Intrusion Detection by XGBoost Model Tuned by Improved Multi-verse Optimizer

Aleksandar Petrovic, Milos Antonijevic, Ivana Strumberger(✉), Nebojsa Budimirovic, Nikola Savanovic, and Stefana Janicijevic

Singidunum University, Danijelova 32, 11000 Belgrade, Serbia
{aleksandar.petrovic,mantonijevic,istrumberger,nsavanovic, sjanicijevic}@singidunum.ac.rs

Abstract. Artificial intelligence and internet of things (IoT) fields have contributed to the flourishment of the industry 4.0 concept. The main benefits include the improvements in terms of device communication, productivity, and efficiency. Nevertheless, there is a downside concerning the security of these systems. The amount of devices and their diversity prove a security risk. Due to this intrusion detection systems are paramount. This paper proposes a novel framework exploiting extreme gradient boosting machine learning model which is optimized by a modified version of the multi-verse optimizer metaheuristic. The UNSW-NB intrusion dataset was used for experimental purposes on which the other cutting-edge techniques were tested and compared. The results provide the proof of improvement as the proposed method outperformed all other overall metaheuristic performances. Furthermore, the units for truthfulness and polarity for the case have been established as a standard evaluation system. True and false positives exist alongside the same negative counterparts. The results provided by these metrics have been visualized and used for further comparison proving the superiority of the performance of the proposed solution.

Keywords: intrusion detection · swarm intelligence · XGBoost · optimization · multi-verse optimizer algorithm

1 Introduction

Industrial Revolution 4.0 has emerged due to advancements in the field of Internet of Things (IoT). This influences every aspect of the industrial process which includes factories. Benefits such as a larger network of devices capable of communication results in a new concept called smart factories. The list of improvements does not stop at communication of devices. Rises in productivity and quality are expected with implementations of such concepts. Furthermore, the resulting product can be more customizable towards the client's requests.

N. Bacanin and H. Shaker (Eds.): ICIITB 2022, ACSR 104, pp. 203–218, 2023.
https://doi.org/10.2991/978-94-6463-110-4_15

Although, strong enhancements as such do not come without disadvantages. With the increase of devices on the network risk of intrusion also increases. Multiple factors have an influence on security. Difficulties for maintaining satisfactory security levels include lack of standardization, device compatibility, substantial financial resources, lack of staff expertise, and proper response and mitigation to failure as well to intrusion. Additionally the number of devices especially influences the last two mentioned hindrances. Real-time intrusion detection systems are the most common countermeasure of choice for these cases [51].

The improvements in the security filed have benefited from advancements in the fields of artificial intelligence (AI) [28] and machine Learning (ML) [29]. Furthermore, recent sophisticated systems exploit the developments in hardware terms, as more powerful hardware becomes accessible. The improvements of AI and ML solutions contribute as well to the security field. Success follows such algorithms especially with the optimization of NP-hard problems.

The downside to these types of solutions is that they have high cost in terms of computational resources as they commonly work with enormous datasets. In terms of the complexity of the algorithms themselves, important control parameters that can be abundant only increase the complexity. This problem is optimizable and it is referred to as hyper-parameter optimization. Tuning hyper-parameters is a process that requires manual adjustments as it is considered NP-hard. The mentioned NP-hard optimizers shine with this sort of problem. Considering the no free lunch theorem [58], every problem requires a tailored solution as no single solution solves all problems equally well.

Consequentially, new solutions are constantly explored while the existing are improved. A distinguished algorithm in terms of performance is eXtreme Gradient Boosting (XGBoost) [27]. Ensemble learning methods have proven to give higher performance stability in comparison to traditional models. The random forest optimizer is exceptional for the problems of classification. XGBoost is the improved version with the introduction of regularization terms as well as second-order derivatives. These solutions have been applied for various problems like business [42], finance [30], the industry [35], and last but not least security [72].

However, XGBoost has flaws. The optimization of the XGBoost is complex due to large amount of control parameters. Swarm intelligence (SI) algorithms [18] have yielded more than satisfactory results in terms of optimization considering the complexity of the problem [1]. Stochastic and population-based nature of these algorithms makes them prone to symbiosis with ML techniques.

The key contributions of this work:

- Development of a modified multi-verse optimizer algorithm (MVO), with changes towards the addression of known deficiencies.
- Configuration of the manufactured algorithm for the framework that optimizes the XGBoost parameters.
- Assessment on the UNSW-NB 15, popular intrusion detection benchmark dataset.

– Evaluation with different cutting-edge techniques designed to tackle the same issue.

The organization of the rest of the paper:

Section 2 gives information on this research's required technologies, Sect. 3 describes and shows the original version of the modified algorithm alongside the performed modifications, Sect. 4 provides the experimental setup followed by the results of the research, lastly Sect. 5 has the goal of summarizing this work with final comments.

2 Preliminaries and Related Works

2.1 Overview of XGBoost

As mentioned regularization terms and second-order derivatives are exploited with the XGBoost. The reason for such modifications to the random forest optimizer are justified in terms of performance which as a result is higher than with similar algorithms used for complex problems. Furthermore, objective function is improved for better optimization characteristics by additive training method. Introduction of such mechanism allows for learning based on past outcome, taking into account the previous step. The Eq. '1 describes the XGBoost model t-th objective function:

$$F_o{}^i = \sum_{k=1}^{n} l\left(y_k, \hat{y}_k^{i-1} + f_i\left(x_k\right)\right) + R(f_i) + const \tag{1}$$

for l term loss in the t-th round, $const$ constants term, and R regularization term. The R is calculated by the Eq. (2).

$$R(f_i) = \gamma T_i + \frac{\lambda}{2}\sum_{j=1}^{T} w_j^2 \tag{2}$$

in which the customization parameters are γ and λ, and it is important to emphasize that the tree structure is simpler for larger values of these parameters.

