Babylonian Journal of Internet of Things
Vol.2026, **pp**. 197–205
DOI: https://doi.org/10.58496/BJIoT/2026/013; ISSN: 3006-1083
https://mesopotamian.press/journals/index.php/BJIoT

Research Article

An Enhanced Hybrid Memetic mRMR–FOA Framework for Feature Selection in IoT Intrusion Detection Systems

Yusra Sh Ajaj [1,*]

[1] *University of Baghdad, Baghdad, Iraq.*

ARTICLE INFO

Article History
Received 10 Jun 2026
Revised 15 Jul 2026
Accepted 12 Aug 2026
Published 04 Sep 2026

Keywords
Internet of Things,
Intrusion detection,
Feature selection,
Memetic algorithm,
Fossa Optimization Algorithm.

ABSTRACT

The rapid expansion of the Internet of Things (IoT) has enlarged the attack surface of connected systems, increasing their exposure to sophisticated cyber-attacks and creating a strong demand for effective Intrusion Detection Systems (IDS). High-dimensional network traffic, however, degrades both the accuracy and the efficiency of IDS classifiers. This paper proposes a hybrid feature-selection framework, Memetic mRMR-FOA, that couples the Minimum Redundancy Maximum Relevance (mRMR) filter with a memetic extension of the Fossa Optimization Algorithm (FOA). Within the framework, mRMR plays a single, well-defined role: it ranks features by an information-theoretic relevance-redundancy score and this ranking is used both to seed a high-quality initial population and to guide the memetic local search. FOA performs the global search, while a Hamming-neighbourhood local search refines elite solutions. The objective is a minimization fitness that jointly rewards classification accuracy and low redundancy while penalizing subset size. On the RT-IoT2022 benchmark, the proposed method attains the best feature-reduction ratio of 68.3 percent, selecting only 13 features at the lowest fitness value of 0.038, and improves detection accuracy to 98.43 percent. Comparative analysis against Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Grey Wolf Optimizer (GWO), standard FOA, a memetic symmetrical-uncertainty variant, and standalone mRMR confirms consistent gains in accuracy, precision, and false-alarm rate. The results indicate that the Memetic mRMR-FOA framework is an efficient and scalable feature-selection strategy for protecting resource-constrained IoT environments.

1. INTRODUCTION

Nature-inspired optimization and swarm-intelligence methods have had a marked impact on modern data-driven systems that operate over large-scale, complex, and high-dimensional problem domains [1]–[5]. These paradigms provide a methodological foundation for intelligent decision-making and optimization in emerging cyber-physical environments.

Internet of Things (IoT) has revolutionized the communication between digital and physical world through persistent communication between sensors, devices and cloud services [6] – [8]. It is a very interconnected ecosystem and its applications include transportation, smart agriculture, industrial automation and healthcare, making operations and real-time decisions more efficient and effective [6]. As connected systems continue to grow and spread around the world, industry reports predict tens of billions of IoT connections by 2030 [9, 10]. In the context of this paradigm, the Internet of Things (IoT) is emerging as one of the critical technologies in smart manufacturing, using real-time data acquisition and intelligent automation to enhance productivity, efficiency and operational resilience [11]. But, the wide-spread interconnection of the IIoT environments has also expanded the cyber surface of attack. Devices, protocols and networks are not homogenous, which means that additional vulnerabilities exist that adversaries can exploit to potentially compromise the confidentiality, integrity and availability of data [11]–[14].

Intrusion Detection Systems (IDS) have become key components of IIoT security architectures for mitigating these risks. Network-based IDS (NIDS) watches traffic flows to identify surprising or damaging activity in particular [15]. However, the volume, velocity and variety of the IIoT data poses many challenges to traditional IDS solutions. Static or rule-based models are not able to cope with novel attack patterns and high-dimensional feature space, thus reducing detection accuracy and increasing computational complexity [11, 13]. As a result, feature selection (FS) has become one of the most significant preprocessing phases of IDS pipelines: its goal is to select only the most discriminative and informative features and to eliminate redundant and noisy features [16]–[19].

*Corresponding author. Email: yusra.s.955@cois.uobaghdad.edu.iq

The feature selection problem in IDS is inherently difficult to solve as an optimization problem. In many practical systems, the formulations are nonlinear, multimodal, and high dimensional, thus classical optimization methods [20]–[22] are not very efficient. In this context, the flexibility and scalability of nature inspired metaheuristics, which are able to balance exploration and exploitation in complex search spaces, are appealing [20] [23]–[27].

