biases. Fig. 6 illustrates the vector representation of an MLP architecture, mapping its weights and biases into a solution vector [17]. The diagram demonstrates how the connections between input, hidden, and output layers are encoded into a vector format, enabling MHAs to optimize the parameters systematically. Such encoding ensures a structured approach to ANN parameter tuning, contributing to improved training efficiency and accuracy.

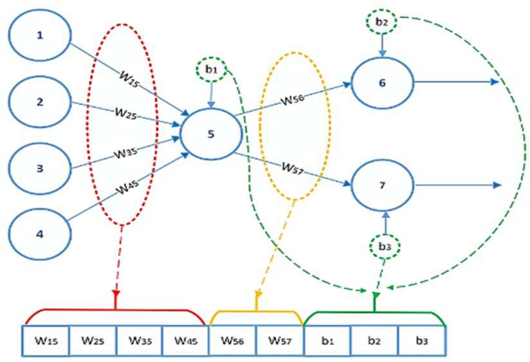

Fig. 6 Vector representation of an MLP architecture

In the representation of MHAs, each individual in the population *(P)* is expressed as a vector in (*D)* dimensions, which is computed using equation (1):

$$D = (N \times M) + (M \times R) + M + R \qquad (1)$$

where *N*, *M*, and *R* represent the corresponding numbers of nodes in the input, hidden, and output layers. The network's connections' weights and biases are included in the vector as shown in equation (2):

$$P_i = \{\overrightarrow{W1}_{NM}, \overrightarrow{B1}_M, \overrightarrow{W2}_{MR}, \overrightarrow{B2}_R\} \; i = 1, \dots NP \qquad (2)$$

where $\overrightarrow{W1}_{NM}$ and $\overrightarrow{W2}_{MR}$ are weights between the input to the hidden layer and the hidden to the output layer. $\overrightarrow{B1}_M$ and $\overrightarrow{B2}_R$ are the biases of the hidden to the output layer, respectively, and *NP* is the total number of agents.

As shown in Fig. 7 [27], training an ANN with MHAs begins by randomly initializing weights and biases. The training data are then fed into the network, and the MLP generates outputs compared to the target values using an objective function. This comparison produces an error used to calculate a fitness value, typically measured by error metrics such as Mean Square Error (MSE) or Root Mean Square Error (RMSE). Based on this error, the MHA iteratively adjusts the weights and biases to minimize the fitness value and improve performance. This process continues until specific stopping criteria are met, and the solution with the optimal fitness value, indicating the lowest error, is selected for future classification tasks.

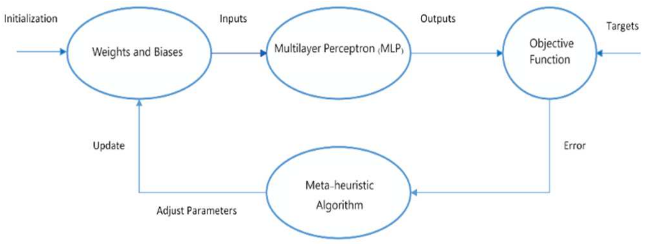

Fig. 7 ANN training process using MHA

III. Results and Discussion

In this section, we present the results of our study on applying various MHAs for training ANNs. The findings are divided into four algorithm groups: EAs, SIs, PBAs, and HBAs.

A. Evolutionary Algorithms for ANN Training

1) Genetic Algorithm (GA): GA is an algorithm derived from the natural selection principle and evolution [23]The initial population is generated with random assignments and then assessed using the fitness function. GA contains three operators: selection (deciding which individuals should be used), crossover (exchanging information between two selected solutions), and mutation (changing a particular portion of this information). Rojas et al. [13] introduced the Cellular Genetic Algorithm (CGA) with Damped Crossover (DX) for optimizing MLP weights and biases for medical data classification. The method demonstrated outstanding results with lower MSE values compared to other MHAs. Bansal et al. [28] proposed GGA-MLP, using a Greedy Genetic Algorithm (GGA) to train MLPs. The GGA employs a greedy approach for the initial population generation. The approach was applied to ten benchmark medical datasets, some of which had outstanding results.

2) Biogeography-based Optimizer (BBO): BBO is an MHA that uses biogeography principles like species migration and evolution to find optimal solutions [29]. Mirjalili et al. [30] presented a BBO approach to training MLPs. The study findings indicated a good performance, especially in avoiding local minima. Another study by Zhang et al. [31] presents an enhanced BBO approach for training MLPs. The advanced strategy introduced various probability distributions, which include gamma, beta, and Gaussian, to migrate the BBO algorithm, whereas the prior method had the random distribution and known migration probability.

3) Artificial Algae Algorithm (AAA): AAA mimics the behavior of microalgae specifically for the light-seeking behavior of photosynthesis [32]. Turkoglu and Kaya [33] presented a study using AAA to train MLPs. AAA showed superior performance in classification tasks compared to BP and other methods. Another survey by Karakoyun [34] presents a modified version of the AAA employed for ANN training. The new improvement uses a multi-selection in the position update step to improve how the algorithm avoids local optima and generally brings better results to the search.

EAs have been widely utilized in ANN training due to their ability to optimize complex, non-linear problems. They work by evolving populations and breeding superior individuals to avoid local minima. Table III highlights the application of various EAs in ANN training, providing detailed insights into the specific algorithms employed, key metrics, significant findings such as enhanced convergence speed or improved classification accuracy, and the Interpretation field, which explains the methodologies and mechanisms that contributed to these outcomes.

TABLE III
Overview of Studies Utilizing Evolutionary Algorithms for Training Multilayer Perceptron Neural Networks

Ref.	MHA	Metrics	Key Findings	Interpretation
[13]	GA	Accuracy Specificity Sensitivity	Cellular Genetic Algorithm-Damped Crossover (CGA-DX) showed lower MSE values than other MHAs.	A novel mutation mechanism with a modified version of GA leads to improved results.
[28]	GA	Accuracy Specificity Sensitivity	Greedy Genetic Algorithm (GGA) outperformed traditional MLP on several medical datasets.	Enhanced initial population selection ensures better convergence for domain-specific tasks like medical datasets.
[30]	BBO	Accuracy	Effective in avoiding local minima, enhancing MLP performance.	Migration strategies improve global search but can be sensitive to parameter tuning.
[31]	BBO	Area Under Curve (AUC) Sensitivity Specificity	Effective in avoiding local minima and enhancing performance.	Migration strategies improve global search but can be sensitive to parameter tuning.
[33]	AAA	Accuracy Wilcoxon Friedman	A superior performance compared to BP and others.	Mimics algae's adaptive behavior for adequate classification. Less validated in larger datasets.
[34]	AAA	Sensitivity Specificity Accuracy MAE R2	Avoids local optima and has better performance in in some datasets.	Balances exploration and exploitation, making it versatile but sensitive to parameter fine-tuning.

Table IV provides a comparative overview of these EAs, highlighting their advantages and disadvantages as reported in the reviewed studies [35], [36]This comparison offers valuable insights into each algorithm's strengths and limitations, aiding in selecting the most suitable approach for specific ANN training tasks.

TABLE IV
COMPARATIVE ANALYSIS OF EVOLUTIONARY ALGORITHMS FOR TRAINING ANN MODELS

MHA	Advantages	Disadvantages
GA	Good coverage of initial solutions. Easy to implement.	Can converge to local minima. Performance depends on tuning parameters.
BBO	Fewer parameters to tune. Sharing information between solutions, helping avoid local minima.	Sensitive to initial parameters.
AAA	Combines exploration and exploitation. Solving both discrete and continuous problems.	Less studied; hence, there are fewer empirical validations. Sensitive to parameter settings.

B. Swarm Intelligence Algorithms for ANN Training

1) Practical Swarm Optimization (PSO): PSO is an optimization technique emulating the social behaviors of birds, flocks, or fish schooling [37]. It involves finding the solution to the problem within the search space using a population of candidate solutions called particles. Rauf et al. [11] present a novel initialization technique called (PSOLL-NN) to improve the training of FNNs. PSOLL-NN improved the training of FNNs compared to standard PSO and BP.

2) Animal Migration Optimization (AMO): The AMO algorithm simulates the movement of animals from one location to another in the search for a resource or better standards of living and iteratively scans the search space in an optimal problem solution [38]. Gülcü et al. [39] improved the AMO algorithm by integrating the function based on Lévy flight to train MLPs. Thus, the IAMO-MLP algorithm optimizes the search and acquisition process and provides a good way to avoid being trapped in local optima.

3) Grey Wolf Optimization (GWO): GWO, inspired by the social structure and hunting tactics of grey wolves, emulates the roles of alpha, beta, delta, and omega wolves within a pack to guide the search process toward optimal solutions [40]. Mirjalili [4] applied GWO to train MLPs. The performance was assisted by using diverse datasets for classification and function approximation. GWO demonstrated superior performance in training MLPs compared to traditional methods. Similarly, Altay and Varol [41] presented an improved GWO (IMP-GWO), which addressed standard GWO's limitations, such as premature convergence and exploration-exploitation balance. The IMP-GWO-MLP method showcased superior performance compared to conventional optimization techniques.