Second-order Taylor expansion is used for overfitting mitigation as in Eq. (1):

$$obj^{(t)} = \sum_{i=1}^{n}[l(y_i, \hat{y}_i^{t-1}) + g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)] + \Omega(f_t) + const \tag{3}$$

The Eq. 4 calculates the first derivative g, and the Eq. 5 calculates the second derivative h.

$$g_j = \partial_{\hat{y}_k^{i-1}} l\left(y_j, \hat{y}_k^{i-1}\right) \tag{4}$$

$$h_j = \partial^2_{\hat{y}_k^{i-1}} l\left(y_j, \hat{y}_k^{i-1}\right) \tag{5}$$

The combined equations Eq. 2, Eq. 4 and Eq. 5 into Eq. 3 form the following equations:

$$w_j^* = -\frac{\sum g_t}{\sum h_t + \lambda} \tag{6}$$

$$F_o{}^* = -\frac{1}{2}\sum_{j=1}^{T}\frac{(\sum g)^2}{\sum h + \lambda} + \gamma T \tag{7}$$

for which the score of the loss function is $F_o{}^*$, and the weights of solutions are given as w_j^*.

2.2 Swarm Intelligence Applications in Machine Learning

Nature inspired algorithms are considered metaheuristic to which group swarm intelligence belongs to. Swarming behaviors observed in nature translate well to computer algorithms which alongside SI stochastic and population characteristics makes them excellent optimizers for NP-hard problems. Even so, the space for improvement still exists and the best results are obtained by employing hybridization. The goal of this process is to eliminate the shortcomings of combined algorithms. The unimproved versions of the optimizing algorithms mostly have problems with the balance between the two main phases, exploration and exploitation. The first phase focuses on expanding the area of search, and the second phase localizes the best solution. The end goal is the global optima, for which trade-off has to be considered between these two phases.

Widely applied and optimized algorithms are ABC [34], firefly algorithm (FA) [60], elephant herding optimization (EHO) [56], bat algorithm (BA) [61], particle swarm optimization (PSO) [46], and whale optimization algorithm (WOA) [45].

More recent powerful algorithms are the MVO [40], salp swarm algorithm (SSA) [38], monarch butterfly optimization (MBO) [57], and grasshopper optimization algorithm (GOA) [41].

NP-hard complexity with real world problems is common and hence the application of these algorithms is diverse. Some notable examples are artificial neural network optimization [7–10,12,14,15,19,21,26,32,36,48,53,54], wireless sensors networks (WSNs) [4,11,13,52,65,75], cryptocurrency trends estimations [44,49], finally the COVID-19 global epidemic-associated applications [22,25,64,66,69–71,73], computer-conducted MRI classification and sickness determination [17,20,24,33,55], cloud-edge and fog computing and task scheduling [3,5,6,16,23,50,67], and lastly securing networks through intrusion detection [2,31,43,62,63,68,72,74].

3 Proposed Method

3.1 Basic Multi-verse Optimizer Algorithm

The multi-verse optimizer algorithm belongs to population-based stochastic algorithm group. The inspiration for this algorithm is a physics concept based on multi-verse theory. It states that the possibility of multiple big bangs, or otherwise regarded in science as the birth of a universe, is real and that there are other universes. For such phenomena to exist certain characteristics are required. The universes should be able to overlap and communicate. The concepts from this theory that influence the MVO are wormholes, black holes, and white holes. These concepts are closely related to the previously mentioned interaction between universes. The big bang can be considered as a white hole, as physicists believe that it played a role in the birth of a universe. The black holes are the opposite having stronger gravity which attracts everything. Wormhole is concept in-between as the role for it has been determined as a bridge between universes. The algorithm respects the basic population-based algorithms key characteristics as it is divided into exploration and exploitation phases. For the exploration the white and black holes are applied to mathematical formulas, and the wormholes for exploitation.

The process of optimization is achieved along these rules:

- For high universe inflation rate, the greater the chance for a white hole.
- For low universe inflation rate, the chances are higher of a black hole.
- High inflation rate universes send objects by white holes, while the opposite receive them by black holes.
- All objects can spontaneously change universes towards the best one without regards to the inflation rate.

Therefore, the chance for objects to travel to universes with lower inflation rate through white holes is higher and this justifies the process of improvement over iterations. The model for the exchange of objects through tunnels is a roulette wheel mechanism. During all iterations, the universes are firstly sorted by their inflation rate, and choosing one for its chance to have a white hole.

