INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 5, May 2026 | ISSN: 2456-4184 | IJNRD.ORG

CYBERATTACKS DETECTION THROUGH IOT ENVIRONMENT VIA HYBRID INTELLIGENCE

K.Vrishin Ram
Dept of Cybersecurity
School of Engineering and Technology
Dhanalakshmi Srinivasan University
Trichy , India
vrishinram4646@gmail.com

G.Vaishnavi
Dept of Cybersecurity
School of Engineering and Technology
Dhanalakshmi Srinivasan University
Trichy , India
vaishnaviganesan2006@gmail.com

Dr.V.Ravindra Krishna Chandar
Associate Professor
Dept of Cybersecurity
School of Engineering and Technology
Dhanalakshmi Srinivasan University
Trichy , India
vrkchandar@gmail.com

ABSRACT:

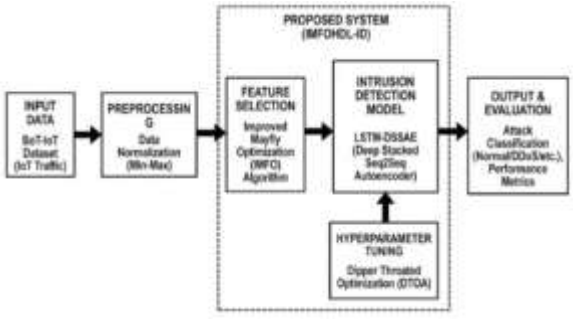
The Internet of Things (IoT) ecosystem has experienced exponential growth because traditional Intrusion Detection Systems (IDS) cannot handle the current data volume and complexity shown in modern networks. This paper presents IMFOHDL-ID as a new framework which combines bio-inspired evolutionary logic with hybrid deep learning to defend against both existing and new cyber threats. Our architecture uses Improved Mayfly Optimization (IMFO) to filter high-dimensional network traffic which allows us to identify mission-critical features while reducing the processing requirements. The Long Short-Term Memory-based Deep Stacked Sequence-to-Sequence Autoencoder (LSTM-DSSAE) uses these refined inputs to detect complex patterns of temporal attack signatures. We use the Dipper Throated Optimization Algorithm (DTOA) to conduct autonomous hyperparameter tuning. The experimental results demonstrate that the hybrid synergy provides both improved detection accuracy and throughput which creates a scalable security solution appropriate for IoT environments with limited resources.

Keywords: Internet of Things (IoT),Intrusion Detection System (IDS),Deep Learning,Meta-Heuristic Optimization, Feature Selection,Long Short-Term Memory (LSTM).

INTRODUCTION:

The Internet of Things is currently a significant research topic of focus. An international electronic network.What we refer to as IoT is devices. To enable the normal routine operations to be self-realized in life circumstances,that that is its essence of soul - no man required. Hackers attack as additional gadgets become part of the system intentionally. They use them much more frequently . Alongside this growth comes increased worry: to what degree are these related tools safe. When attacked? The most remarkable thing is that it is all the more important when dealing with sensitive data. Extracted out of IoT devices—detection of threats requires explicit understanding. These systems frequently fail to due to the presence of flaws. Crumble when pressure is exerted on them due to digital intrusions. Other professionals combined the type of attack, areas of weakness, and,protection issues are merely to facilitate easy fixes in future. One approach is take layer-based design. Where risks were predominantly manifest at the body-levels where protection was not in place.

Not all the gadgets speak securely to the internet. Difficulties arise as the setup becomes too tangled or difficult to control and slips away. The data circulates without any clear guidelines, occasionally finding their way into wobbly positions. Weak passwords open doors. Entry points of networks remain open. Greener hacks have now applied learning machines to the detriment of the hack's systems. The swarms of stolen devices generate new problems . Scholars have observed loopholes in the manner in which these are. things hold together. Their general argument has a tendency to fall on the reuse of old-style networks, where they are not f it well. The addition of more small gadgets onto the web adds strain, in particular the low-power endpoints. They can hardly compute and so are excellent targets .