Swarm Intelligence (SI) algorithms are based on studying the collective behavior of social organisms for efficient solving of optimization problems [1]. The following are typical algorithms that have been used to feature selection and other IDS applications: Ant Colony Optimization (ACO) [2], Artificial Bee Colony (ABC) [3] and Grey Wolf Optimizer (GWO) [4] and [28]. Later, the Fossa Optimization Algorithm (FOA) was suggested as a bioinspired SI algorithm based on the two stage cooperative hunting behavior of the fossa, combining the global exploration and strategic exploitation [5].

FOA is promising, but similar to other swarm-based algorithms, premature convergence and lack of final exploitation of elite solutions are problems in high dimensional problems [5]. In response to these limitations, this study proposes a Memetic Fossa Optimization Algorithm (M-FOA), which introduces the memetic (local-learning) computing into the FOA algorithm. M-FOA introduces a single, unmistakable local search operator, which fine tunes elite solutions by examining the Hamming-one neighbourhood of the elite solutions in the binary feature space, in addition to the global FOA search. The Memetic mRMR-FOA framework is thus developed by combining the global search with the local search and the mRMR filter to explore the large feature space in the IIoT and boost the feature discriminability and reduce the feature redundancy. Compared to the traditional swarm-based feature selection [20] and [23]–[27] this balanced design reduces early convergence and increases generalization.

In this work some representative state-of-the-art studies are summarized in Table I. It demonstrates that the feature selection has been a key factor in the struggle of recent IoT/IIoT IDS to deal with high-dimensional network traffic and to enhance detection. Classical dimensionality reduction techniques like filter and static wrapper reduction have been employed but are not adaptive; metaheuristic and swarm techniques have been suggested to help better explore the global search space, but these methods tend to suffer from premature convergence or inadequate local search. As a result, various recent studies have resorted to the use of hybrid (multi-stage optimization/learning) techniques, which combine two or more optimization/learning methods. The proposed M-FOA-based IDS does this by implementing memetic optimization, and it offers a more balanced global exploration/adaptive local refinement that directly tackles some of the limitations discussed in the literature.

The main contributions of this paper are:

1. An extension of FOA that uses Hamming-neighbourhood local search (with a Hamming distance) to enhance the convergence and stability in binary feature selection.
2. A one-criterion mRMR-based workflow, where the mRMR is always unambiguously used to initialise the population and to guide the local search, without any confusion on the role of mRMR.
3. Comprehensive experimental comparison on the RT-IoT2022 benchmark that showed the solution quality outperforms FOA and other SI and filter techniques, and the results were statistically tested using non-parametric methods.

The remainder of this paper will be structured as follows. A brief overview of FOA is provided in Section 2. In this section, the proposed Memetic mRMR-FOA framework is presented. The experimental setup and results are reported in section 4. The paper is concluded in Section 5 and future work is outlined.

TABLE I. INSIGHTS FROM STATE-OF-THE-ART FEATURE SELECTION METHODS FOR IOT/IIOT INTRUSION DETECTION SYSTEMS.

Ref	Year	Method	FS Strategy	Classifier / IDS	Dataset / Scenario	Key Strengths	Limitations
[12]	2024	FS vs FE IoT-IDS	Filter & wrapper FS	RF, SVM, KNN	TON-IoT	Improves accuracy & runtime	No adaptive optimization
[13]	2024	FS-anomaly IoT IDS	Correlation-based FS	ML anomaly detector	RT-IoT2022	Lightweight & interpretable	Limited global optimization
[14]	2024	Balanced FS-IDS	ANOVA-based FS	ML classifiers	IoT traffic	Handles class imbalance	Static FS
[15]	2022	Deep AE IDS	Two-layer optimizer FS	KNN + autoencoder	N-BaIoT	Comprehensive FS comparison	High computational cost
[16]	2024	ML-stack + FS IDS	Embedded FS	Ensemble ML	IoT networks	High detection accuracy	High computational cost
[17]	2025	FS + augmentation IDS	Multi-stage FS	Deep-learning IDS	IoT traffic	Handles imbalance & redundancy	Not optimization-driven