Algorithm 1. Pseudocode for the original MVO algorithm

while $maxFFE$ is not satisfied:
for each universe (solution) as i:
  for each object (solution's component) as j:
    r_2 = random value between 0 and 1;
    if $r2$ lt Chance_of_wormhole_esixting:
      r_3 = random value between 0 and 1;
      r_4 = random value between 0 and 1;
      if $r3 < 0.5$:
        $U(i, j) = Optimal\_universe(j) + Travelling\_distance\_rate * ((ub(j) - lb(j))) * r_4 + lb(j)$
      else:
        $U(i, j) = Optimal\_universe(j) - Travelling\_distance\_rate * ((ub(j) - lb(j))) * r_4 + lb(j)$
      end if
    end if
  end for
end for
Replace each solution whose $trial = limit$ with pseudo-random individual from the search
end while
Show best solution, post-process results and visualization

3.2 Modified Multi-verse Optimizer Algorithm

A simple modification in the code has been introduced, to improve the capabilities of the original implementation of MVO. Each solution is assigned *trial* variable, that is initially set to 0, and incremented after each iteration. If a particular solution has not been improved after k rounds, it is removed from the populace, and replaced by a novel, randomly produced individual. Due to the small number of solutions and rounds, k was set to 2 in the experiments that follows. Modified algorithm has been simply named improved MVO - IMVO.

4 Experimental Reports and Discussion

In this section the dataset for network intrusion experiments is described alongside the used metrics of performance. Afterwards, the comparative analysis against other excellent performing techniques is performed.

4.1 Datasets and Metrics

The dataset chosen by authors is UNSW-NB 15 https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15, created with IXIA PerferctStorm toolset for raw network traffic collection. Cyber Range Laboratory of the Australian Center for Cyber Security (ACCS) has authored this dataset. The content of the dataset includes regular traffic observed, and mock cyber-attack patterns. Multi-class and binary classification are both supported by the dataset. The use in this work is with the binary classification. The dataset is pre-split into training and testing set, which were combined for the purposes of this research. In terms of preprocessing the data the string data type had to be converted to integers. Furthermore, the incomplete data was removed, and the data did not have a value of state was considered as such. Protocol state numerical values are: FIN - 0, INT - 1, CON - 2, ACC - 3, REQ - 4, RST - 5. The Fig. 1 shows the class distribution and the heatmap of the variable correlation in UNSW-NB 15 dataset.

Proper testing basis was established with the use of standard metrics related to this problem. True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). From their exploitation an accuracy formula is derived:

$$ACC = (TP + TN) \big/ (TP + FP + TN + FN) \tag{8}$$

Furthermore, precision, recall, and F-measure are used and calculated by the (9)-(11).

$$Precision = TP \big/ (TP + FP) \tag{9}$$

$$Recall(sensitivity) = TP \big/ (TP + FN) \tag{10}$$

$$F - measure = (2 \cdot Precision \cdot Recall)/(Precision + Recall) \tag{11}$$

The execution of test cases was performed over 8 iterations for 10 solutions, with 10 runs in overall. The parameters for XGBoost were setup as in [31] for the same ranges.

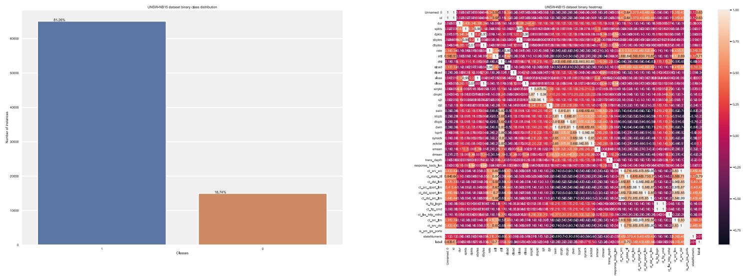

Fig. 1. Class distribution and heatmap of the correlation between the variables of the binary UNSW-NB 15 dataset

4.2 Experimental Findings and Comparative Analysis

Every compared metaheuristic has been tested over 30 independent runs with 8 iterations, while the XGBoost model was developed by each approach by optimization of the following six hyper-parameters within pre-defined boundaries:

- learning rate (η), boundaries: $[0.1, 0.9]$, type: continuous,
- *min_child_weight*, boundaries: $[0, 10]$, type: continuous,
- subsample, boundaries: $[0.01, 1]$, type: continuous,
- collsample_bytree, boundaries: $[0.01, 1]$, type: continuous,
- max_depth, boundaries: $[3, 10]$, type: integer and
- γ, boundaries: $[0, 0.5]$, type: continuous.

Thus the vector length of the solution is provided with $L = 6$. The parameters' boundaries were empirically decided. The presented IMVO was used to optimize the hyperparameters, and the approach was named XGBoost-IMVO.

The level of performance of the introduced XGBoost-IMVO model were tested against the basic MVO and five other superior metaheuristics, namely FA [59], BA [61], SCA [37], WOA [39] and TLB [47]. All other contenders were validated in the same experimental setup and with the same population size and number of rounds, with default control parameters. Table 1 presents the simulation results related to the obtained error over the course of 10 runs. The prime result for each metric has been noted in bold. The proposed XGBoost-IMVO model obtained the most predominant best, worst and mean values with respect to other contending metaheuristics, XGBoost-MVO obtained the best median, while XGBoost-BA achieved best Std and Var scores.