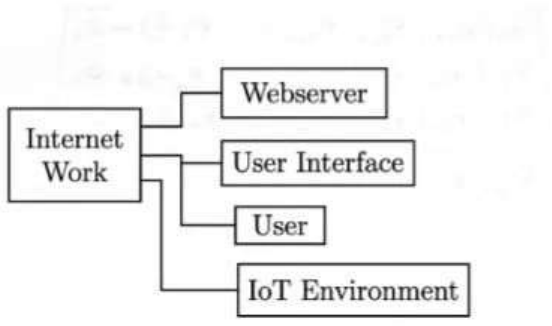


System Architecture

Although it has certain safety precautions in place, location, and IoT systems are powerless against most of the attacks. To their huge attack surface . Therefore, it is essential to plan defense devices to detect attacks. One more

defense should be offered to protect the IoT systems not only against cyberattacks but also against physical attacks. An Intrusion Detection System (IDS) is one of the effective models that will fulfill this purpose. Numerous surveys have attempted to identify the concept of machine learning (ML)-based IDS to maintain security in their presence . Many studies have been done on IDSs for cyber-physical systems, wireless sensor networks, cloud-based IoT systems, and mobile ad hoc networks (MANET). In addition, traditional IDS schemes are not as efficient or even. poor to the security of IoT networks because of their atypical characteristics, which are uncovered above, primarily rich, heterogeneity, limited energy, restricted bandwidth capacity, and overall connectivity. Deep Learning (DL) and ML systems have currently gained huge popularity in effective usage for the identification of threats in the system, which has IoT devices. The answer to this is that the ML and DL-based methods are capable of seizure, benign, as well as irregular behavior in an IoT atmosphere. IoT: To be educated about ordinary patterns. systems, as well as examined, can be taken. Any irregularity in these accustomed learned designs is employed to determine abnormal behavior. Moreover, ML/DL approaches have been verified for predicting novel attacks . Thus, the outputs of ML/DL methods are. Strong safety measures to put the safety of IoT networks and devices together.

This is a sophisticated Mayfly algorithm, along with deep digital undertones of that place. Training the ability to identify attacks in Internet of Things networks. Immediately in the beginning, raw data is scaled into homogeneous ranges prior to anything else occurring. Rather than haphazard selection, there are designated characteristics that emerge due to having a better swarm-inspired search. Pattern lurks: the sequences lurk in patterns. Traverse memory-endowed neural strata in deep layers within. Lastly, as though he were tuning or f ine-tuning an instrument. Instrument by ear, an optimizer that imitates nature adapts the key settings to optimum performance. The gains in performance are very evident when simulations are carried out with the IMFOHDL-ID model. These tests, conducted in phases, show better results than more aged procedures. In the meantime, old methods are in the rear. This one, behind, gives a further impulse—with no additional weight, accuracy increases. With each iteration of analysis comes an addition. Proof, not promises. The benefits never come as a gift but in a shape. Where others plateau, shifts. Happen here. Repeatability between trials gives more credence to findings.



Internet access model

II. RELATED WORKS

A new deep learning methodology of IoT security is presented. It is constructed quickly and is built in four layers. Connected system to identify malicious data streams towards linked devices. There are no confining rules of communication. This construction- it facilitates the obstacles to usage rather. Based on Saba et al. again, an avenue comes up: a CNN model that will be used to identify odd behavior in IoT streams. As the Internet of Things expands, so does its power, allowing. monitoring the entire network activity. This new system will be able to identify probable intrusions as well as suspicious data. Flow patterns. Rather than more traditional models, the method employs LSTM to identify cyberattacks in IoT.softwarecontrolled networks. Instead of restrictive tests, the study considers the extent to which machine learning and deep learning work on two datasets that revolve around SDN and IoT. Another model constructed based on LSTM is used to detect the different kinds of network threats that strike smart devices simultaneously