4) Tree-Seed Algorithm (TSA): TSA is inspired by nature, as in the case of tree seed propagation, resulting in the growth of seeds into mature trees by adapting to environmental factors [42]. Cinar [43] developed an approach to train MLPs using TSA. This approach compared results with other algorithms using two statistical analysis techniques: the Wilcoxon Signed Rank Test and Friedman's Test. The study stresses TSA's exploration capabilities and advises more research into the improved versions of TSA for MLP training.

5) Ant Lion Optimizer (ALO): ALO symbolically represents antlions as candidate solutions, with their hunting strategies guiding the optimization process within the search space [44]. Yamany et al. [45] applied ALO to train MLPs. ALO-MLP was effective in training MLPs, showing improved performance over other algorithms. In addition, ALO was also enhanced by Heidari and Faris (Heidari et al., 2020), and the model's effectiveness was evaluated against DE, GA, PSO, and PBIL. The improved model showed enhanced performance in diverse datasets.

6) Moth-flame Optimization Algorithm (MFO): MFO is an algorithm designed to replicate the flying behavior of a moth's flight used in navigation. Transverse orientation is a direction-finding mechanism that moths use to stay oriented relative to light sources [34]. In a paper by Yamany et al. [47], the MFO algorithm was applied to train the MLPs. The model demonstrated practical training of MLPs with better performance in classification tasks. Yang et al. [48] propose the Floating Flame MFO (FMFO) algorithm to avoid problems with local optimization traps and optimize search speed and accuracy in MLP classifier training. Thus, the segmented search improves by learning to combine exploitation and exploration of the FMFO algorithm.

7) Monarch Butterfly Optimization (MBO): The MBO algorithm is based on the behavioral aspects of the monarch butterfly. This algorithm is based on constructive and explorative behavior patterns seen in monarch butterfly migration [49]. Firas et al. [50] proposed an Improved MBO (IMBO) algorithm to train ANNs. The IMBO modifies the position updating mechanism to include previous solutions and the best solution. This modification enhances the algorithm's ability to avoid local optima and accelerates convergence.

8) Crow Search Algorithm (CSA): CSA is based on crows' foraging skills that are closest to crows' habit of storing and finding food [51]. Khamees et al. [52] used CSA to train ML. While GA and PSO algorithms stuck to local minima, the CSA algorithm was much better at sliding into the global minima. Besides, the CSA model has been used by Erdoğan and Gülcü [53], which was thoroughly tried on several classifying datasets. The CSA-CSA-MLP model is compared to different models and found to be the best algorithm for training the MLPs.

9) Whale Optimization Algorithm (WOA): WOA simulates whale behavior using techniques like encircling prey and spiral movements to update the positions of whale agents, which represent potential solutions [54]. Bhesdadiya et al. [55] presented the WOA-MLP approach that shows superiority in terms of better classification results, solutions to the problem of local minima, and quick termination. Likewise, Aljarah et al. [56] have shown their algorithm to be competent at training MLPs on 20 different datasets with varying complexity levels. Kushwah et al. [57] introduce a modified WOA by enhancing the optimization with a roulette wheel selection mechanism, which balances exploration and exploitation. The modified WOA improved convergence and classification accuracy. Similarly, Raziani et al. [58] propose a novel WOA that gives an advanced nonlinear function to enhance the exploration and exploitation stages and tackle the prevalent problems of local optima and slow convergence resulting from random initialization of weights and biases. In another study, Chatterjee et al. [59] combined chaotic functions and oppositional-based learning with the WOA model to train FNNs. The modified WOA addressed local optima issues and showed fast convergence.

10) Dragonfly Algorithm (DA): DA is inspired by the group behavior observed in dragonflies, where agents simulate natural dragonfly interactions to resolve optimization problems [60]. Abo-Elsoud et al. [61] used DA to train MLP. DA-MLP-based optimization compared favorably with traditional methods such as GA, ACO, and PSO, bringing impressive results. In addition, Gülcü [62] proposed a hybrid DA-MLP algorithm as an enhancer technique. The effectiveness of this method was demonstrated on a civil engineering task as well as several datasets involving classification problems.

11) Spotted Hyena Optimizer (SHO): SHO models spotted hyenas' complex social and hunting skills; every hyena in the algorithm symbolizes the possible solutions through the hunting and local processes [63]. Panda and Majhi [64] present a study using SHO to train the MLP. They perform some statistical tests to evaluate whether the SHO-MLP method is superior. Moreover, Luo et al. [65] examined the effect of SHO in enhancing the performance of FNNs in the tested scenarios, including classification and function approximator tasks. They showed and improved efficiency, especially in searching the complex search spaces.

12) Grasshopper Optimization Algorithm (GOA): GOA mimics the behavior of grasshoppers to tackle several real-world optimization problems. Each grasshopper adjusts its position according to its previous position, the best solution encountered, and the collective information shared by neighboring grasshoppers [66]. Heidari et al. [67] utilized the GOA algorithm to train the MLPs. The approach of GOA-MLP is compared with the results of eight other optimization algorithms. The results demonstrated the highest classification of all datasets.

13) Butterfly Optimization Algorithm (BOA): BOA mimics butterflies' random foraging behavior, where the movements are driven by environmental cues and the attractiveness of possible food sources as measures of solutions' fitness features [68]. Jalali et al. [69] proposed an approach using ANN trained by BOA to classify vertebral and Parkinson's diseases. The ANN-BOA model showed superior performance in medical data classification—furthermore, Irmak et al. [70] also proposed a BOA-MLP model that enhanced exploration and exploitation for MLP training. In another study, Irmak et al. [71] proposed an improved BOA incorporating chaotic properties to improve the exploration and exploitation of the search space.

14) Seagull Optimization Algorithm (SOA): SOA mimics the seagulls' approach to finding food and avoiding predators, translating these behaviors into strategies for exploring and exploiting the search space in optimization problems [72]. Bacanin et al. [73] presented a modified version of the SOA to enhance the training of ANNs. The proposed enhanced SOA was then validated on ten binary classification benchmark datasets. The research has shown that the improved SOA offered better total performance than other current algorithms.

15) Mayfly Algorithm (MA): MA mimics mayflies' swarm behavior and life cycle. The male mayflies explore the solution space, and during the period of exploitation, they may find a mate and improve solutions [74]. Moosavi et al. [75] present the MFANN algorithm for training ANNs. The algorithm tested on two benchmark datasets: Banknote Authentication and Cryotherapy. Comparative analysis showed that MFANN outperformed the PSO and GWO. by 1-2% in training accuracy and 2% in testing accuracy.

16) Marine Predators Algorithm (MPA): The MPA algorithm is based on the hunting method of marine predators. Using the same search pattern, it applies optimal solutions in rugged search space optimization [76]. Bagchi et al.[16] applied MPA in improving ANNs for medical data classification, using it in ten benchmark datasets that opposed its results to those of the LM method and PSO. Likewise, Zhang and Xu [77] proposed an MPA variation with a ranking-based mutation operator. This operator helps lead the identification of the optimal agent, which searches avoid stagnation, and as a result, the convergence rates improve.

17) Coot Optimization Algorithm (COOT): COOT took ideas after the coot bird creole's foraging behaviors. The algorithm recognizes differences among leaders (high-quality solutions) and members (lower-quality solutions) within the flock [78]. Özde and İşeri [79] focused on COOT's use in training MLPs and classifying benchmark datasets. The COOT-ANN method was benchmarked against the classical optimizer methods like GD and the Levenberg-Marquardt algorithm.

SI algorithms are inspired by the characteristics, habits, and properties of different animals and insects and offer unique advantages in ANN training by leveraging decentralized interactions. Table V highlights the application of various SI algorithms, detailing their metrics, key findings, and interpretation, which explains the mechanisms driving their effectiveness. Table VI provides a comparative overview of SI algorithms, focusing on their specific strengths and their limitations based on previous studies [36], [80], [81], [82].

TABLE V
OVERVIEW OF STUDIES UTILIZING SWARM INTELLIGENCE ALGORITHMS FOR TRAINING MULTILAYER PERCEPTRON NEURAL NETWORKS

Ref.	MHA	Metrics	Key Findings	Interpretation
[11]	PSO	Accuracy	The proposed PSO Log Logistic (PSOLL-NN) approach improved the training of FNNs compared to standard PSO-NN and BP.	Practical and straightforward, it may need help with premature convergence in high-dimensional problems.
[39]	AMO	Sensitivity Specificity Accuracy	Enhanced search capabilities and avoided local optima effectively.	Adaptive movement mimics ant behavior, ensuring robust optimization but slower convergence.