Table 1. Overall Metric

Method	Best	Worst	Mean	Median	Std	Var
XGBoost-IMVO	**3.12E-03**	**3.69E-03**	**3.39E-03**	3.44E-03	1.93E-04	3.73E-08
XGBoost-MVO	3.19E-03	4.25E-03	3.52E-03	**3.40E-03**	3.54E-04	1.26E-07
XGBoost-FA	3.19E-03	3.87E-03	3.58E-03	3.59E-03	2.06E-04	4.25E-08
XGBoost-BA	3.37E-03	3.87E-03	3.62E-03	3.62E-03	**1.49E-04**	**2.21E-08**
XGBoost-SCA	3.44E-03	4.43E-03	3.74E-03	3.69E-03	3.37E-04	1.14E-07
XGBoost-WOA	3.37E-03	3.81E-03	3.61E-03	3.62E-03	1.50E-04	2.26E-08
XGBoost-TLB	3.62E-03	4.50E-03	3.99E-03	3.97E-03	3.19E-04	1.02E-07

Table 2. Detailed Metric

	XGB-IMVO	XGB-MVO	XGB-FA	XGB-BA	XGB-SCA	XGB-WOA	XGB-TLB
Accuracy (%)	**99.6878**	99.6815	99.6815	99.6628	99.6565	99.6628	42.3156
Precision 0	0.992323	0.992321	0.992321	0.990346	**0.992640**	0.991656	0.185667
Precision 1	0.997926	0.997849	0.997849	**0.998078**	0.997466	0.997772	0.809915
M.Avg. Precision	**0.996876**	0.996813	0.996813	0.996630	0.996562	0.996626	0.692963
Recall 0	0.991000	0.990667	0.990667	**0.991667**	0.989000	0.990333	0.614000
Recall 1	0.998233	0.998233	0.998233	0.997771	**0.998309**	0.998079	0.379159
M.Avg. Recall	**0.996878**	0.996815	0.996815	0.996628	0.996565	0.996628	0.423156
F1-score 0	**0.991661**	0.991493	0.991493	0.991006	0.990816	0.990994	0.285117
F1-score 1	**0.998079**	0.998041	0.998041	0.997925	0.997888	0.997925	0.516514
M.Avg. F1-score	**0.996877**	0.996814	0.996814	0.996629	0.996563	0.996627	0.473162

Table 2 depicts the detailed experimental outcomes on the UNSW-NB 15 dataset, and it is obvious that the XGBoost-IMVO scored the highest accuracy of 99.6878%, leaving gehind XGBoost-MVO and XGBoost-FA that scored 99.6815%.

In order to graphically present the superiority of the XGBoost-IMVO approach over other contenders, Fig. 2 depicts the convergence graphs and box plots of the error and the objective function, for all described algorithms. To provide more insight into the results achieved by XGBoost-IMVO, Figs. 3, 4 and 5 yield precision-recall (ROC) and receiver under operating characteristics (ROC), area below the curve (AUC), confusion matrix over the employed dataset, and one versus rest (OvR) ROC curves with respect to the best individual.

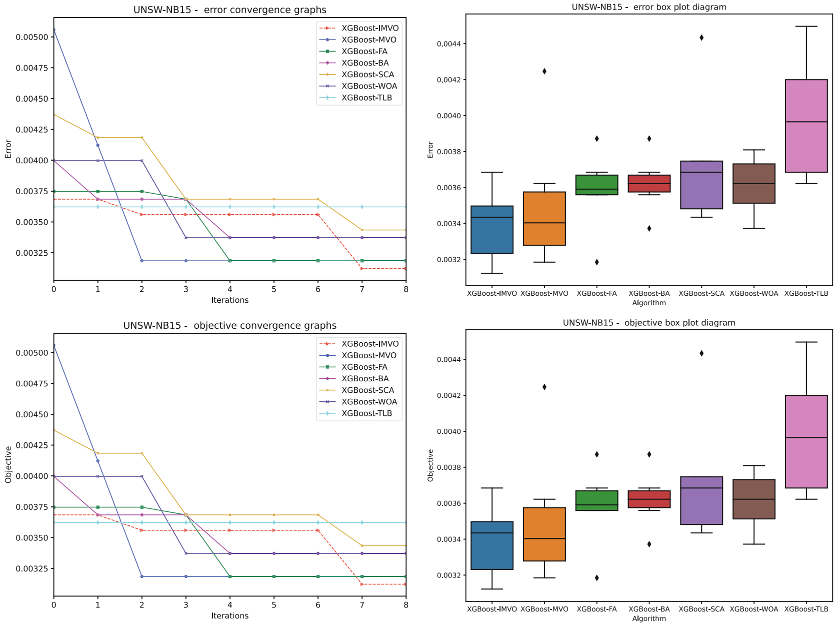

Fig. 2. Convergence and box plot graphics for all contenders

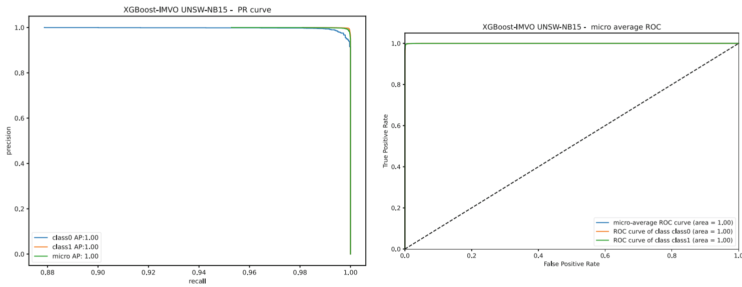

Fig. 3. PR and ROC plots of the presented XGBoost-IMVO model over the UNSW-NB 15 dataset