Rather than the conventional methods, Ramaiah and colleagues propose new identifiers to identify danger. Activities within smart environments. To improve the performance, their intrusion system connects variable selection using correlation tools as well as a random forest method. Important characteristics are f iltered out first, then a neural method that has been fine-tuned to attacks is used. Next comes shallow neural nets, paired, and a refined detection mechanism that will help in detecting stealthy attacks. Along a different route, the group of Ravi.Construct a complete pipeline with deep learning, but instead of analyzing features, repetitive patterns are analyzed over time. Hidden layers infirst get stripped away in recurrent models. Once more, the methodology is based on the use of kernel-based PCA. Picking traits. Ultimately, the combination of major aspects of the looping patterns—detection follows—is what is left. with the wisdom of grouped classification.

A new IDS approach, in which the DL and optimization techniques are clustered, is created.Primary, a feature model extraction, which is founded on CNNs, is proposed. Next, a novel FS process is utilized, relying on an adapted version of Growth Optimizer (GO), called MGO.In order to enhance the searching process of GO, it is also provided with the Whale Optimization Algorithm (WOA). Wanget al. built a DL-powered bidirectional LSTM (BiLSTM) trivial IoT intrusion detection approach. TheBoth the BiLSTMs and deep neural networks (DNNs) approaches are combined when it comes to extracting the features. The Incremental Principal Component Analysis (IPCA) model is employed to decrease feature dimensionality. In addition to that, dynamic quantization is used. In [18], the various authors came up with a new LSTM-based intrusion detection approach with a Dynamic Access Control (DAC) model to detect and defend against intrusion. The DAC approach defends further intrusions from a similar source by blocking it for periods related with thenumber of intrusions.

To a genetic algorithm approach for feature selection, aiming to strengthen intrusion detection within IoT systems under pressure from botnet threats. On another path,the deep learning into a layered method—three stages deep—to catch harmful actions swiftly inside smart device networks.

As of today, identifying cyberattacks in the IoT using deep learning continues to have a low success rate in terms of identifying the right features and making efficient adjustments. Although the models have been enhanced, many studies are excessive. Do not take time to tune carefully and to pick the relevant data —this compromises your speed and accuracy. One wayforward Smart techniques of searching, such as particle swarm, chaotic mayfly behavior, or genetic. algorithms that come into the limelight. These methods investigate possibilities extensively, discovering robustness. Systems that use the limited resources to the full extent without any extra features like parameter setups and feature sets, without the need to drain resources. Harder with tight power and processing. Exploring the ways deep learning gets intertwined with smart search. Methods continue to attract attention, but few have delved deep—particularly in the area of identifying cyber threats. threats in IoT devices. Due to this gap, the feature selection method that is brought in by the current work

is guided by IMFO together with hyperparameter tuning informed by DTOA, with the aim of enhancing defenses across. internet-connected systems.

III. THE PROPOSED MODEL

This is where we can see the development of automatic intrusion detection systems based on the IMFOHDL-ID setup within IoT spaces. The IMFOHDL-ID strategy is effective and designed to identify breaches and increase protection. By one after another. First of all, scaling the data, then selecting features through IMFO-driven selection sneaks in. Subsequently, pattern spotting comes in with stacked LSTM layers that are shaped like that autoencoders. An optimizer called DTOA then takes over and sets the tuning fine to the optimum signalto-noise ratio.

A. *DATA NORMALIZATION*

Immediately, the model works with data by normalising it and then proceeding with other actions. Because IoT systems gather data about numerous devices and sensors, agreeing on values becomes.necessary - here minmax scaling fits in. Rather than crude figures floating about randomly, they are readjusted to a certain range between zero and one. With everything on even footing, odd patterns are easier to spot, and the detection process is easier. Rather than fighting with mutually incompatible scales, the system compares mutually compatible inputs. Odd shifts are very easy to note when all signals use the same numerical system. Security alerts gain accuracy since distortions due to differences in scale are eliminated. This arrangement enhances the effectiveness. The layer of defence is one that supports in the long run. Scaling is not an aesthetic procedure, but it creates the degree of reliability that threats are caught early.