Ref.	MHA	Metrics	Key Findings	Interpretation
[4]	GWO	Accuracy MSE	Superior performance in training MLPs compared to traditional methods.	Balances exploration and exploitation effectively, reducing the chance of local minima.
[41]	IGWO	Accuracy Precision F1	Addressed limitations of standard GWO, showing better performance and convergence.	Improved GWO variant with enhanced accuracy but slightly increased computational cost.
[43]	TSA	Accuracy MSE Wilcoxon Friedman	Excellent exploration capabilities and improved MLP training performance.	Excels in exploring solution spaces but is less effective in exploitation than other SIs.
[47]	ALO	Accuracy MSE	Improved performance over other algorithms.	Spiral-based movement enhances global optimization, but convergence is relatively slow.
[46]	ALO	Accuracy MSE	Improved ALO showed enhanced performance in diverse datasets.	Adaptive updates improve generalization, but high computational costs limit scalability.
[47]	MFO	Classifica-tion Rate Test Error	Demonstrated better performance in some datasets.	Compelling flame-based attraction prevents local minima but works best with smaller datasets.
[48]	FMFO	MSE Accuracy Speed Time	Improved search speed and accuracy in MLP training.	Combines flame-based search with adaptive behavior for fast, precise results.
[50]	IMBO	Accuracy Standard Deviation Speed	Enhanced convergence and avoided local optima.	Iterative migration ensures robust convergence and scalability for large datasets.
[52]	CSA	Accuracy MSE	Outperformed GA and PSO in avoiding local minima.	Ideal for escaping local minima; it requires careful parameter tuning.
[53]	CSA	Accuracy Sensitivity Specificity	Superior performance compared to other models.	Ideal for escaping local minima; requires careful parameter tuning.
[55]	WOA	Accuracy MSE	Better classification results and quick termination.	Whale-inspired search effectively balances speed and accuracy in classification tasks.
[56]	WOA	Accuracy	Improved convergence and classification accuracy.	Exploits bubble-net for efficient convergence but struggles with parameter-heavy datasets.
[57]	MWOA	Accuracy MSE Time	Modified WOA improved convergence and classification accuracy.	Adaptive updates improve real-time performance, albeit with higher computational demand.
[58]	MWOA	Accuracy AUC Specificity Sensitivity	Modified WOA addressed local optima issues and showed fast convergence.	Robust for avoiding local minima; particularly effective for medical datasets.
[59]	CWOA	Accuracy Sensitivity Specificity	Improved performance in training FNNs with chaotic functions and oppositional-based learning.	Incorporates chaotic theory to enhance global search and exploration capabilities.
[61]	DA	Accuracy MSE	Impressive results compared to traditional methods.	Uses dragonfly-inspired group behavior to balance exploration and exploitation effectively.
[62]	DA	Accuracy MSE	Enhanced performance in classification tasks.	Uses dragonfly-inspired group behavior to balance exploration and exploitation effectively.
[64]	SHO	Accuracy RMSE	Statistical superiority in training MLPs.	Performs well in high-dimensional tasks but is computationally intensive.
[65]	SHO	Accuracy MSE	Improved FNN performance in various scenarios.	Robust in diverse scenarios but requires fine-tuning of parameters for optimal results.
[67]	GOA	AUC Accuracy Specificity Sensitivity,	GOA-MLP approach demonstrated high classification accuracy.	Grasshopper-based movement excels in classification tasks but is resource-intensive for large datasets.
[69]	BOA	AUC Accuracy Specificity Sensitivity	Superior performance in medical data classification.	Butterfly movements adapt well to medical datasets; parameter sensitivity may require fine adjustments.
[70]	BOA	Accuracy Specificity Sensitivity F1, MSE	Improved BOA enhanced exploration and exploitation for MLP training.	Combines adaptive attraction mechanisms for balanced optimization, but exploitation could be more robust.

Ref.	MHA	Metrics	Key Findings	Interpretation
[71]	IBOA	Sensitivity Specificity Precision, F1 Friedman	Improved BOA incorporated chaotic properties for better search performance.	Chaotic strategies boost search efficiency; implementation complexity can be a drawback.
[73]	SOA	Accuracy Standard Deviation Speed	Improved total performance in binary classification tasks.	Reliable for binary classification but slower in datasets with high variability.
[75]	MA	Accuracy	Outperformed PSO and GWO in training accuracy and testing accuracy.	Strong balance of exploration and exploitation; it requires parameter adjustments for optimal results.
[16]	MPA	AUC Accuracy Specificity Sensitivity	Superior performance in medical data classification.	Robust for medical datasets, but computational complexity may limit scalability.
[77]	MPA	Accuracy p-value	Improved search performance and convergence rates.	Reliable for high-accuracy classifications; requires significant computational resources.
[79]	COOT	Accuracy	Outperformed classical optimization methods.	Models cooperative optimization; limited validation in diverse datasets.

TABLE VI
COMPARATIVE ANALYSIS OF SWARM INTELLIGENCE ALGORITHMS FOR TRAINING ANN MODELS

MHA	Advantages	Disadvantages
PSO	Converges quickly in the early stages. Requires fewer parameters.	May get trapped in local minima. Premature convergence.
AMO	Adjusts migration behavior automatically, enhancing performance.	Slow convergence.
GWO	Fewer parameters. Simplicity in implementation.	May suffer from premature convergence.
TSA	Strong exploration capabilities.	Less effective in exploitation.
ALO	Strong global search capabilities.	Slow convergence rate. Complexity in parameter tuning.
MFO	Avoid local minima. Faster convergence.	May suffer from premature convergence.
MBO	Enhance global exploration.	Slow convergence.
CSA	The use of memory to store the best solutions helps improve performance.	May require parameter tuning.
WOA	Efficient in global optimization. Simple to implement.	Risk of premature convergence. May be stuck at local optima.
DA	Efficiently balances exploration and exploitation.	Complexity in modeling behavior.
SHO	Robust against local optima.	Computational complexity.
GOA	Strong global search Dynamic adaptation.	Increased computation time.
BOA	High exploration capability. Effective for global optimization.	Sensitivity to parameter settings.
SOA	Effectively explores the search space.	May converge slowly.
MA	Effectively combines exploration and exploitation.	Depends on careful tuning of attraction.
MPA	High exploration and exploitation.	Complex parameter tuning. Computationally expensive.
COOT	High efficiency in exploration. Easy to implement.	May get stuck in local optima. Parameter sensitivity.

C. Physics-Based Algorithms for ANN Training

1) Gravitational Search Algorithm (GSA): GSA is a model based on principles of geophysics, such as the law of gravity and mass interactions. It employs gravity's mass, weight, and force to quantitatively traverse the search space and arrive at the ideal solutions. [83]. Rather et al. [84] present an enhanced GSA model, which combines Lévy flight and chaos theory to train MLPs. Lévy flight enhances the diversification of the search space, while chaotic maps intensify candidate solutions towards the global optimum.

2) Hypercube Optimization Search (HOS): HOS is inspired by doves' foraging behavior and tailored for high-dimensional numerical optimization tasks [85]. Tunay et al. [86] applied HOS to train MLPs. The findings demonstrated the HOS-MLP model's superior performance in MSE, classification accuracy, and convergence speed, highlighting its potential as an effective decision-support tool in medical applications.

3) Lighting Search Algorithm (LSA): LSA imitates the occurrence of lightning and the step leader traveling mechanism by using the projectiles concept that travels fast and are particles [87]. Aljarah et al. [88] used LSA to train MLPs. The LSA-MLP performance had been compared to a range of machine learning algorithms, and the findings showed that the LSA could escape from the local minima and achieve fast convergence.

4) Sine Cosine Algorithm (SCA): SCA uses sine and cosine functions as inspirational roots to move into the problem of optimization [89]. The paper by Gupta and Deep [90] showed the utilization of SCA and its structural change in the classifier adaptation version technique (HSCA) to train the ANN. In contrast with other optimization techniques such as PSO and ACO, they noted SCA as having better performance, especially in datasets like Breast Cancer.

5) Multi-Verse Optimizer (MVO): MVO utilizes the concept of the Multiverse Cosmology, in which different universes function under disparate principles, to identify and solve optimization problems [91]. Faris et al. [92] used the MVO algorithm to train MLPs evaluated on bio-medical datasets. MVO performed better in representing local optima and achieving steady convergence than GA and PSO. In another study, Hassanin et al. [93] examine MVO as the training technique for FNN and establish its effectiveness on XOR and Breast Cancer datasets as some function approximation tasks. At the same time, the results display MVO-FNN as the superior option for all such experiments.

The laws and principles of natural physical phenomena inspire PBAs. These algorithms, studied extensively in the literature, are frequently applied to complex applications, such as engineering design and optimization. Table VII presents a summary of PBAs applied in ANN training. It details the specific algorithms, performance metrics, key findings, and the interpretation field, which provides insights into the mechanisms that drive their effectiveness. Table VIII presents a comparative analysis of PBAs, focusing on their advantages and disadvantages [81] [82].