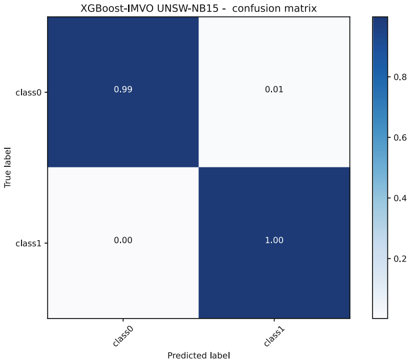

Fig. 4. The XGBoost-IMVO confusion matrix for UNSW-NB 15 dataset

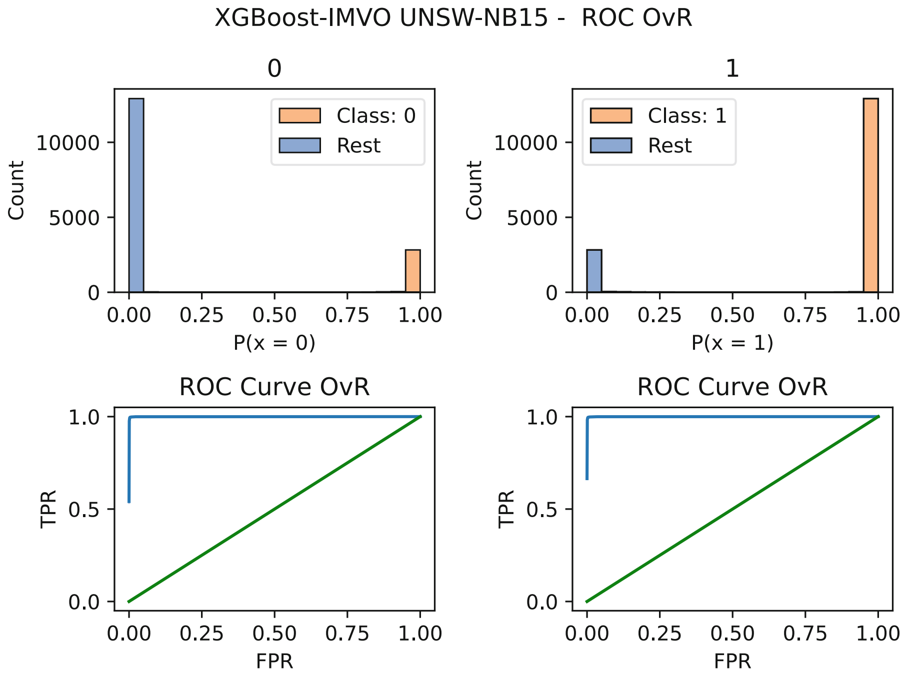

Fig. 5. The XGBoost-IMVO one versus rest (OvR) ROC

5 Conclusion

The presented work in this paper contributes an XGBoost algorithm optimized by the modified multi-verse optimizer contributed in this work. The proposed method has been evaluated on the intrusion detection benchmark dataset UNSW-NB 15, while the comparison was performed for 5 other metaheuristic techniques that have fared well for this problem on the same dataset. The conclusion is that the improvements were achieved as the proposed solution outperformed other solutions. Furthermore, this makes the proposed method suitable for real-life usage. Regarding the space for improvement, the proposed method will be tested on other datasets of the same type for the reasons of furthering the stability of the results for better practical application of the solution.

References

1. Abdulrahman, S.M.: Using swarm intelligence for solving np-hard problems. Academic Journal of Nawroz University **6**(3), 46–50 (2017)
2. AlHosni, N., Jovanovic, L., Antonijevic, M., Bukumira, M., Zivkovic, M., Strumberger, I., Mani, J.P., Bacanin, N.: The xgboost model for network intrusion detection boosted by enhanced sine cosine algorithm. In: International Conference on Image Processing and Capsule Networks. pp. 213–228. Springer (2022)
3. Antonijevic, M., Strumberger, I., Lazarevic, S., Bacanin, N., Mladenovic, D., Jovanovic, D.: Robust encrypted face recognition robot based on bit slicing and fourier transform for cloud environments. Journal of Electronic Imaging **31**(6), 061808 (2022)
4. Bacanin, N., Antonijevic, M., Bezdan, T., Zivkovic, M., Rashid, T.A.: Wireless sensor networks localization by improved whale optimization algorithm. In: Proceedings of 2nd International Conference on Artificial Intelligence: Advances and Applications. pp. 769–783. Springer (2022)
5. Bacanin, N., Antonijevic, M., Bezdan, T., Zivkovic, M., Venkatachalam, K., Malebary, S.: Energy efficient offloading mechanism using particle swarm optimization in 5g enabled edge nodes. Cluster Computing pp. 1–12 (2022)
6. Bacanin, N., Arnaut, U., Zivkovic, M., Bezdan, T., Rashid, T.A.: Energy efficient clustering in wireless sensor networks by opposition-based initialization bat algorithm. In: Computer Networks and Inventive Communication Technologies, pp. 1–16. Springer (2022)
7. Bacanin, N., Bezdan, T., Venkatachalam, K., Zivkovic, M., Strumberger, I., Abouhawwash, M., Ahmed, A.: Artificial neural networks hidden unit and weight connection optimization by quasi-refection-based learning artificial bee colony algorithm. IEEE Access (2021)
8. Bacanin, N., Bezdan, T., Zivkovic, M., Chhabra, A.: Weight optimization in artificial neural network training by improved monarch butterfly algorithm. In: Mobile Computing and Sustainable Informatics, pp. 397–409. Springer (2022)
9. Bacanin, N., Petrovic, A., Zivkovic, M., Bezdan, T., Antonijevic, M.: Feature selection in machine learning by hybrid sine cosine metaheuristics. In: International Conference on Advances in Computing and Data Sciences. pp. 604–616. Springer (2021)
10. Bacanin, N., Petrovic, A., Zivkovic, M., Bezdan, T., Chhabra, A.: Enhanced salp swarm algorithm for feature selection. In: International Conference on Intelligent and Fuzzy Systems. pp. 483–491. Springer (2021)