B. *FEATURE SELECTION USING IMFO ALGORITHM*

In this phase, the proposed architecture employs the Improved Mayfly Optimization (IMFO) algorithm to identify and extract the most relevant feature subsets. The fundamental Mayfly Optimization (MFO) is a bioinspired, hybrid swarm intelligence technique modeled after the social, foraging, and mating behaviors of mayflies. The algorithm simulates the survival of the fittest, where superior offspring endure to subsequent generations.

During the initialization stage, two distinct populations are randomly generated to represent male and female mayflies. Each mayfly acts as a candidate solution within a multidimensional search space, represented by a position vector.

Male Mayfly Position and Velocity Updates

A male mayfly adjusts its flight trajectory based on its personal historical best and the swarm's global best position. Let denote the current position of the i-th male candidate solution at iteration step d. The position is updated by adding the velocity vector:

$$x_i^{d+1} = x^d_i + v_i^{d+1}$$

To compute the updated velocity v_{ij}^{d+1} at the next iteration $d+1$, the algorithm factors in the attraction to both the personal best $(pbest_i)$ and the global best . $(gbest)$ With the updated parameter notation, the velocity is formulated as:

$$v_{ij}^{d+1} = v^d_{ij} + a_1 e^{-\beta r_p^2}(pbest_{ij} - x^d_{ij}) + a_2 e^{-\beta r_g^2}(gbest_j - x^d_{ij})$$

Where:

- d represents the current discrete iteration step.
- a_1 and a_2 are positive constants dictating attraction strength.
- α serves as the visibility coefficient, controlling the exponential decay of attraction over distance.
- r_p and r_g denote the Euclidean distances from the current position to the $pbest$ and $gbest$ positions, respectively.

If a male mayfly achieves an optimal state, it performs a characteristic "nuptial dance" by introducing a randomized speed fluctuation to its movement:

$$v^{d+1}_{ij} = v^d_{ij} + k \times r$$

Here, k is a movement constant (substituted from the original d to avoid index conflicts), and r is a random continuous variable ranging between $[-1, 1]$

Female Mayfly Movement Dynamics

Unlike their male counterparts, female mayflies do not aggregate toward a global best position. Instead, they fly toward the males for mating. Let y_i^d represent the position of the i-th female at iteration d. Its position is similarly updated via its velocity:

$$y_i^{d+1} = y^d_i + v_i^{d+1}$$

The female's velocity is determined by her distance to the corresponding male mayfly (r_{mf}). The update rule is mathematically expressed as:

$$v_{ij}^{d+1} = \begin{cases} v_{ij}^d + a_2 e^{-\alpha r_{mf}^2}(x_{ij}^d - y_{ij}^d), & \text{if } f(y_i) > f(x_i) \\ v_{ij}^d + fl \times r, & \text{if } f(y_i) \le f(x_i) \end{cases}$$

If the female is not attracted to the male (based on the fitness function evaluation f), a random walk is initiated governed by the coefficient fl.

Mating and Crossover

The mating phase is executed using a crossover operator. A selected pair of male and female parents breed to generate two new offspring, effectively sharing genetic information:

$$Offspring_1 = L \times Male + (1-L) \times Female$$

$$Offspring_2 = L \times Female + (1-L) \times Male$$

Where L is an arbitrary random weight. Initially, the velocities of these new offspring are set to zero.

Chaotic Mapping Enhancement

Standard optimization problems often rely on pure randomization for initial agent placement, which can lead to slow convergence. The IMFO algorithm enhances this initial population generation by incorporating chaotic maps. Specifically, a logistic chaotic map is utilized due to its superior computational efficiency and its ability to rapidly perform local searches:

$$y_{i+1}=4\times y_i\times(1-y_i)$$

By utilizing this logistic mapping instead of standard random vectors, the initial population distribution is significantly improved, preventing the model from stagnating in local optima early in the search process.