TABLE VII
OVERVIEW OF STUDIES UTILIZING PHYSICS-BASED ALGORITHMS FOR TRAINING MULTILAYER PERCEPTRON NEURAL NETWORKS

Ref.	MHA	Metrics	Key Findings	Interpretation
[84]	GSA	MSE Convergence Rate	Superior performance in training MLPs with Lévy flight and chaos theory.	Chaos theory enhances search, improving performance in complex spaces.
[86]	HOS	Accuracy MSE	Superior performance in medical applications.	Tailored for domain-specific tasks, ensuring precision.
[88]	LSA	Accuracy p-value	Fast convergence and effective avoidance of local minima.	Mimics natural phenomena, ensuring rapid convergence while avoiding local minima.
[90]	SCA	Accuracy MSE	Better performance in training MLPs, especially in complex datasets.	Sine-cosine mechanics navigate non-linear datasets effectively.
[92]	MVO	Accuracy MSE	Better performance in local optima and convergence.	Multiverse-inspired exploration excels at avoiding local optima and requires high computational resources.
[93]	MVO	MSE	MVO-FNN proved superior in classification tasks.	Effective in classification, computational demand grows.

TABLE VIII
COMPARATIVE ANALYSIS OF PHYSICS-BASED ALGORITHMS FOR TRAINING ANN MODELS

MHA	Advantages	Disadvantages
GSA	Efficiently explores the search space.	The calculation of forces between agents increases computational complexity.
HOS	Can adaptively change search granularity.	May require many function evaluations. Computationally intensive for high dimensions.
LSA	Strong exploration capabilities.	Complexity increases with iterations.
SCA	Fewer parameters. Stable and reliable.	May suffer from slow convergence.
MVO	Good global search capability.	Computationally expensive. May get stuck in local optima.

D. Human-Based Algorithms for ANN Training

1) Teaching Learning-Based Optimization (TLBO): TLBO models the educational dynamics of a classroom to optimize solutions. The algorithm is divided into two main phases: the 'Teacher Phase,' where the most optimal solution guides others, and the 'Learner Phase,' where solutions improve by mutual learning [94]. Ang et al. [95] explored TLBO's application in training FNNs. The approach was tested on classification problems, comparing its efficacy with other advanced TLBO variants. TLBO-ANN showed improved learning efficiency and classification performance.

2) Yin-Yang-pair Optimization (YYPO): YYPO is a lightweight optimization technique inspired by the concept in the Yin-Yang philosophy. This algorithm efficiently solves complex optimization problems by leveraging unidirectional and multi-directional search capabilities [96]. Shekhar et al. [97] introduce the PYYPO algorithm for MLP training. The results showed that the algorithm achieved better classification accuracy on most datasets.

3) Battle Royale Optimization Algorithm (BRO): The BRO algorithm is influenced by battle royale gaming, where agents, represented as players or soldiers, compete with each other to be the last standing by going to safe locations on the virtual battlefields [98]. Agahian and Akan [99] investigated the application of BRO in MLP training and compared its findings with BP and six other optimization algorithms based on ten benchmark datasets. The BRO-MLP method showed superior performance in MSE, accuracy, and convergence.

4) Political Optimizer (PO): PO is a novel optimization algorithm derived from the politics process, which is multi-phased. It mathematically models the phases of politics, such as constituency allocation, party switching, election campaigns, inter-party elections, and parliamentary affairs [100]. Askari and Younas [101] introduce the PO approach for FNN training. The algorithm was tested on five classifications and five function-approximation datasets, presenting equal or even better results than several meta-heuristic competitor algorithms. The study highlights PO's excellent convergence speed and balanced exploration and exploitation behavior.

5) Ali Baba and the Forty Thieves Optimization Algorithm (AFT): AFT is a novel algorithm inspired by the tale of Ali Baba and the Forty Thieves. It focuses on conceptualizing solutions as thieves attempting to maximize their gains while minimizing risks [102]. Al-Hiary et al. [103] We introduce a training method for ANNs based on the AFT algorithm. It was tested on 15 benchmark datasets and showed superior performance in faster convergence compared to other MHAs.

HBAs leverage human-inspired decision-making processes for optimization tasks, making them particularly effective for ANN training. Table IX summarizes various HBAs applied in this context, detailing the algorithm's name, metrics used, significant findings, and the interpretation field. The table provides a comprehensive overview of how HBAs optimize ANN models by mimicking human-inspired strategies like teaching, competition, and resource allocation. Table X provides a comparative overview of these algorithms, summarizing their advantages and disadvantages [82], [104].

TABLE IX
OVERVIEW OF STUDIES UTILIZING HUMAN-BASED ALGORITHMS FOR TRAINING MULTILAYER PERCEPTRON NEURAL NETWORKS

Ref.	MHA	Metrics	Key Findings	Interpretation
[95]	TLBO	Classification Accuracy Rate (CAR) p-value	Improved learning efficiency and classification performance.	The structured Teacher-Learner phases enhance knowledge transfer and accurate convergence.
[97]	YYPO	Accuracy Standard Deviation	Better classification accuracy on most datasets.	The dual search strategy ensures robust exploration and minimizes the risk of local optima.
[99]	BRO	Accuracy MSE	Superior performance in MSE accuracy and convergence.	BRO's competitive mechanisms drive solutions toward optimal regions, improving convergence speed and accuracy.
[101]	PO	MSE Cross-entropy Accuracy	Excellent convergence speed and balanced exploration-exploitation behavior.	Adaptive multi-phased modeling balances global search and local refinement effectively.
[103]	AFT	Accuracy	Faster convergence and superior performance in various datasets.	Strategic resource allocation ensures solution diversity, accelerating convergence.

TABLE X
COMPARATIVE ANALYSIS OF HUMAN-BASED ALGORITHMS FOR TRAINING ANN MODELS

MHA	Advantages	Disadvantages
TLBO	Fewer parameters to tune. Balances global search capability with convergence.	Slower convergence. May require multiple iterations for stability.
YYPO	Dual search mechanism. Adaptability.	Parameter sensitivity. Slower convergence.
BRO	Promotes competition among solutions, enhancing global exploration.	Potential for premature convergence. Computationally expensive due to battles.
PO	Escape local minima. Simple and flexible.	Slow convergence. Parameter sensitivity.
AFT	Avoid local minima in MLP training. Diversity preservation.	Complex implementation. Slow convergence.

E. Hybrid Meta-heuristic Algorithms

Hybrid meta-heuristic algorithms use two or more algorithms to make them work better, enhancing the strengths and mitigating the weaknesses of each algorithm. Tarkhaneh [105] introduced LPSONS, an evolutionary algorithm combining PSO's velocity with Mantegna Lévy distribution for diverse solution generation and preventing premature convergence. Tested on UCI datasets, LPSONS outperformed traditional gradient-based and evolutionary methods.

Agrawal et al. [106] developed a hybrid algorithm, HWBO, combining GWO and Bat Algorithm (BA) for training ANNs. Tested on ten UCI medical datasets, HWBO integrates GWO's exploration abilities with BA's exploitative strengths. Results showed HWBO outperformed other MHAs in most cases, with better accuracy and notably faster convergence speed. HWBO's execution time was also competitive. Zhou et al. [107] introduced an improved TLBO with the WOA algorithm (TSWOA) for training MLPs. The TSWOA enhances exploration through TLBO's teacher phase and incorporates the simplex method to increase search agent diversification, enabling quicker convergence to the global optimum. Tested on 15 UCI datasets, The results show it to be highly effective in optimizing MLPs.

Aleksa et al. [108] proposed a hybrid approach, GGE-ABC, combining GA and ABC algorithms to train MLP. This method merges GA's evolutionary process with ABC's key phases to find optimal solutions. Evaluated on two benchmarks, GGE-ABC outperformed nine other MHAs, demonstrating its robustness and efficiency. Younis et al. [109] propose the (IHGWOSCA) that combines GWO and SCA. The hybrid algorithm leverages GWO's social hierarchy, hunting strategies, and SCA's superior exploration and exploitation capabilities. Yilmaz et al. [110] proposed a hybrid algorithm, MVOSANN, for training ANNs, combining Simulated Annealing (SA) and MVO algorithms. The MVOSANN algorithm was tested on 12 benchmark datasets and compared with 12 well-recognized meta-heuristic algorithms.

F. Analyzing Scientific Maps

In the existing literature, researchers work on systematic reviews and meta-analyses to provide a clear heading toward synthesizing results and locating areas lacking research. However, due to potential bias, bibliometric methods like the R-tool and VOS viewer were employed for the analysis [111]. These procedures provided simple visual representation methods of data, making them convenient. This research uses the bibliometric approach, which is explained in subsequent sections.