Ref	Year	Method	FS Strategy	Classifier / IDS	Dataset / Scenario	Key Strengths	Limitations
[18]	2024	Enhanced FS-IoT IDS	Wrapper FS	ML IDS	IoT networks	Reduced dimensionality	Local-optima risk
[19]	2024	Systematic review	Survey-based	ML/DL IDS	IoT / IIoT	Comprehensive taxonomy	No new algorithm
[20]	2023	Two-phase FS	Filter + wrapper FS	ML classifiers	IDS datasets	Reduces redundancy	Lacks adaptive search
[21]	2021	Metaheuristic FS-IDS	Nature-inspired FS	SVM, RF	Network IDS	Global optimization	Convergence instability
[23]	2023	Hybrid metaheuristics IDS	Hybrid heuristic FS	ML IDS	Cybersecurity datasets	Improved exploration	High tuning complexity
[24]	2023	Swarm-based FS IDS	SI-based FS	ML IDS	IDS benchmarks	Handles high dimensionality	Weak local refinement
[25]	2024	Hybrid optimization FS-IDS	Hybrid heuristic FS	ML IDS	Network IDS	Improved accuracy	Static exploitation
[26]	2024	Nature-inspired IDS	Heuristic FS	IDS models	IoT / IIoT	Robust search	High runtime
This work	2026	M-FOA-based IDS	Memetic wrapper FS (mRMR-guided)	NIDS (ML)	RT-IoT2022 (IIoT)	Balanced exploration/exploitation, fast convergence	Higher design complexity

2. PRELIMINARIES: FOSSA OPTIMIZATION ALGORITHM

This section presents the brief summary of the Fossa Optimization Algorithm (FOA) used for the proposed framework. FOA is inspired by the hunting behaviour of the fossa Cryptoprocta ferox which is a Madagascarian endoant species of carnivore, belonging to the family Eupleridae. The fossa uses a unique two-part approach in its lemur hunting: (i) a jumping leap towards a spotted lemur; (ii) a jumping follow-up through the treetops until it reaches the prey. FOA represents these two phases mathematically as 'Exploration' and 'Exploitation' phases.

2.1 FOA Initialization

FOA FOA is a population-based algorithm, each fossa is a candidate solution. The habitat is the search space and the location of a fossa in the space defines a candidate solution as a vector with components that are decision variables. The matrix in equation (1) represents the population of N fossae in an m-dimensional space, and the equation (2) shows the expanded element-wise version of the matrix. The position of each fossa is initialised randomly in the range of the variable in each of the fossa according to Equation (3).

$$X = [X_1; X_i; \dots ; X_N]^T \ (N \times m) \quad (1)$$

$$X = [x_{i,j}], \ i = 1, \dots, N; \ j = 1, \dots, m \quad (2)$$

$$x_{i,j} = lb_j + r \cdot (ub_j - lb_j) \quad (3)$$

Here, X is the FOA population matrix; X_i is the i^{th} fossa (a candidate solution); $x_{i,j}$ is the j-th dimension of the i-th fossa; N is the number of fossae; m is the number of decision variables; $r \in [0,1]$ is a uniform random number; and lb_j and ub_j are the lower and upper bounds of the j^{th} variable, respectively.

Each candidate position is assessed by the objective (fitness) function, and the evaluated values are collected in the vector F defined in Equation (4).

$$F = [F(X_1), F(X_i), \dots , F(X_N)]^T \ (N \times 1) \quad (4)$$

where $F_i = F(X_i)$ is the objective value of the i^{th} fossa.

2.2 FOA Strategic Movements

FOA updates the positions of its members in two distinct phases that imitate the strategic movement of fossae in nature.

Exploration phase. This phase mimics the first attack, in which a fossa jumps at a spotted lemur. This algorithm is a general solution for searching the space for interesting areas. The set of candidate lemurs for fossa i includes the members of the population whose objective values are strictly less (smaller) than the objective value of fossa i, based on Equation (5). A candidate SL is drawn at random from this set and used to compute an exploratory displacement that is large in Equation (6). A greedy rule (Equation (7)) will only take the new position if it does not degrade the fitness.

$$CL_i = \{ X_k : F_k < F_i, \ k \neq i \} \quad (5)$$

where the indices i, k ∈ {1, …, N}.

$$x_{i,j}^{(P1)} = x_{i,j} + r_{i,j} \cdot (SL_{i,j} - I_{i,j} \cdot x_{i,j}) \quad (6)$$

$$X_i = X_i^{(P1)} \; if \; F_i^{(P1)} \leq F_i \; ; \; otherwise \; X_i \quad (7)$$

Exploitation phase. This phase is a reversion of the pursuit through the trees, with an improved attack in the vicinity of a favorable spot. Both fossa take a small random step (scaled by the variable range and annealed by the iteration index t, as in Equation (8)), and only allow moves that are not worse than the greedy rule (Equation (9)). The smaller the step size the more intense the search locally, as t is increased.

$$x_{i,j}^{(P2)} = x_{i,j}^{(P1)} + (1 - 2\, r_{i,j}) \cdot \frac{ub_j - lb_j}{t} \quad (8)$$

$$X_i = X_i^{(P2)} \; if \; F_i^{(P2)} \leq F_i \; ; \; otherwise \; X_i \quad (9)$$

3. PROPOSED MEMETIC MRMR-FOA FRAMEWORK

The proposed Memetic mRMR-Feature selection approach is introduced. The framework is composed of three parts: (i) the mRMR filter that gives an information-theoretic ranking for features; (ii) the FOA global search, which performs a search over the binary feature space; and (iii) a memetic local search that refines elite solutions. We make explicit the role of each component in the description below, to avoid the ambiguity of previous descriptions, and use the same local-search operator throughout.