Fitness Function Evaluation

For the specific task of feature selection, the objective is to strike a balance between maximizing classification accuracy and minimizing the dimensionality of the selected feature subset. The fitness of a given subset is calculated using the following equation:

$$Fitness=w_1\ \gamma R(D)+w_2\ (\frac{R}{C})$$

Where:

- $\gamma R(D)$ denotes the classification error rate.
- R is the cardinality (number of features) in the selected subset.
- C is the total number of features available in the original dataset.
- w_1 and w_2 are adjustable weight parameters (representing the original α and β)that govern the trade-off between the classifier's performance quality and the subset's minimal length, satisfying $w_2=1-w_1$

C. *INTRUSION DETECTION USING HDL MODEL*

For the intrusion detection phase, the proposed framework implements a Hybrid Deep Learning (HDL) approach utilizing a Long Short-Term Memory-based Deep Stacked Sequence-to-Sequence Autoencoder (LSTM-DSSAE). An LSTM-based Autoencoder is highly proficient at processing sequential information, making it exceptionally well-suited for analyzing time-series data and network traffic flows within an IoT environment.

The architecture consists of an encoder-decoder structure formed by stacking multiple LSTM layers. LSTMs are specialized recurrent neural networks (RNNs) equipped with memory cells, enabling them to retain contextual information over long sequences and effectively capture complex temporal dependencies. This memory retention is critical when encoding and reconstructing data sequences, a capability standard autoencoders lack.

During the encoding phase, the network compresses the input data sequence into a concentrated, low-dimensional representation known as the hidden space or bottleneck. Let D represent the specific step or dimension in the sequence.

The sequential progression and state update of the input data vector x_i at step D is mathematically defined as:

$$x_i^{D+1}=x_i^D+v_i^{D+1}$$

where the input variable is bounded such that $x_i^D \in (x_{min}, x_{max})$

Following the encoding process, the LSTM decoder utilizes this compressed contextual representation to reconstruct the original data sequence. The primary objective of the LSTM-DSSAE model is to minimize the discrepancy between the original input sequence and the network's reconstructed output.

By forcing the autoencoder to recreate the input data, the model learns the fundamental semantic characteristics of normal, benign IoT network traffic. To quantify the model's performance, the reconstruction error is calculated using the maximum Mean Squared Error (MSE) formula. Assuming the original input sequence is $x(D)$ and the reconstructed output generated by the decoder is $y(D)$, the loss function is expressed as:

$$Loss=\sum(x^{(D)}-y^{(D)})2$$

By iteratively minimizing this reconstruction loss during the training phase, the LSTM-DSSAE learns to accurately capture the relevant features of standard traffic. Consequently, during the testing phase, any significant deviation from these learned patterns results in a high reconstruction loss, allowing the system to flag the anomalous behavior as a potential cyberattack or intrusion.

D. *HYPERPARAMETER TUNING USING DTOA*

To extract maximum performance from the HDL approach, the optimal selection of hyperparameters is critical. This framework employs the Dipper-Throated Optimization Algorithm (DTOA), a modern metaheuristic inspired by the foraging and cooperative swimming behaviors of dipper birds. The algorithm effectively balances search space exploration and local exploitation to pinpoint the most effective hyperparameter configurations.

In the DTOA process, a flock of birds acts as the population of candidate solutions navigating the search space for food (representing the optimal solution). The multidimensional hyperparameter space is modeled using two primary matrices: the position matrix (P) and the velocity matrix (V).