1) Country Scientific Production: The scientific production map highlights contributions by authors and nations to MHAs training for ANNs. Fig. 8 presents a color-coded map showing the variation in scientific output, with the darkest blue for the highest and bright blue for the lowest production. Gray areas have minimal output. India and China lead in this field, providing critical insights for researchers and policymakers.

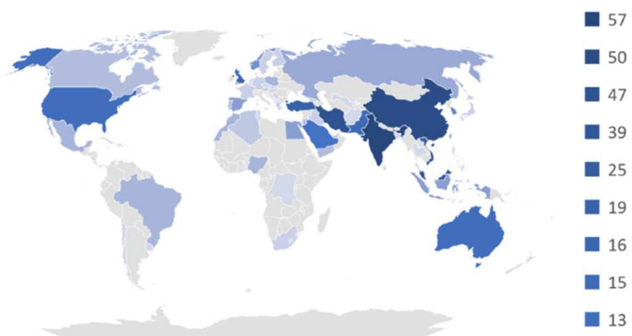

Fig. 8 A color-coded map of literature for MHAs to train ANNs

2) Cloud of Words: The word cloud in Fig. 9 visually represents critical terms from previous research titles, highlighting their prominence.

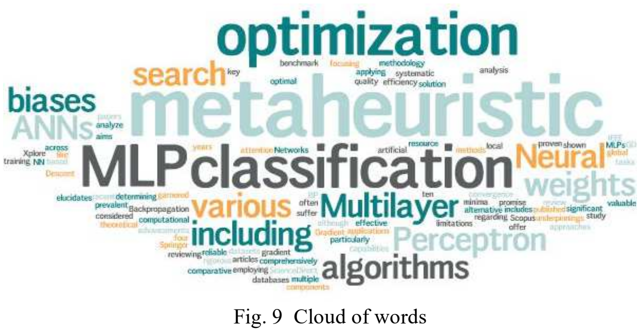

Fig. 9 Cloud of words

The size of each term within the cloud corresponds to its frequency across the studies reviewed, with larger fonts signifying more common occurrences and smaller fonts indicating less frequent mentions. Thus, the cloud effectively encapsulates essential themes in literature.

3) Co-Occurrence: Co-occurrence networks are derived from key terms in the literature illustrated in Fig. 10, revealing the interconnected theoretical foundations of a field. In this network, nodes symbolize topics, with their size reflecting thematic prevalence. This visualization aids researchers in synthesizing existing knowledge, as seen with the prominent mention of 'optimization algorithms' in ANN training, highlighting its critical role in meta-heuristic research.

The findings of this study underscore the critical role of MHAs in advancing ANN training, particularly in addressing the limitations of traditional methods. For example, improved classification accuracy through MHAs can enhance disease diagnosis and patient outcomes by accurately analyzing complex datasets, such as medical imaging or genetic information. Similarly, in finance, the robustness of MHAs can aid in fraud detection, risk management, and market trend prediction by efficiently processing high-dimensional financial data.

One key finding is the superior performance of MHAs compared to traditional gradient-based methods. Approximately 74% of reviewed studies show that MHAs outperform BP regarding test accuracy and training MSE on benchmark datasets. Furthermore, SI algorithms demonstrated better convergence speed and robustness than EAs, HBAs, and PBAs.

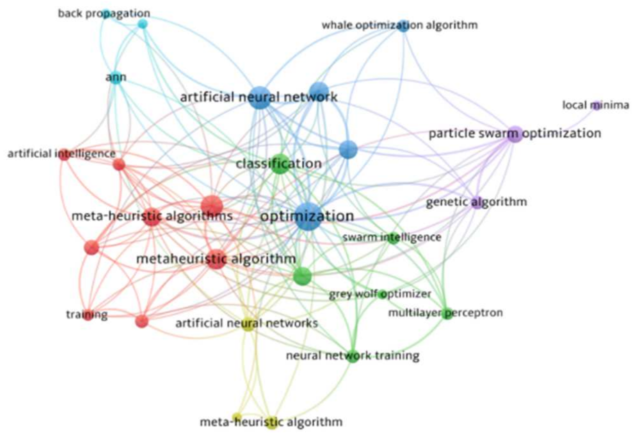

Fig. 10 The co-occurrence network

Another significant finding is the effectiveness of hybrid algorithms, which combine the strengths of different MHAs to achieve superior optimization. By leveraging complementary capabilities, hybrid approaches address common issues like premature convergence and sensitivity to parameter settings, achieving better overall performance.

This study distinguishes itself from previous research by offering a broader analysis of MHAs. Unlike earlier reviews focusing on specific subsets or domains, this work evaluates all significant categories, including hybrid, physics-based, and human-inspired approaches. Using bibliometric tools, such as VOSviewer, adds value by identifying thematic trends and interconnections in MHA research. The emphasis on practical applications and benchmarking diverse datasets also bridges the gap between theoretical advancements and real-world utility.

The study also identifies opportunities for innovation. Most studies rely on traditional metrics like MSE and fixed network architectures, which, while enabling comparability, limit flexibility. Exploring alternative loss functions and dynamic architecture could enhance the adaptability of ANN training. Similarly, reliance on benchmark datasets highlights the need to test MHAs on real-world datasets to understand their practical utility and limitations better.

In this sense, this study provides a comprehensive review of all MHAs used during ANN training. It suggests future research directions for adaptive MHAs, dynamic parameter control, and scaling to deep neural networks. However, these innovations can also increase MHAs' efficiency, scalability, and robustness to optimize neural network performance.

IV. Conclusion

This study highlights the critical advancements and contributions of Meta-heuristic Algorithms (MHAs) in optimizing the training of Artificial Neural Networks (ANNs), particularly for classification tasks. The study comprehensively analyzes their strengths and limitations by systematically reviewing recent developments across evolutionary, swarm intelligence, physics-based, and human-inspired algorithms. The findings emphasize how MHAs address key challenges in ANN training, such as avoiding local optima, improving convergence rates, and enhancing robustness to initial parameter settings.

Through the comparative analysis presented in this work, we identified that hybrid MHAs offer significant advantages by leveraging the complementary strengths of multiple algorithmic paradigms. For instance, their ability to balance exploration and exploitation has improved accuracy and computational efficiency in benchmark datasets. Additionally, the bibliometric analysis provided novel insights into research trends, highlighting areas such as adaptive MHAs and underexplored domains that warrant further investigation.

These findings have profound practical implications. MHAs can enable practitioners to develop accurate and efficient models for critical applications, such as early disease detection, personalized treatment planning, and healthcare resource optimization. In finance, MHAs can support the development of predictive models for credit scoring, fraud detection, and algorithmic trading.

This study bridges the gap between theoretical advancements and their real-world implementation by comprehensively evaluating MHAs and emphasizing their practical applications. The insights and recommendations establish a solid foundation for future exploration, ensuring that MHAs continue to drive innovation in ANN training and optimization.

Future research in this field should prioritize developing adaptive MHAs that dynamically adjust parameters during training, creating hybrid algorithms combining strengths of different MHAs, tailoring algorithms to specific domains, and extending their application to training Deep Neural Networks (DNNs) for high-dimensional optimization tasks. Additionally, exploring dynamic parameter control and testing MHAs on complex real-world datasets can enhance performance and practical utility.

Acknowledgment

The authors thank the Faculty of Computer Sciences and Information Technology / Universiti Putra Malaysia, Northern Technical University (NTU), and Technical Institute / Mosul (MTI) for their tremendous support in completing this research.

References

[1] S. S. Haykin, *Neural networks: A comprehensive foundation*, 6th ed. Upper Saddle River, N.J: Prentice Hall., 1999.

[2] F. Murtagh, "Multilayer perceptrons for classification and regression," *Neurocomputing*, vol. 2, no. 5, pp. 183–197, 1991, doi:10.1016/0925-2312(91)90023-5.

[3] E. Soria, J. D. Martín, and P. J. G. Lisboa, "Classical Training Methods BT - Metaheuristic Procedures for Training Neutral Networks," E. Alba and R. Martí, Eds., Boston, MA: Springer US, 2006, pp. 7–36. doi: 10.1007/0-387-33416-5_1.

[4] S. Mirjalili, "How effective is the Grey Wolf optimizer in training multi-layer perceptrons," *Applied Intelligence*, vol. 43, no. 1, pp. 150–161, Jul. 2015, doi: 10.1007/s10489-014-0645-7.

[5] M. Mohammed Rashid, O. M. Yaseen, R. Riyadh Saeed, and M. T. Alasaady, "An Improved Ensemble Machine Learning Approach for Diabetes Diagnosis," *Pertanika J Sci Technol*, vol. 32, no. 3, pp. 1335–1350, Apr. 2024, doi: 10.47836/pjst.32.3.19.