3.1 Problem Formulation

Let Given a dataset of intrusions, let the intrusion-dataset be denoted as in Equation (10): d features per instance, with a class label. Each candidate feature subset is represented by a binary vector z (Equation (11)): if z_j = 1, the feature j is kept, otherwise, it is discarded. The index set of the ones of z is the selected subset S(z) (Equation (12)).

$$D = \{ (x^{(i)}, y^{(i)}) \}, \; i = 1, \ldots, N, \; x^{(i)} \in \mathbb{R}^d \quad (10)$$

$$z = [z_1, z_2, \ldots, z_d], \; z_j \in \{0, 1\} \quad (11)$$

$$S(z) = \{ j \mid z_j = 1 \} \quad (12)$$

3.2 The mRMR Criterion

Two information-theoretic terms are used to quantify the quality of features in the mRMR filter. The relevance of feature j to the class label Y is the normalized mutual information in Equation (13); the mean relevance of a subset is in Equation (14). The normalized mutual information (Equation (15)) is called the redundancy between two features, and the mean redundancy of a subset (Equation (16)) is the mean of the redundancies between all pairs of features within that subset. The joint mRMR criteria in (17) favours high relevance and low redundancy, which is determined by the value of $\lambda \geq 0$.

$$Rel(j) = \frac{I(Xj; Y)}{H(Y)} \quad (13)$$

$$\overline{Rel}(S) = \frac{1}{|S|} \sum_{j \in S} Rel(j) \quad (14)$$

$$Red(j, k) = \frac{I(Xj; Xk)}{\sqrt{H(Xj) \cdot H(Xk)}} \quad (15)$$

$$\overline{Red}(S) = \frac{2}{|S|(|S| - 1)} \sum_{j,k \in S} Red(j, k) \quad (16)$$

$$J_{mRMR}(S) = \overline{Rel}(S) - \lambda \cdot \overline{Red}(S) \quad (17)$$

In the proposed framework, there is only one reason for using the mRMR criterion: to score and rank features. This single ranking is then used twice: to select a high-quality initial population (Section 3.4); and to select, first, which coordinates to examine in the local search (Section 3.5). The mRMR criterion is not defined as a fitness criterion on its own and is not re-defined elsewhere in the pipeline.

3.3 Fitness Function

A wrapper-classifier based on k-fold cross validation is used to assess each candidate subset. The accuracy of a subset, defined in Equation (18), is used to classify the subset. In Equation (19) the overall fitness contains three values: the classification error (1 − Acc), the mean of the redundancy of the subset, and the relative size of the subset |S|/d. All three are costs, and the cost minimization objective is to find a subset that is as small, redundant-free, and accurate as possible:

the lower the fitness, the more accurate, redundant-free, and compact the subset will be. The weights α, β, γ > 0 are used to trade-off the quality of detection and model compactness. For all search operators below, the minimization convention is used, this will resolve the above inconsistency between the stated objective and the update equations.

$$Acc(S) = \frac{1}{k}\sum_{f=1}^{k} Score(C, D_f(S)) \quad (18)$$

$$Fit(z) = \alpha(1 - Acc(S)) + \beta Red(S) + \gamma \frac{|S|}{d} \quad (19)$$

3.4 mRMR-Guided Population Initialization

The binary population is not randomly initialized, instead the framework is seeded using the mRMR ranking of Equation (17). Only a few of the original fossae are built by randomly selecting the top few features (with some randomly small modifications to ensure diversity), and the rest are randomly created to ensure exploration. The mRMR based initialization begins the global search process from informative regions of the feature space and thus speeds up the convergence of the search process without favouring a specific subset of the population.