11. Bacanin, N., Sarac, M., Budimirovic, N., Zivkovic, M., AlZubi, A.A., Bashir, A.K.: Smart wireless health care system using graph lstm pollution prediction and dragonfly node localization. Sustainable Computing: Informatics and Systems **35**, 100711 (2022)
12. Bacanin, N., Stoean, C., Zivkovic, M., Jovanovic, D., Antonijevic, M., Mladenovic, D.: Multi-swarm algorithm for extreme learning machine optimization. Sensors **22**(11), 4204 (2022)
13. Bacanin, N., Tuba, E., Zivkovic, M., Strumberger, I., Tuba, M.: Whale optimization algorithm with exploratory move for wireless sensor networks localization. In: International Conference on Hybrid Intelligent Systems. pp. 328–338. Springer (2019)
14. Bacanin, N., Zivkovic, M., Al-Turjman, F., Venkatachalam, K., Trojovskỳ, P., Strumberger, I., Bezdan, T.: Hybridized sine cosine algorithm with convolutional neural networks dropout regularization application. Scientific Reports **12**(1), 1–20 (2022)
15. Bacanin, N., Zivkovic, M., Bezdan, T., Cvetnic, D., Gajic, L.: Dimensionality reduction using hybrid brainstorm optimization algorithm. In: Proceedings of International Conference on Data Science and Applications. pp. 679–692. Springer (2022)
16. Bacanin, N., Zivkovic, M., Bezdan, T., Venkatachalam, K., Abouhawwash, M.: Modified firefly algorithm for workflow scheduling in cloud-edge environment. Neural Computing and Applications **34**(11), 9043–9068 (2022)
17. Basha, J., Bacanin, N., Vukobrat, N., Zivkovic, M., Venkatachalam, K., Hubálovskỳ, S., Trojovskỳ, P.: Chaotic harris hawks optimization with quasi-reflection-based learning: An application to enhance cnn design. Sensors **21**(19), 6654 (2021)
18. Beni, G.: Swarm intelligence. Complex Social and Behavioral Systems: Game Theory and Agent-Based Models pp. 791–818 (2020)
19. Bezdan, T., Cvetnic, D., Gajic, L., Zivkovic, M., Strumberger, I., Bacanin, N.: Feature selection by firefly algorithm with improved initialization strategy. In: 7th Conference on the Engineering of Computer Based Systems. pp. 1–8 (2021)
20. Bezdan, T., Milosevic, S., Venkatachalam, K., Zivkovic, M., Bacanin, N., Strumberger, I.: Optimizing convolutional neural network by hybridized elephant herding optimization algorithm for magnetic resonance image classification of glioma brain tumor grade. In: 2021 Zooming Innovation in Consumer Technologies Conference (ZINC). pp. 171–176. IEEE (2021)
21. Bezdan, T., Stoean, C., Naamany, A.A., Bacanin, N., Rashid, T.A., Zivkovic, M., Venkatachalam, K.: Hybrid fruit-fly optimization algorithm with k-means for text document clustering. Mathematics **9**(16), 1929 (2021)
22. Bezdan, T., Zivkovic, M., Bacanin, N., Chhabra, A., Suresh, M.: Feature selection by hybrid brain storm optimization algorithm for covid-19 classification. Journal of Computational Biology (2022)
23. Bezdan, T., Zivkovic, M., Bacanin, N., Strumberger, I., Tuba, E., Tuba, M.: Multi-objective task scheduling in cloud computing environment by hybridized bat algorithm. Journal of Intelligent & Fuzzy Systems **42**(1), 411–423 (2022)
24. Bezdan, T., Zivkovic, M., Tuba, E., Strumberger, I., Bacanin, N., Tuba, M.: Glioma brain tumor grade classification from mri using convolutional neural networks designed by modified FA. In: International Conference on Intelligent and Fuzzy Systems. pp. 955–963. Springer (2020)