[6] M. T. Alasaady, T. N. M. Aris, N. M. Sharef, and H. Hamdan, "A proposed approach for diabetes diagnosis using neuro-fuzzy technique," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 6, pp. 3590–3597, 2022.

[7] S. Mirjalili, "How effective is the Grey Wolf optimizer in training multi-layer perceptrons," *Applied Intelligence*, vol. 43, no. 1, pp. 150–161, 2015.

[8] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[9] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput*, vol. 18, no. 7, pp. 1527–1554, 2006.

[10] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans Neural Netw*, vol. 5, no. 2, pp. 157–166, 1994, doi: 10.1109/72.279181.

[11] H. T. Rauf, W. H. Bangyal, J. Ahmad, and S. A. Bangyal, "Training of Artificial Neural Network Using PSO With Novel Initialization Technique," in *2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, IEEE, Nov. 2018, pp. 1–8. doi: 10.1109/3ICT.2018.8855743.

[12] T. Si and R. Dutta, "Partial Opposition-Based Particle Swarm Optimizer in Artificial Neural Network Training for Medical Data Classification," in *International Journal of Information Technology and Decision Making*, World Scientific Publishing Co. Pte Ltd, Sep. 2019, pp. 1717–1750. doi: 10.1142/S0219622019500329.

[13] M. G. Rojas, A. C. Olivera, and P. J. Vidal, "Optimising Multilayer Perceptron weights and biases through a Cellular Genetic Algorithm for medical data classification," *Array*, vol. 14, no. January, p. 100173, 2022, doi: 10.1016/j.array.2022.100173.

[14] H. Faris, S. Mirjalili, and I. Aljarah, "Automatic selection of hidden neurons and weights in neural networks using grey wolf optimizer based on a hybrid encoding scheme," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 10, pp. 2901–2920, Oct. 2019, doi: 10.1007/s13042-018-00913-2.

[15] V. K. Ojha, A. Abraham, and V. Snášel, "Metaheuristic design of feedforward neural networks: A review of two decades of research," *Eng Appl Artif Intell*, vol. 60, pp. 97–116, 2017.

[16] J. Bagchi and T. Si, "Artificial Neural Network Training Using Marine Predators Algorithm for Medical Data Classification," 2022, pp. 137–148. doi: 10.1007/978-981-16-3802-2_11.

[17] S. J. Mousavirad, G. Schaefer, S. M. J. Jalali, and I. Korovin, "A Benchmark of Recent Population-Based Metaheuristic Algorithms for Multi-Layer Neural Network Training," in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, in

GECCO '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1402–1408. doi: 10.1145/3377929.3398144.
[18] B. A. S. Emambocus, M. B. Jasser, and A. Amphawan, "A Survey on the Optimization of Artificial Neural Networks Using Swarm Intelligence Algorithms," *IEEE Access*, vol. 11, pp. 1280–1294, 2023, doi: 10.1109/ACCESS.2022.3233596.
[19] D. Devikanniga, K. Vetrivel, and N. Badrinath, "Review of meta-heuristic optimization based artificial neural networks and its applications," in *Journal of Physics: Conference Series*, IOP Publishing, 2019, p. 012074.
[20] Z. Munn, M. D. J. Peters, C. Stern, C. Tufanaru, A. McArthur, and E. Aromataris, "Systematic review or scoping review? Guidance for authors when choosing between a systematic or scoping review approach," *BMC Med Res Methodol*, vol. 18, no. 1, p. 143, Dec. 2018, doi: 10.1186/s12874-018-0611-x.
[21] M. J. Page *et al.*, "PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews," *bmj*, vol. 372, 2021.
[22] M. A. Awadallah, I. Abu-Doush, M. A. Al-Betar, and M. S. Braik, "Metaheuristics for optimizing weights in neural networks," in *Comprehensive Metaheuristics: Algorithms and Applications*, Elsevier, 2023, pp. 359–377. doi: 10.1016/B978-0-323-91781-0.00005-3.
[23] D. Whitley, "A genetic algorithm tutorial," *Stat Comput*, vol. 4, no. 2, pp. 65–85, 1994, doi: 10.1007/BF00175354.
[24] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015. doi:10.1007/978-3-662-44874-8.
[25] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm Intelligence Algorithms for Feature Selection: A Review," *Applied Sciences*, vol. 8, no. 9, p. 1521, Sep. 2018, doi: 10.3390/app8091521.
[26] R. Salgotra, P. Sharma, S. Raju, and A. H. gandomi, "A Contemporary Systematic Review on Meta-heuristic Optimization Algorithms with Their MATLAB and Python Code Reference," *Archives of Computational Methods in Engineering*, vol. 31, no. 3, pp. 1749–1822, Apr. 2024, doi: 10.1007/s11831-023-10030-1.
[27] A. Askarzadeh and A. Rezazadeh, "Artificial neural network training using a new efficient optimization algorithm," *Appl Soft Comput*, vol. 13, no. 2, pp. 1206–1213, Feb. 2013, doi: 10.1016/j.asoc.2012.10.023.
[28] P. Bansal *et al.*, "GGA-MLP: A Greedy Genetic Algorithm to Optimize Weights and Biases in Multilayer Perceptron," *Contrast Media Mol Imaging*, vol. 2022, 2022, doi: 10.1155/2022/4036035.
[29] D. Simon, "Biogeography-Based Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702–713, 2008, doi:10.1109/TEVC.2008.919004.
[30] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Let a biogeography-based optimizer train your Multi-Layer Perceptron," *Inf Sci (N Y)*, vol. 269, pp. 188–209, Jun. 2014, doi: 10.1016/j.ins.2014.01.038.
[31] X.-D. Li, J.-S. Wang, W.-K. Hao, M. Wang, and M. Zhang, "Multi-layer perceptron classification method of medical data based on biogeography-based optimization algorithm with probability distributions," *Appl Soft Comput*, vol. 121, p. 108766, May 2022, doi:10.1016/j.asoc.2022.108766.
[32] S. A. Uymaz, G. Tezel, and E. Yel, "Artificial algae algorithm (AAA) for nonlinear global optimization," *Appl Soft Comput*, vol. 31, pp. 153–171, 2015, doi:10.1016/j.asoc.2015.03.003.
[33] B. Turkoglu and E. Kaya, "Training multi-layer perceptron with artificial algae algorithm," *Engineering Science and Technology, an International Journal*, vol. 23, no. 6, pp. 1342–1350, Dec. 2020, doi:10.1016/j.jestch.2020.07.001.
[34] M. Karakoyun, "Artificial neural network training using a multi selection artificial algae algorithm," *Engineering Science and Technology, an International Journal*, vol. 53, p. 101684, May 2024, doi: 10.1016/j.jestch.2024.101684.
[35] D. E. Golberg, "Genetic algorithms in search, optimization, and machine learning," *Addion wesley*, vol. 1989, no. 102, p. 36, 1989.
[36] O. W. Khalid, N. A. M. Isa, and H. A. Mat Sakim, "Emperor penguin optimizer: A comprehensive review based on state-of-the-art meta-heuristic algorithms," Feb. 01, 2023, *Elsevier B.V.* doi:10.1016/j.aej.2022.08.013.
[37] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, IEEE, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.
[38] X. Li, J. Zhang, and M. Yin, "Animal migration optimization: an optimization algorithm inspired by animal migration behavior," *Neural Comput Appl*, vol. 24, no. 7–8, pp. 1867–1877, Jun. 2014, doi:10.1007/s00521-013-1433-8.
[39] Ş. Gülcü, "An Improved Animal Migration Optimization Algorithm to Train the Feed-Forward Artificial Neural Networks," *Arab J Sci Eng*, vol. 47, no. 8, pp. 9557–9581, Aug. 2022, doi: 10.1007/s13369-021-06286-z.
[40] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014, doi:10.1016/j.advengsoft.2013.12.007.
[41] O. Altay and E. Varol Altay, "A novel hybrid multilayer perceptron neural network with improved grey wolf optimizer," *Neural Comput Appl*, vol. 35, no. 1, pp. 529–556, Jan. 2023, doi: 10.1007/s00521-022-07775-4.
[42] M. S. Kiran, "TSA: Tree-seed algorithm for continuous optimization," *Expert Syst Appl*, vol. 42, no. 19, pp. 6686–6698, 2015, doi:10.1016/j.eswa.2015.04.055.
[43] A. C. Cinar, "Training Feed-Forward Multi-Layer Perceptron Artificial Neural Networks with a Tree-Seed Algorithm," *Arab J Sci Eng*, vol. 45, no. 12, pp. 10915–10938, Dec. 2020, doi:10.1007/s13369-020-04872-1.
[44] S. Mirjalili, "The Ant Lion Optimizer," *Advances in Engineering Software*, vol. 83, pp. 80–98, 2015, doi:10.1016/j.advengsoft.2015.01.010.
[45] W. Yamany, A. Tharwat, M. F. Hassanin, T. Gaber, A. E. Hassanien, and T.-H. Kim, "A New Multi-layer Perceptrons Trainer Based on Ant Lion Optimization Algorithm," in *2015 Fourth International Conference on Information Science and Industrial Applications (ISI)*, 2015, pp. 40–45. doi: 10.1109/ISI.2015.9.
[46] H. and M. S. and A. I. and M. M. Heidari Ali Asghar and Faris, "Ant Lion Optimizer: Theory, Literature Review, and Application in Multi-layer Perceptron Neural Networks," in *Nature-Inspired Optimizers: Theories, Literature Reviews and Applications*, J. and L. A. Mirjalili Seyedali and Song Dong, Ed., Cham: Springer International Publishing, 2020, pp. 23–46. doi: 10.1007/978-3-030-12127-3_3.
[47] W. Yamany, M. Fawzy, A. Tharwat, and A. E. Hassanien, "Moth-flame optimization for training Multi-Layer Perceptrons," in *2015 11th International Computer Engineering Conference (ICENCO)*, 2015, pp. 267–272. doi: 10.1109/ICENCO.2015.7416360.
[48] Z. Yang, "FMFO: Floating flame moth-flame optimization algorithm for training multi-layer perceptron classifier," *Applied Intelligence*, vol. 53, no. 1, pp. 251–271, Jan. 2023, doi: 10.1007/s10489-022-03484-6.
[49] G.-G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Comput Appl*, vol. 31, no. 7, pp. 1995–2014, Jul. 2019, doi:10.1007/s00521-015-1923-y.
[50] H. Faris, I. Aljarah, and S. Mirjalili, "Improved monarch butterfly optimization for unconstrained global search and neural network training," *Applied Intelligence*, vol. 48, no. 2, pp. 445–464, Feb. 2018, doi: 10.1007/s10489-017-0967-3.
[51] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," *Comput Struct*, vol. 169, pp. 1–12, 2016, doi:10.1016/j.compstruc.2016.03.001.
[52] M. Khamees, W. S. Ahmed, and S. Q. Abbas, "Train the Multi-Layer Perceptrons Based on Crow Search Algorithm," in *2020 1st. Information Technology To Enhance e-learning and Other Application (IT-ELA*, 2020, pp. 160–164. doi: 10.1109/IT-ELA50150.2020.9253073.
[53] F. Erdoğan and Ş. Gülcü, "Training of the Artificial Neural Networks using Crow Search Algorithm," *Original Research Paper International Journal of Intelligent Systems and Applications in Engineering IJISAE*, vol. 2021, no. 3, pp. 101–108, 2021, doi:10.1039/b000000x.
[54] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016, doi:10.1016/j.advengsoft.2016.01.008.
[55] R. Bhesdadiya, P. Jangir, N. Jangir, I. N. Trivedi, and D. Ladumor, "Training multi-layer perceptron in neural network using whale optimization algorithm," *Indian J Sci Technol*, vol. 9, no. 19, pp. 28–36, 2016.
[56] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft comput*, vol. 22, no. 1, pp. 1–15, Jan. 2018, doi: 10.1007/s00500-016-2442-1.
[57] R. Kushwah, M. Kaushik, and K. Chugh, "A modified whale optimization algorithm to overcome delayed convergence in artificial neural networks," *Soft comput*, vol. 25, no. 15, pp. 10275–10286, Aug. 2021, doi: 10.1007/s00500-021-05983-z.