3.5 Global Search and Memetic Local Search

In the global phase, each fossa refines its binary position by applying the FOA operators of Section 2.2, in a compact form, denoted by the update operator Φ in Equation (20), which advances the solution towards the best solution found but keeps stochastic exploration. The continuous FOA are mapped into a binary decision using a standard transfer function and threshold.

$$z_i(t+1) = \Phi(z_i(t), z_{best}(t), t) \quad (20)$$

After A subset of elite solutions is fed into a memetic local search to exploit more intensely at the end of each global iteration. It is based on a single local-search operator: the Hamming-one neighbourhood, for which all neighbours differ from the current solution by only one feature (one bit flip) (see Equation (21)). The neighbourhood is checked starting from the most promising bit flips according to the mRMR ranking. This best neighbour is the one which most reduces the fitness (Equation (22)) and if the fitness is strictly improved (Equation (23)), the current solution is replaced. When there is no flip that further enhances the fitness, the local search stops, and a local optimum is obtained.

$$N_1(z) = \{ z \oplus e_j \mid j = 1, \ldots, d \} \quad (21)$$

$$u* = arg\,min\ Fit(u),\ u \in N_1(z) \quad (22)$$

$$if\ Fit(u*) < Fit(z)\ then\ z \leftarrow u* \quad (23)$$

This design uses one global operator (FOA) and one local operator (mRMR-ordered Hamming-one search). Gaussian hill-climbing and coordinate pattern search are mentioned in the previous version of the manuscript and are not included in the proposed framework, so they have been eliminated here for a single, consistent description. Main algorithm level pseudocode is a loop of local memetic updates on the elite set with one global FOA update per loop until the stopping criterion (maximum iterations, convergence) is reached.

The overall process is summarized in Figure 1. It starts with the complete set of features, and mRMR calculates the relevance-redundancy ranking for initialization. An elite solution is then further refined by the Hamming-one local search ordered by the mRMR, and the FOA's attacking and chasing process is performed globally. The score of each candidate is given by the minimization of fitness Equation (19). The loop continues until it reaches the stopping criterion, and the optimal feature subset is evaluated and displayed on the classification task.

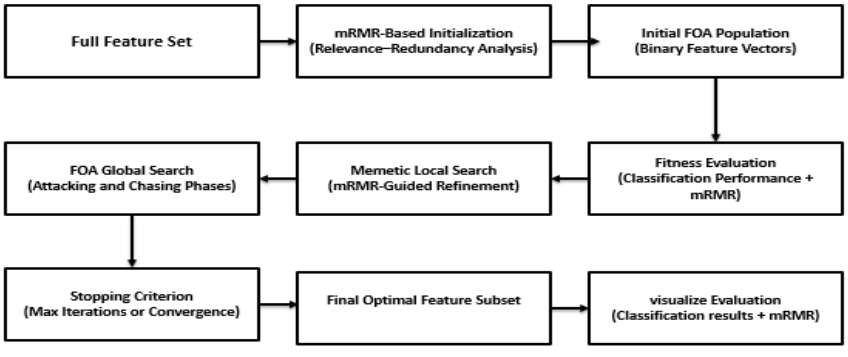

Fig. 1. Workflow of the proposed mRMR-guided Memetic Fossa Optimization Algorithm (M-FOA) for IDS feature selection.

4. EXPERIMENTAL RESULTS AND DISCUSSION

This section The performance of the proposed Memetic mRMR-FOA framework is analyzed in terms of classification performance, feature-reduction capability, fitness convergence, and computational efficiency and compared with classical evolutionary and swarm-based optimizers.

4.1 Experimental Setup Dataset

The public RT-IoT2022 intrusion-detection dataset [13] is used for all experiments, with data of real IoT/IIoT network traffic including normal traffic and multiple attack families. The continuous features were scaled into [0,1]; the categorical features were one-hot encoded; and the entire set of d = 41 features were used as the search space. The data were divided into 70% training and 30% testing sets using stratified sampling and class imbalance was addressed through class-weighted training. Classifier. All algorithms were evaluated with the same wrapper, a Random Forest (100 trees) classifier, so that any differences reported is due to the features selected, and not the classifier. The accuracy of the fitness function was evaluated using 5-fold cross validation of the training partition set. Parameter settings. All population-based methods used equal number of population size N = 30 and a maximum of number of iterations T = 100. For the fitness function, $\alpha = 0.7$, $\beta = 0.2$, and $\gamma = 0.1$; the mRMR trade-off was $\lambda = 1.0$. The elite 10% of local population were refined. GA used single point crossover (0.8) and bit flip mutation (0.02); PSO used the following control schedules: c1 = c2 = 1.5 and inertia = 0.7; GWO and FOA used their default control schedules. Runs and protocol. The algorithms were tested 30 times with different random seeds. Classification metrics are reported as mean values on a held-out test set, and the statistical analysis in Section 4.4 is done on the per-run results. The evaluation measures are Accuracy (ACC), Precision, Recall, F1-score, False Alarm Rate (FAR). Environment. All methods were coded in Python 3.11 (scikit-learn, NumPy, SciPy) and were run on a workstation equipped with an Intel Core i7 CPU and 16 GB memory and 64-bit Windows 11 OS. In Table III, the runtimes are averaged over 30 runs.