25. Budimirovic, N., Prabhu, E., Antonijevic, M., Zivkovic, M., Bacanin, N., Strumberger, I., Venkatachalam, K.: Covid-19 severity prediction using enhanced whale with salp swarm feature classification. Computers, Materials and Continua pp. 1685–1698 (2022)
26. Bukumira, M., Antonijevic, M., Jovanovic, D., Zivkovic, M., Mladenovic, D., Kunjadic, G.: Carrot grading system using computer vision feature parameters and a cascaded graph convolutional neural network. Journal of Electronic Imaging **31**(6), 061815 (2022)
27. Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., et al.: Xgboost: extreme gradient boosting. R package version 0.4-2 **1**(4), 1–4 (2015)
28. Dick, S.: Artificial intelligence (2019)
29. El Naqa, I., Murphy, M.J.: What is machine learning? In: machine learning in radiation oncology, pp. 3–11. Springer (2015)
30. Gumus, M., Kiran, M.S.: Crude oil price forecasting using xgboost. In: 2017 International conference on computer science and engineering (UBMK). pp. 1100–1103. IEEE (2017)
31. Jovanovic, D., Antonijevic, M., Stankovic, M., Zivkovic, M., Tanaskovic, M., Bacanin, N.: Tuning machine learning models using a group search firefly algorithm for credit card fraud detection. Mathematics **10**(13), 2272 (2022)
32. Jovanovic, D., Marjanovic, M., Antonijevic, M., Zivkovic, M., Budimirovic, N., Bacanin, N.: Feature selection by improved sand cat swarm optimizer for intrusion detection. In: 2022 International Conference on Artificial Intelligence in Everything (AIE). pp. 685–690. IEEE (2022)
33. Jovanovic, L., Zivkovic, M., Antonijevic, M., Jovanovic, D., Ivanovic, M., Jassim, H.S.: An emperor penguin optimizer application for medical diagnostics. In: 2022 IEEE Zooming Innovation in Consumer Technologies Conference (ZINC). pp. 191–196. IEEE (2022)
34. Karaboga, D.: Artificial bee colony algorithm. scholarpedia **5**(3), 6915 (2010)
35. Kiangala, S.K., Wang, Z.: An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-xgboost and random forest ensemble learning algorithms in an industry 4.0 environment. Machine Learning with Applications **4**, 100024 (2021)
36. Latha, R.S., Saravana Balaji, B., Bacanin, N., Strumberger, I., Zivkovic, M., Kabiljo, M.: Feature selection using grey wolf optimization with random differential grouping. Computer Systems Science and Engineering **43**(1), 317–332 (2022). https://doi.org/10.32604/csse.2022.020487, http://www.techscience.com/csse/v43n1/47062
37. Mirjalili, S.: Sca: a sine cosine algorithm for solving optimization problems. Knowledge-based systems **96**, 120–133 (2016)
38. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. Advances in engineering software **114**, 163–191 (2017)
39. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Advances in engineering software **95**, 51–67 (2016)
40. Mirjalili, S., Mirjalili, S.M., Hatamlou, A.: Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications **27**(2), 495–513 (2016)
41. Mirjalili, S.Z., Mirjalili, S., Saremi, S., Faris, H., Aljarah, I.: Grasshopper optimization algorithm for multi-objective optimization problems. Applied Intelligence **48**(4), 805–820 (2018)

42. Muslim, M.A., Dasril, Y.: Company bankruptcy prediction framework based on the most influential features using xgboost and stacking ensemble learning. International Journal of Electrical & Computer Engineering (2088-8708) **11**(6) (2021)
43. Petrovic, A., Bacanin, N., Zivkovic, M., Marjanovic, M., Antonijevic, M., Strumberger, I.: The adaboost approach tuned by firefly metaheuristics for fraud detection. In: 2022 IEEE World Conference on Applied Intelligence and Computing (AIC). pp. 834–839. IEEE (2022)
44. Petrovic, A., Strumberger, I., Bezdan, T., Jassim, H.S., Suleiman Nassor, S.: Cryptocurrency price prediction by using hybrid machine learning and beetle antennae search approach. In: 2021 29th Telecommunications Forum (TELFOR). pp. 1–4 (2021). https://doi.org/10.1109/TELFOR52709.2021.9653305
45. Pham, Q.V., Mirjalili, S., Kumar, N., Alazab, M., Hwang, W.J.: Whale optimization algorithm with applications to resource allocation in wireless networks. IEEE Transactions on Vehicular Technology **69**(4), 4285–4297 (2020)
46. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. Swarm intelligence **1**(1), 33–57 (2007)
47. Rao, R.V.: Teaching-learning-based optimization algorithm. In: Teaching learning based optimization algorithm, pp. 9–39. Springer (2016)
48. Salb, M., Bacanin, N., Zivkovic, M., Antonijevic, M., Marjanovic, M., Strumberger, I.: Extreme learning machine tuning by original sine cosine algorithm. In: 2022 IEEE World Conference on Applied Intelligence and Computing (AIC). pp. 143–148. IEEE (2022)
49. Salb, M., Zivkovic, M., Bacanin, N., Chhabra, A., Suresh, M.: Support vector machine performance improvements for cryptocurrency value forecasting by enhanced sine cosine algorithm. In: Computer Vision and Robotics, pp. 527–536. Springer (2022)
50. Sreekanth, G., Ahmed, S.A.N., Sarac, M., Strumberger, I., Bacanin, N., Zivkovic, M.: Mobile fog computing by using sdn/nfv on 5g edge nodes. Comput. Syst. Sci. Eng. **41**(2), 751–765 (2022)
51. Stone-Gross, B., Cova, M., Cavallaro, L., Gilbert, B., Szydlowski, M., Kemmerer, R., Kruegel, C., Vigna, G.: Your botnet is my botnet: analysis of a botnet takeover. In: Proceedings of the 16th ACM conference on Computer and communications security. pp. 635–647 (2009)
52. Strumberger, I., Bezdan, T., Ivanovic, M., Jovanovic, L.: Improving energy usage in wireless sensor networks by whale optimization algorithm. In: 2021 29th Telecommunications Forum (TELFOR). pp. 1–4. IEEE (2021)
53. Strumberger, I., Rakic, A., Stanojlovic, S., Arandjelovic, J., Bezdan, T., Zivkovic, M., Bacanin, N.: Feature selection by hybrid binary ant lion optimizer with covid-19 dataset. In: 2021 29th Telecommunications Forum (TELFOR). pp. 1–4. IEEE (2021)
54. Strumberger, I., Tuba, E., Bacanin, N., Zivkovic, M., Beko, M., Tuba, M.: Designing convolutional neural network architecture by the firefly algorithm. In: 2019 International Young Engineers Forum (YEF-ECE). pp. 59–65. IEEE (2019)
55. Tair, M., Bacanin, N., Zivkovic, M., Venkatachalam, K.: A chaotic oppositional whale optimisation algorithm with firefly search for medical diagnostics. Computers, Materials & Continua **72**(1), 959–982 (2022). https://doi.org/10.32604/cmc.2022.024989, http://www.techscience.com/cmc/v72n1/46919
56. Wang, G.G., Deb, S., Coelho, L.d.S.: Elephant herding optimization. In: 2015 3rd international symposium on computational and business intelligence (ISCBI). pp. 1–5. IEEE (2015)