[58] S. Raziani, S. Ahmadian, S. M. J. Jalali, and A. Chalechale, "An Efficient Hybrid Model Based on Modified Whale Optimization Algorithm and Multilayer Perceptron Neural Network for Medical Classification Problems," *J Bionic Eng*, vol. 19, no. 5, pp. 1504–1521, Sep. 2022, doi: 10.1007/s42235-022-00216-x.

[59] R. Chatterjee, R. Mukherjee, P. K. Roy, and D. K. Pradhan, "Chaotic oppositional-based whale optimization to train a feed forward neural network," *Soft comput*, vol. 26, no. 22, pp. 12421–12443, Nov. 2022, doi: 10.1007/s00500-022-07141-5.

[60] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput Appl*, vol. 27, no. 4, pp. 1053–1073, May 2016, doi: 10.1007/s00521-015-1920-1.

[61] M. E. Abo-Elsoud, M. Morsy, and S. Elnazer, "A New Multi-layer Perceptron Trainer based on Dragonfly Optimization Algorithm.," *Bulletin of the Faculty of Engineering. Mansoura University*, vol. 41, no. 4, pp. 11–19, Dec. 2016, doi: 10.21608/bfemu.2020.103962.

[62] Ş. Gülcü, "Training of the feed forward artificial neural networks using dragonfly algorithm," *Appl Soft Comput*, vol. 124, p. 109023, 2022, doi:10.1016/j.asoc.2022.109023.

[63] G. Dhiman and V. Kumar, "Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications," *Advances in Engineering Software*, vol. 114, pp. 48–70, 2017, doi:10.1016/j.advengsoft.2017.05.014.

[64] N. Panda and S. K. Majhi, "How effective is spotted hyena optimizer for training multilayer perceptrons," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2, pp. 4915–4927, Jul. 2019, doi: 10.35940/ijrte.B3736.078219.

[65] Q. Luo, J. Li, Y. Zhou, and L. Liao, "Using spotted hyena optimizer for training feedforward neural networks," *Cogn Syst Res*, vol. 65, pp. 1–16, 2021, doi:10.1016/j.cogsys.2020.09.001.

[66] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper Optimisation Algorithm: Theory and application," *Advances in Engineering Software*, vol. 105, pp. 30–47, 2017, doi:10.1016/j.advengsoft.2017.01.004.

[67] A. A. Heidari, H. Faris, I. Aljarah, and S. Mirjalili, "An efficient hybrid multilayer perceptron neural network with grasshopper optimization," *Soft comput*, vol. 23, no. 17, pp. 7941–7958, Sep. 2019, doi:10.1007/s00500-018-3424-2.

[68] S. Arora and S. Singh, "Butterfly optimization algorithm: a novel approach for global optimization," *Soft comput*, vol. 23, pp. 715–734, 2019.

[69] Jalali Seyed Mohammad, Sajad Ahmadian, Parham M. Kebria, Abbas Khosravi, Chee Peng Lim, and Saeid Nahavandi, "Evolving Artificial Neural Networks Using Butterfly Optimization Algorithm for Data Classification," in *Neural Information Processing*, K. W. and L. M. Gedeon Tom and Wong, Ed., Cham: Springer International Publishing, 2019, pp. 596–607.

[70] B. Irmak and Ş. Gülcü, "Training of the feed-forward artificial neural networks using butterfly optimization algorithm," *MANAS Journal of Engineering*, vol. 9, no. 2, pp. 160–168, Dec. 2021, doi:10.51354/mjen.917837.

[71] B. Irmak, M. Karakoyun, and Ş. Gülcü, "An improved butterfly optimization algorithm for training the feed-forward artificial neural networks," *Soft comput*, vol. 27, no. 7, pp. 3887–3905, Apr. 2023, doi:10.1007/s00500-022-07592-w.

[72] G. Dhiman and V. Kumar, "Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems," *Knowl Based Syst*, vol. 165, pp. 169–196, Feb. 2019, doi:10.1016/j.knosys.2018.11.024.

[73] N. Bacanin, M. Antonijevic, N. Vukobrat, T. Bezdan, and M. Zivkovic, "Enhanced Seagull Optimization Algorithm Adapted for Artificial Neural Network Training," 2022, pp. 753–761. doi:10.1007/978-981-16-4177-0_74.

[74] K. Zervoudakis and S. Tsafarakis, "A mayfly optimization algorithm," *Comput Ind Eng*, vol. 145, p. 106559, Jul. 2020, doi:10.1016/j.cie.2020.106559.

[75] S. K. R. Moosavi, M. H. Zafar, M. N. Akhter, S. F. Hadi, N. M. Khan, and F. Sanfilippo, "A Novel Artificial Neural Network (ANN) Using The Mayfly Algorithm for Classification," in *2021 International Conference on Digital Futures and Transformative Technologies (ICoDT2)*, IEEE, May 2021, pp. 1–6. doi:10.1109/ICoDT252288.2021.9441473.

[76] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine Predators Algorithm: A nature-inspired metaheuristic," *Expert Syst Appl*, vol. 152, p. 113377, 2020, doi:10.1016/j.eswa.2020.113377.

[77] J. Zhang and Y. Xu, "Training Feedforward Neural Networks Using an Enhanced Marine Predators Algorithm," *Processes*, vol. 11, no. 3, 2023, doi: 10.3390/pr11030924.

[78] I. Naruei and F. Keynia, "A new optimization method based on COOT bird natural life model," *Expert Syst Appl*, vol. 183, p. 115352, 2021, doi:10.1016/j.eswa.2021.115352.