4.2 Classification Performance Analysis

The classification performance of each algorithm's feature subsets is reported in Table II, using five performance metrics: Accuracy (overall proportion of correct predictions), Precision (ability to avoid false positives), Recall (ability to detect real positives), F1-score (harmonic mean of Precision and Recall, useful in case of class imbalance), and FAR (false positive alert rate, critical for IDS reliability)..

TABLE II. IOT INTRUSION-DETECTION CLASSIFICATION RESULTS.

Algorithm	ACC (%)	Precision (%)	Recall (%)	F1 (%)	FAR (%)
GA	92.33	91.41	90.17	90.78	7.4
PSO	94.12	93.88	92.94	93.41	6.1
GWO	95.47	95.19	94.71	94.94	4.9
FOA	96.22	96.03	95.62	95.82	4.3
Memetic SU	97.11	96.92	96.88	96.90	3.3
mRMR	95.84	95.12	95.01	95.06	4.7
Memetic mRMR-FOA	98.43	98.22	98.31	98.26	1.9

The proposed Memetic mRMR-FOA obtains the best accuracy of 98.43%, which is better than all the base lines. The improvement is uniform throughout Precision (98.22%), Recall (98.31%) and F1-score (98.26%), which is indicative of good performance in detecting both majority and minority classes. The memetic extension achieves 2.21% more accuracy than the standard FOA and a FAR of 1.9% compared with 4.3%. These gains validate the importance of the joint control of information-theoretic relevance and redundancy. Memetic SU is a memetic variant of SU based on the standard FOA, but it is not explicitly capable of removing redundant features and it depends only on the symmetry of the uncertainty, outperforming standard FOA; while mRMR-FOA explicitly balances relevance and redundancy while outperforming standard FOA in terms of generalization and false alarms. In the proposed method false alarms are not costly and the low FAR is significant for intrusion detection applications.

4.3 Feature Selection and Optimization Analysis

Results of feature selection and optimization are given in Table III. The proposed Memetic mRMR-FOA selects the least number of features (13), and achieves the best (lowest) fitness value (0.038), with a higher reduction ratio (68.3%), which represents the percentage of the original features removed from the model. The standalone mRMR filter is the fastest (6.3 s), but less accurate than the hybrid methods; the proposed method achieves a competitive run time of 23.3 s that

demonstrates that the presented hybrid method is a good dimensionality reduction approach without losing too much detection quality.

TABLE III. FEATURE-SELECTION RESULTS.

Algorithm	Selected Features	Reduction (%)	Fitness	Runtime (s)
GA	23	43.9	0.089	34.2
PSO	21	48.7	0.081	29.1
GWO	19	53.6	0.073	25.4
FOA	18	56.1	0.068	22.7
Memetic SU	16	60.9	0.055	24.1
mRMR	20	51.2	0.076	6.3
Memetic mRMR-FOA	13	68.3	0.038	23.3

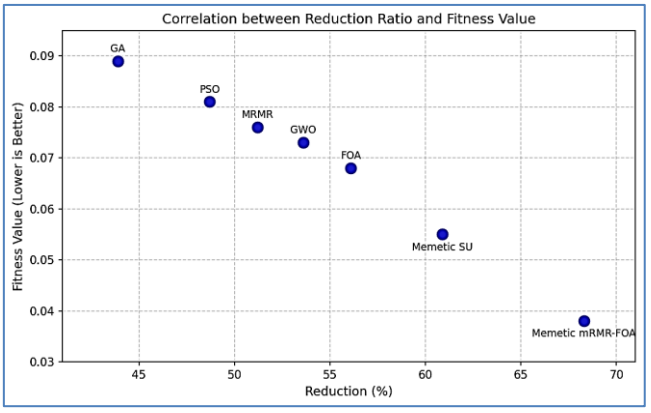

Fig. 2. Correlation between reduction ratio and fitness value (lower fitness is better).

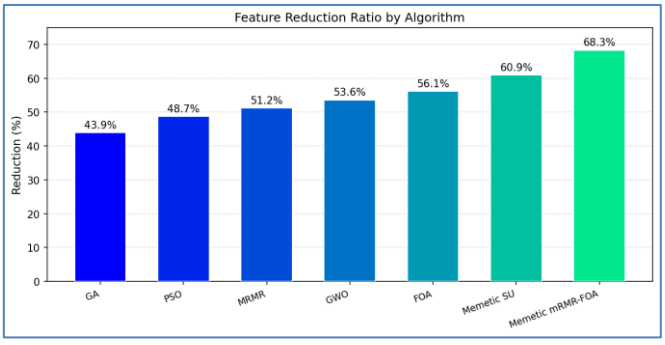

Fig. 3. Feature-reduction ratio by algorithm.