57. Wang, G.G., Deb, S., Cui, Z.: Monarch butterfly optimization. Neural computing and applications **31**(7), 1995–2014 (2019)
58. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE transactions on evolutionary computation **1**(1), 67–82 (1997)
59. Yang, X.S.: Firefly algorithms for multimodal optimization. In: International symposium on stochastic algorithms. pp. 169–178. Springer (2009)
60. Yang, X.S.: Firefly algorithm, stochastic test functions and design optimisation. arXiv preprint arXiv:1003.1409 (2010)
61. Yang, X.S.: Bat algorithm for multi-objective optimisation. International Journal of Bio-Inspired Computation **3**(5), 267–274 (2011)
62. Zivkovic, M., Bacanin, N., Arandjelovic, J., Rakic, A., Strumberger, I., Venkatachalam, K., Joseph, P.M.: Novel harris hawks optimization and deep neural network approach for intrusion detection. In: Proceedings of International Joint Conference on Advances in Computational Intelligence. pp. 239–250. Springer (2022)
63. Zivkovic, M., Bacanin, N., Arandjelovic, J., Strumberger, I., Venkatachalam, K.: Firefly algorithm and deep neural network approach for intrusion detection. In: Applications of Artificial Intelligence and Machine Learning, pp. 1–12. Springer (2022)
64. Zivkovic, M., Bacanin, N., Djordjevic, A., Antonijevic, M., Strumberger, I., Rashid, T.A., et al.: Hybrid genetic algorithm and machine learning method for covid-19 cases prediction. In: Proceedings of International Conference on Sustainable Expert Systems. pp. 169–184. Springer (2021)
65. Zivkovic, M., Bacanin, N., Tuba, E., Strumberger, I., Bezdan, T., Tuba, M.: Wireless sensor networks life time optimization based on the improved firefly algorithm. In: 2020 International Wireless Communications and Mobile Computing (IWCMC). pp. 1176–1181. IEEE (2020)
66. Zivkovic, M., Bacanin, N., Venkatachalam, K., Nayyar, A., Djordjevic, A., Strumberger, I., Al-Turjman, F.: Covid-19 cases prediction by using hybrid machine learning and beetle antennae search approach. Sustainable Cities and Society **66**, 102669 (2021)
67. Zivkovic, M., Bezdan, T., Strumberger, I., Bacanin, N., Venkatachalam, K.: Improved harris hawks optimization algorithm for workflow scheduling challenge in cloud–edge environment. In: Computer Networks, Big Data and IoT, pp. 87–102. Springer (2021)
68. Zivkovic, M., Jovanovic, L., Ivanovic, M., Bacanin, N., Strumberger, I., Joseph, P.M.: Xgboost hyperparameters tuning by fitness-dependent optimizer for network intrusion detection. In: Communication and Intelligent Systems, pp. 947–962. Springer (2022)
69. Zivkovic, M., Jovanovic, L., Ivanovic, M., Krdzic, A., Bacanin, N., Strumberger, I.: Feature selection using modified sine cosine algorithm with covid-19 dataset. In: Evolutionary Computing and Mobile Sustainable Networks, pp. 15–31. Springer (2022)
70. Zivkovic, M., Petrovic, A., Bacanin, N., Milosevic, S., Veljic, V., Vesic, A.: The covid-19 images classification by mobilenetv3 and enhanced sine cosine metaheuristics. In: Shakya, S., Ntalianis, K., Kamel, K.A. (eds.) Mobile Computing and Sustainable Informatics. pp. 937–950. Springer Nature Singapore, Singapore (2022)
71. Zivkovic, M., Petrovic, A., Venkatachalam, K., Strumberger, I., Jassim, H.S., Bacanin, N.: Novel chaotic best firefly algorithm: Covid-19 fake news detection application. In: Advances in Swarm Intelligence, pp. 285–305. Springer (2023)

72. Zivkovic, M., Stoean, C., Chhabra, A., Budimirovic, N., Petrovic, A., Bacanin, N.: Novel improved salp swarm algorithm: An application for feature selection. Sensors **22**(5), 1711 (2022)
73. Zivkovic, M., Stoean, C., Petrovic, A., Bacanin, N., Strumberger, I., Zivkovic, T.: A novel method for covid-19 pandemic information fake news detection based on the arithmetic optimization algorithm. In: 2021 23rd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC). pp. 259–266 (2021). https://doi.org/10.1109/SYNASC54541.2021.00051
74. Zivkovic, M., Tair, M., Venkatachalam, K., Bacanin, N., Hubálovskỳ, Š., Trojovskỳ, P.: Novel hybrid firefly algorithm: an application to enhance xgboost tuning for intrusion detection classification. PeerJ Computer Science **8**, e956 (2022)
75. Zivkovic, M., Zivkovic, T., Venkatachalam, K., Bacanin, N.: Enhanced dragonfly algorithm adapted for wireless sensor network lifetime optimization. In: Data Intelligence and Cognitive Informatics, pp. 803–817. Springer (2021)