[79] A. Özden and İ. İşeri, "COOT optimization algorithm on training artificial neural networks," *Knowl Inf Syst*, vol. 65, no. 8, pp. 3353–3383, Aug. 2023, doi: 10.1007/s10115-023-01859-w.

[80] B. A. S. Emambocus, M. B. Jasser, and A. Amphawan, "A Survey on the Optimization of Artificial Neural Networks Using Swarm Intelligence Algorithms," *IEEE Access*, vol. 11, pp. 1280–1294, 2023, doi: 10.1109/access.2022.3233596.

[81] Z. Ma, G. Wu, P. N. Suganthan, A. Song, and Q. Luo, "Performance assessment and exhaustive listing of 500+ nature-inspired metaheuristic algorithms," *Swarm Evol Comput*, vol. 77, p. 101248, Mar. 2023, doi: 10.1016/j.swevo.2023.101248.

[82] M. G. M. Abdolrasol *et al.*, "Artificial Neural Networks Based Optimization Techniques: A Review," *Electronics (Basel)*, vol. 10, no. 21, 2021, doi: 10.3390/electronics10212689.

[83] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Inf Sci (N Y)*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009, doi: 10.1016/j.ins.2009.03.004.

[84] S. A. Rather and P. S. Bala, "Lévy flight and chaos theory based gravitational search algorithm for multilayer perceptron training," *Evolving Systems*, vol. 14, no. 3, pp. 365–392, Jun. 2023, doi:10.1007/s12530-022-09456-y.

[85] R. H. Abiyev and M. Tunay, "Optimization of High-Dimensional Functions through Hypercube Evaluation," *Comput Intell Neurosci*, vol. 2015, p. 967320, 2015, doi: 10.1155/2015/967320.

[86] M. Tunay, E. Pashaei, and E. Pashaei, "Hybrid Hypercube Optimization Search Algorithm and Multilayer Perceptron Neural Network for Medical Data Classification," *Comput Intell Neurosci*, vol. 2022, p. 1612468, 2022, doi: 10.1155/2022/1612468.

[87] H. Shareef, A. A. Ibrahim, and A. H. Mutlag, "Lightning search algorithm," *Appl Soft Comput*, vol. 36, pp. 315–333, 2015, doi:10.1016/j.asoc.2015.07.028.

[88] H. Faris, I. Aljarah, N. Al-Madi, and S. Mirjalili, "Optimizing the Learning Process of Feedforward Neural Networks Using Lightning Search Algorithm," *International Journal on Artificial Intelligence Tools*, vol. 25, no. 06, p. 1650033, Dec. 2016, doi:10.1142/S0218213016500330.

[89] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," *Knowl Based Syst*, vol. 96, pp. 120–133, 2016, doi:10.1016/j.knosys.2015.12.022.

[90] S. Gupta and K. Deep, "A novel hybrid sine cosine algorithm for global optimization and its application to train multilayer perceptrons," *Applied Intelligence*, vol. 50, no. 4, pp. 993–1026, Apr. 2020, doi: 10.1007/s10489-019-01570-w.

[91] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization," *Neural Comput Appl*, vol. 27, no. 2, pp. 495–513, Feb. 2016, doi: 10.1007/s00521-015-1870-7.

[92] H. Faris, I. Aljarah, and S. Mirjalili, "Training feedforward neural networks using multi-verse optimizer for binary classification problems," *Applied Intelligence*, vol. 45, no. 2, pp. 322–332, Sep. 2016, doi: 10.1007/s10489-016-0767-1.

[93] M. F. Hassanin, A. M. Shoeb, and A. E. Hassanien, "Designing Multilayer Feedforward Neural Networks Using Multi-Verse Optimizer," 2017, pp. 1076–1093. doi: 10.4018/978-1-5225-2229-4.ch048.

[94] R V Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, 2011, doi:10.1016/j.cad.2010.12.015.

[95] K. M. Ang, W. H. Lim, S. S. Tiang, C. K. Ang, E. Natarajan, and M. K. A. Ahamed Khan, "Optimal Training of Feedforward Neural Networks Using Teaching-Learning-Based Optimization with Modified Learning Phases," *Proceedings of the 12th National Technical Seminar on Unmanned System Technology 2020*, pp. 867–887, Sep. 2021, doi: 10.1007/978-981-16-2406-3_65.

[96] V. Punnathanam and P. Kotecha, "Yin-Yang-pair Optimization: A novel lightweight optimization algorithm," *Eng Appl Artif Intell*, vol. 54, pp. 62–79, Sep. 2016, doi: 10.1016/j.engappai.2016.04.004.

[97] M. Shekhar, "Training Multi-Layer Perceptron Using Population-Based Yin-Yang-Pair Optimization," 2021, pp. 417–425. doi:10.1007/978-981-15-4992-2_39.

[98] T. Akan, S. Agahian, and R. Dehkharghani, “Battle Royale Optimizer for solving binary optimization problems,” *Software Impacts*, vol. 12, p. 100274, 2022.

[99] S. Agahian and T. Akan, “Battle royale optimizer for training multi-layer perceptron,” *Evolving Systems*, vol. 13, no. 4, pp. 563–575, Aug. 2022, doi: 10.1007/s12530-021-09401-5.

[100] Q. Askari, I. Younas, and M. Saeed, “Political Optimizer: A novel socio-inspired meta-heuristic for global optimization,” *Knowl Based Syst*, vol. 195, p. 105709, May 2020, doi:10.1016/j.knosys.2020.105709.

[101] Q. Askari and I. Younas, “Political Optimizer Based Feedforward Neural Network for Classification and Function Approximation,” *Neural Process Lett*, vol. 53, no. 1, pp. 429–458, Feb. 2021, doi:10.1007/s11063-020-10406-5.

[102] M. Braik, M. H. Ryalat, and H. Al-Zoubi, “A novel meta-heuristic algorithm for solving numerical optimization problems: Ali Baba and the forty thieves,” *Neural Comput Appl*, vol. 34, no. 1, pp. 409–455, Jan. 2022, doi: 10.1007/s00521-021-06392-x.

[103] H. Al-Hiary, M. Braik, and M. A. Al-Betar, “Optimizing Connection Weights in Neural Networks Using Ali Baba and the Forty Thieves Algorithm,” in *2023 24th International Arab Conference on Information Technology (ACIT)*, 2023, pp. 1–7. doi:10.1109/ACIT58888.2023.10453886.

[104] Z. Beheshti and S. M. H. Shamsuddin, “A review of population-based meta-heuristic algorithms,” *Int. j. adv. soft comput. appl*, vol. 5, no. 1, pp. 1–35, 2013.

[105] O. Tarkhaneh and H. Shen, “Training of feedforward neural networks for data classification using hybrid particle swarm optimization, Mantegna Lévy flight and neighborhood search,” *Heliyon*, vol. 5, no. 4, p. e01275, Apr. 2019, doi: 10.1016/j.heliyon.2019.e01275.

[106] U. Agrawal, J. Arora, R. Singh, D. Gupta, A. Khanna, and A. Khamparia, “Hybrid wolf-bat algorithm for optimization of connection weights in multi-layer perceptron,” *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 16, no. 1s, Apr. 2020, doi: 10.1145/3350532.

[107] Y. Zhou, Y. Niu, Q. Luo, and M. Jiang, “Teaching learning-based whale optimization algorithm for multi-layer perceptron neural network training,” *Mathematical Biosciences and Engineering*, vol. 17, no. 5, pp. 5987–6025, 2020, doi: 10.3934/mbe.2020319.

[108] Aleksa Cuk *et al.*, “Feedforward Multi-Layer Perceptron Training by Hybridized Method between Genetic Algorithm and Artificial Bee Colony,” in *Data Science and Data Analytics*, 1st ed., Chapman and Hall/CRC, 2021, ch. 17, p. 14.

[109] H. Bin Younis, M. H. Zafar, N. M. Khan, P. R. Ali, U. A. Khan, and S. K. R. Moosavi, “Improved Hybrid Grey Wolf Optimizer Sine Cosine Algorithm (IHGWOSCA) Trained Artificial Neural Network (ANN) for Classification,” in *2021 16th International Conference on Emerging Technologies (ICET)*, IEEE, Dec. 2021, pp. 1–6. doi:10.1109/ICET54505.2021.9689822.

[110] Ö. Yılmaz, A. A. Altun, and M. Köklü, “Optimizing the learning process of multi-layer perceptrons using a hybrid algorithm based on MVO and SA,” *International Journal of Industrial Engineering Computations*, vol. 13, no. 4, pp. 617–640, 2022, doi:10.5267/j.ijiec.2022.5.003.

[111] M. Aria and C. Cuccurullo, “bibliometrix: An R-tool for comprehensive science mapping analysis,” *J Informetr*, vol. 11, no. 4, pp. 959–975, 2017, doi:10.1016/j.joi.2017.08.007.