The results are illustrated in Figures 2 and 3. It is observed that there is a definite negative correlation between reduction ratio and fitness value as seen from figure 2, where higher reduction leads to lower fitness (which is a cost) while the proposed method places itself at the most favorable corner, with the highest reduction and the lowest fitness. The height of each bar in Figure 3 represents the reduction ratio direct comparison, with Memetic mRMR-FOA showing 68.3%, which is significantly higher than the classical metaheuristics, GA and PSO. The figures show that the combination of mRMR and memetic search eliminates significantly more redundancy than either of the two single filters or the traditional optimizers, thus balancing redundancy reduction with a high detection quality.

4.4 Statistical Significance Analysis

To The non-parametric Friedman test was applied on all of the 30 independent runs of the seven algorithms to conclude whether the differences between these algorithms are statistically significant or random. The Friedman test in this case gives a result in Table IV of $\chi^2 = 22.71$ with $p = 0.00039$, which is below the critical level of 0.05, so that the null hypothesis that all the performances are equal is rejected. A Nemenyi post-hoc test was used as the omnibus test only shows that differences exist, and indicates which pairwise differences are significant. The proposed Memetic mRMR is the best at the level of 0.05, and is significantly different from GA, PSO, GWO, standalone mRMR, and standard FOA, with the difference

being smaller but still in favour of the proposed method in the case of Memetic SU. The results indicate statistical evidence of the superiority of the proposed framework over the compared evolutionary and swarm based feature-selection methods.

TABLE IV. STATISTICAL SIGNIFICANCE ANALYSIS (FRIEDMAN TEST WITH NEMENYI POST-HOC).

Test	Chi-Square (χ^2)	p-value	Interpretation
Friedman test (k = 7, n = 30)	22.71	0.00039	Significant differences among algorithms
Nemenyi post-hoc (proposed vs. baselines)	—	< 0.05	Proposed method ranks first and differs significantly

5. CONCLUSIONS

This research proposed and tested a combination of feature-selection approach (Memetic mRMR-FOA) to overcome high dimensionality and security issues existing in IoT network traffic. The framework combines the filtering capacity of mRMR and the combination of global search+local search from a memetic Fossa Optimization Algorithm, which enables the inclusion of highly relevant features while eliminating redundant ones. The optimization has only one aim: to rank features, and throughout the pipeline it aims to optimise a fitness function which jointly represents detection error, redundancy, and subset size. The model proposed in this paper outperforms the standalone filters and the traditional metaheuristics like GA, PSO and GWO in terms of feature-reduction ratio on the RT-IoT2022 benchmark with 68.3%.

Most important, the minimalization of the features, to 13 attributes, did not affect classification performance, but rather enhanced detection performance, which resulted in the highest fitness value and the lowest false-alarm rate of all methods. While standalone mRMR can be significantly faster in terms of raw runtime, the proposed hybrid provides a desirable balance between speed and prediction accuracy, and is well suited for deployment on the resource constrained IoT gateways, where security is a major concern. The framework will be tested for its resistance to zero-day attacks and its performance in edge-computing and decentralized IoT systems in the future.

Conflicts of Interest

The author declare no conflicts of interest.

Funding

This research received no external funding.

Acknowledgment

The author thank the editors and anonymous reviewers for their constructive comments.

References

[1] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*. San Francisco, CA, USA: Morgan Kaufmann, 2001.

[2] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Cambridge, MA, USA: MIT Press, 2004.

[3] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” Erciyes Univ., Kayseri, Turkey, Tech. Rep. TR06, 2005.

[4] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey wolf optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014.

[5] T. Hamadneh *et al.*, “Fossa optimization algorithm: A new bio-inspired metaheuristic algorithm for engineering applications,” *Int. J. Intell. Eng. Syst.*, vol. 17, no. 5, pp. 1038–1047, 2024.

[6] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.

[7] M. Ammar, G. Russello, and B. Crispo, “Internet of Things: A survey on architectures, technologies, and applications,” *Ad Hoc Netw.*, vol. 74, pp. 1–33, 2018.

[8] Y. S. Ajaj, B. R. Al-Kaseem, and Y. Al-Dunainawi, “Improving Internet of Things cybersecurity using deep learning in software defined network environment,” in *Proc. AIP Conf.*, vol. 3211, no. 1, Art. no. 030042, 2025.

[9] Cisco, “Cisco annual internet report (2018–2023),” Cisco Systems, San Jose, CA, USA, White Paper, 2023.