For a population of m birds operating across variable dimensions (denoted by D 1 ,D 2 ,…,d), the collective positions and speeds are defined mathematically as:

$$P=\begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & \cdots & P_{1,D_1} \\ P_{2,1} & P_{2,2} & P_{2,3} & \cdots & P_{2,D_2} \\ P_{3,1} & P_{3,2} & P_{3,3} & \cdots & P_{3,D_3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_{m,1} & P_{m,2} & P_{m,3} & \cdots & P_{m,d} \end{bmatrix}$$

$$V = \begin{bmatrix} V_{1,1} & V_{1,2} & V_{1,3} & \dots & V_{1,D_1} \\ V_{2,1} & V_{2,2} & V_{2,3} & \dots & V_{2,D_2} \\ V_{3,1} & V_{3,2} & V_{3,3} & \dots & V_{3,D_3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ V_{m,1} & V_{m,2} & V_{m,3} & \dots & V_{m,d} \end{bmatrix}$$

To evaluate the quality of each candidate solution within the flock, a fitness function is applied to the position vectors. The fitness evaluation matrix f incorporates a dimensional or step penalty/addition (D i) to account for the varying sequence lengths in the feature space:

$$f = \begin{bmatrix} f_1(P_{1,1}, P_{1,2}, P_{1,3}, \dots, P_{1,d}) + D_1 \\ f_2(P_{2,1}, P_{2,2}, P_{2,3}, \dots, P_{2,d}) + D_2 \\ f_3(P_{3,1}, P_{3,2}, P_{3,3}, \dots, P_{3,d}) + D_3 \\ \vdots \\ f_m(P_{m,1}, P_{m,2}, P_{m,3}, \dots, P_{m,d}) \end{bmatrix}$$

The fitness selection process heavily dictates the convergence quality of the DTOA. For this intrusion detection framework, the optimizer is strictly driven to maximize the precision of the classification. The objective fitness function is established as:

$$Fitness = max\ (P)$$

where P represents Precision, calculated using True Positives (TP) and False Positives (FP):

$$P = \frac{TP}{TP + FP}$$

To further validate the selected hyperparameter sets and ensure the model does not disproportionately favor precision at the cost of missed attacks, the F-score metric is tracked. The F-score integrates both False Positives (FP) and False Negatives (FN) into a single harmonic mean:

$$F-score = \frac{(2(TP))}{(2(TP) + FP + FN)}$$

By continuously updating the position matrices relative to the best-performing "mother bird" (the optimal hyperparameter configuration found so far) and maximizing the precision-based fitness function, the DTOA ensures the LSTM-DSSAE model operates at peak efficiency.

IV. PERFORMANCE VALIDATION

A. *DATASET USED*

In this study, intrusion detection outcomes of the IMFOHDLID system can be examined withthe BoT-IoT database [24], as defined in Table 1.

B. *PERFORMANCE MEASURES*

To comprehensively evaluate the intrusion detection capabilities of the proposed IMFOHDL-ID system, a robust set of performance metrics is employed. These metrics provide quantitative assessments of the model's classification effectiveness across multiple dimensions, enabling thorough analysis of its detection accuracy and reliability.

Precision quantifies the proportion of correctly identified intrusion instances among all samples classified as intrusions by the model:

$$Precision = \frac{TP}{TP + FP}$$

This metric is particularly valuable in intrusion detection scenarios where minimizing false alarms is critical, as it measures how many of the predicted attacks are genuine threats.

Recall (also known as sensitivity or true positive rate) measures the model's ability to correctly identify all actual intrusion instances within the dataset:

$$Recall = \frac{TP}{TP + FN}$$

High recall is essential in cybersecurity applications, as it indicates the system's effectiveness in detecting actual attacks without missing critical security breaches.

Accuracy provides an overall measure of the model's correctness by calculating the ratio of all correct predictions to the total number of predictions:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

This metric offers a comprehensive view of the model's performance across both intrusion and normal traffic classification.

F-score represents the harmonic mean of precision and recall, providing a balanced evaluation metric that considers both false positives and false negatives:

$$F-score=\frac{2\,TP}{2\,TP+FP+FN}$$

The F-score is particularly useful when seeking an optimal balance between precision and recall, as it penalizes extreme values in either metric.

Mathew Correlation Coefficient (MCC) is a sophisticated metric that considers all four confusion matrix categories, providing a balanced assessment even with imbalanced class distributions:

$$MCC=\frac{TN\times TP-FN\times FP}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

MCC values range from -1 to +1, where +1 indicates perfect prediction, 0 represents random prediction, and -1 denotes complete disagreement between predictions and actual values. This metric is particularly valuable for intrusion detection systems where class imbalance is common.