[10] I. Alrashdi and S. Duraibi, “Intelligent cybersecurity management in industrial IoT systems using attribute reduction with collaborative deep learning-enabled false data injection attack detection,” *Sci. Rep.*, vol. 16, no. 1, 2026.

[11] I. Butun, S. D. Morgera, and R. Sankar, “A survey of intrusion detection systems in industrial Internet of Things,” *IEEE Access*, vol. 8, pp. 200486–200518, 2020.

[12] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, “Survey of intrusion detection systems: Techniques, datasets and challenges,” *Cybersecurity*, vol. 2, Art. no. 20, 2019.

[13] B. S. Sharmila and R. Nagapadma, “RT-IoT2022: A real-time IoT intrusion detection dataset for anomaly-based IDS,” RT-IoT2022 benchmark, 2024.
[14] T. Saranya and S. Indra Priyadharshini, “Optimized intrusion detection for IoT networks using Cauchy–Gaussian hybrid evolutionary feature selection,” *Sci. Rep.*, vol. 16, no. 1, 2026.
[15] B. Lahasan and H. Samma, “Optimized deep autoencoder model for Internet of Things intruder detection,” *IEEE Access*, vol. 10, pp. 8434–8448, 2022.
[16] J. Li, M. S. Othman, H. Chen, and L. M. Yusuf, “Optimizing IoT intrusion detection system: Feature selection versus feature extraction in machine learning,” *J. Big Data*, vol. 11, Art. no. 36, 2024.
[17] M. Almohaimeed, A. Alqahtani, and F. Alharbi, “Enhancing IoT network security using feature selection for anomaly-based intrusion detection systems,” *Appl. Sci.*, vol. 14, no. 24, Art. no. 11966, 2024.
[18] M. B. Musthafa, S. Praveen, and R. Kumar, “Optimizing IoT intrusion detection using balanced class techniques and feature selection,” *Sensors*, vol. 24, no. 13, Art. no. 4293, 2024.
[19] J. García, R. Martínez, and A. Skarmeta, “Empirical evaluation of feature selection methods for IoT intrusion detection systems,” *Comput. Netw.*, vol. 242, Art. no. 110181, 2024.
[20] X. Li, Y. Wang, and Z. Liu, “Metaheuristic-based feature selection for network intrusion detection,” *Knowl.-Based Syst.*, vol. 275, Art. no. 110668, 2023.
[21] R. P. Parouha and P. Verma, “Design and applications of an advanced hybrid meta-heuristic algorithm for optimization problems,” *Artif. Intell. Rev.*, vol. 54, no. 8, pp. 5931–6010, 2021.
[22] T. D. Nguyen, A. Alazab, A. Khraisat, and T. Jan, “Feature reduction in federated learning for intrusion detection in IoT networks,” *Cybersecurity*, vol. 9, no. 1, 2026.
[23] A. E. Hassanien, M. Kilany, and A. Darwish, “Hybrid metaheuristic algorithms for cybersecurity applications,” *Future Gener. Comput. Syst.*, vol. 146, pp. 431–446, 2023.
[24] M. A. Elaziz, D. Oliva, and S. Xiong, “Swarm-based feature selection for intrusion detection systems,” *Expert Syst. Appl.*, vol. 219, Art. no. 119676, 2023.
[25] A. Kumar, R. Singh, and P. K. Shukla, “Hybrid optimization-based feature selection for intrusion detection systems,” *IEEE Access*, vol. 12, pp. 55621–55635, 2024.
[26] Y. Zhang, Q. Chen, and L. Wang, “Metaheuristic optimization for IoT intrusion detection,” *Appl. Soft Comput.*, vol. 148, Art. no. 110850, 2024.
[27] M. A. Alzubi, O. Al-Shboul, and S. Ramadass, “Nature-inspired optimization techniques for intrusion detection systems,” *Inf. Sci.*, vol. 660, pp. 119–138, 2024.
[28] M. K. Ahmed, S. A. Aliesawi, and O. Y. Abdulhammed, “Load balancing on virtual machines using Lévy flight combined with grey wolf optimization algorithm,” *Iraqi J. Sci.*, pp. 4429–4444, 2025.
[29] E. Alba and F. Chicano, “Hybrid metaheuristics: A review and recent advances,” *Appl. Soft Comput.*, vol. 29, pp. 413–431, 2015.
[30] H. Ma, Y. Zhang, and J. Liu, “An IoT intrusion detection framework based on feature selection and data augmentation,” *Sci. Rep.*, vol. 15, Art. no. 8905, 2025.
[31] O. Achbarou, A. E. Ouahidi, and M. E. Ghmary, “Enhanced intrusion detection system using feature selection for IoT networks,” *Comput. Commun.*, vol. 220, pp. 65–77, 2025.