G-measure computes the geometric mean of precision and recall, offering an alternative balanced metric:

$$G-measure=\sqrt{Precision\cdot Recall}$$

Unlike the F-score's arithmetic mean, the geometric mean in G-measure is more sensitive to extreme values, making it useful for identifying models with significant performance disparities between precision and recall.

In these formulas, TP (True Positives) represents correctly detected intrusions, TN (True Negatives) indicates correctly identified normal traffic, FP (False Positives) denotes normal traffic incorrectly flagged as intrusions, and FN (False Negatives) represents missed intrusions that were incorrectly classified as normal traffic. Together, these metrics provide a multifaceted evaluation framework for assessing the IMFOHDL-ID system's intrusion detection performance.

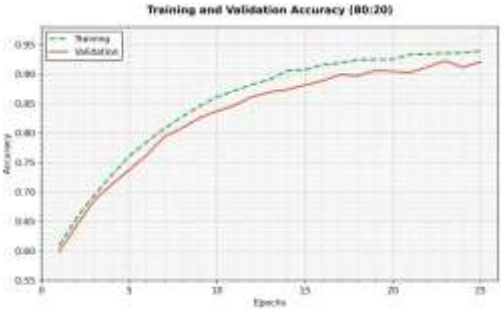
C. RESULTS AND DISCUSSION

The result shows the performance of the IMFOHDL-ID structure using the test data. The confusion matrix obtained through the system gave an 80:20 split of TRAPS to TESPS. In that background, a classification into 9 categories is correct. Meanwhile, indicates the precision-recall behaviour of the same model. Every category scores well, exhibiting a uniform degree of reliability in detection. We can observe the ROC curve of the IMFOHDL-ID method. This indicates the level of performance of the model in various categories. The greater the ROC, the better the results. Even when the type of classes changes, performance remains the same.

Accuracy curve of IMFOHDL-ID model with 80:20 of TRAPS/TESPS (New Run).

Validation of the IMFOHDL-ID method's intrusion detection ability becomes easily understandable with an 80:20 TRAPS to TESPS split. All the classes demonstrate the best performance due to the IMFOHDL-ID approach. The average performance of 80 per cent of TRAPS reached 98.31 per cent accuracy and precision<|human|>The performance averages of 80 per cent of TRAPS were 98.31 per cent accuracy and 98.31 per cent precision. The performance settles at 92.09%, recall at 91.75%. The F-score value is close to 91.90%, and the MCC is 90.96; the G-measure is less than 90.96%. just above 91.91%. A change to 20% of TESPS yields slightly worse results - accuracy decreases a bit to.98.16%. Here MCC is 90.06 and Gmeasure is just behind at 91.09. Accuracy plots: Looking at the performance of IMFOHDL-ID on an 80:20 split of TRAPS to TESPS, accuracy plots emerge for each set. The most remarkable thing is that they show patterns in the data very clearly

Patterns begin to reveal themselves more clearly with the increase in the number of epochs. Improvement in both training and test accuracy will be seen with time. The model begins to tell the differences better between what it has seen and what is new. An increase in performance implies that it is learning how to manage diverse inputs. With long cycles, there are sharper results on both sides.



Accuracy curve of IMFOHDL-ID model with 80:20 of TRAPS/TESPS (New Run).

A glance at 7 shows how the IMFOHDL-ID approach performs in the presence of TRAPS and TESPS at an 80:20 to TRA setup. The system adjusts as the TRA loss decreases as training progresses, itself more specifically with time in both datasets. The form of that downward direction in error is the form of progress.

The most notable thing is how the model aligns with TRAdata. Though surprising, both TRA and TES lossdrop continuously, indicating that it in fact picks up patterns in both sets. And since then, differences. Where fresh and estimated TRA tags start to decrease, directed by its corrective action.

The test results of the IMFOHDL-ID method seem to be well laid out. The confusion matrix is revealed under a 70:30 division of training to testing points of data.Recognition in nine different categories becomes accurate due to the manner in which the model classifies the inputs.This one emphasizes accuracy and the recall behavior of each group. Each category has high scores, with a steady reliability in detection tasks -showing how the IMFOHDL-ID method performs in ROC terms. Both classes demonstrate good results, characterized by a high ROC number throughout the board.

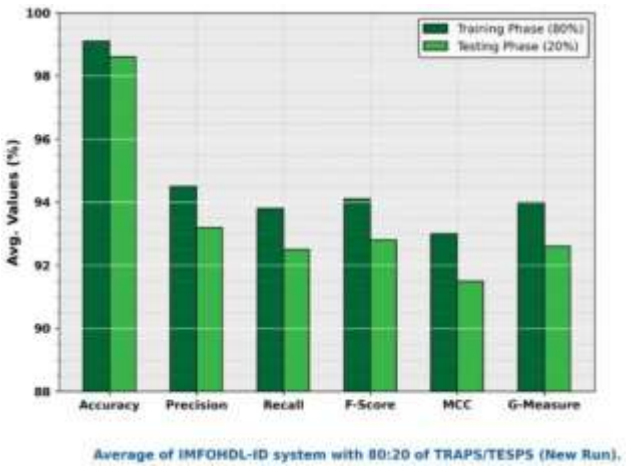
The IMFOHDL-ID method fares when put to the test.with a split of training and testing samples of 70:30. Findings do indicate better results.with the use of this model, across nine categories. IMFOHDL-ID converts to less than 70 percent of the data when trained on 70 percent of the data.increased average values in accuracy, precision, recall, F-score, and MCC.

Starting off, the numbers sit at 97.56%, 88.94%, 88.49%, 88.67%, 87.32%, and 88.69% for accuracy,precision, recall, Fscore, MCC, and Gmeasure - their place and their place. Then follows a twist: when TESPS<|human|>This is then followed by a twist: when TESPShits 30% results are slightly lower than IMFOHDL-ID. The accuracy decreases a notch lower to 97.52%. Precision holdssteady again at 88.94%. Recall slips only slightly behind previous mark to 88.41%. The Fscore lands on 88.62%. MCCdrops marginally to 87.26%. Lastly, Gmeasure finds its rest at 88.65% which completes that series.

Considering the goodness of IMFOHDL-ID model when using a 70:30 split of training and testing data,both sets have accuracy graphs. Out of these plots you can observe how the method.grows better with time - its hold becomes more acute. As the number of training rounds increases, the lines go up the building.uniformly, no sharp turns, but merely gradual accretions. That increase does not consist of mere noise - it is pattern.capturing at learning stages. Every progressive step gives us an idea that the recognition abilities on new samples will be improved.too. Having both curves move in unison implies that there is balance - neither pulling away and leaving the other behind. Patterns are retained as a result of repeated exposure as smarter responses are developed gradually as a result of this repetitive exposure. Progress unfolds it is obvious when numbers creep upwards without hitting the limit at the beginning. The model does not learn through guessing but by progressive perfection in a series of successions. Findings remain the same - the type that are induced by structure,not luck.

The IMFOHDL-ID method fares when TRAPS and TESPS lose 70:30.during the TRAprocedure. TRA loss decreases with every succeeding epoch, since the model adapts its.weights to ensure that the predictions are more accurate in both TRA and TES sets. The closeness with which the curve descends.informs us of how well the model fits the TRA data. The reality that both TRA and TES losses are settled at a low level.implicates good pattern recognition in such datasets. With time, the system adjusts to itself, such that the TRA as predicted is achieved.Outputs remain near the real ones.

Next is a comparison of how IMFOHDLID would fare against the newer techniques in detecting intrusions -details , pulled from sources [16], [19], [20]. Low performance with DT.and GWO, evidently lagging behind. On the other side, MGO and LD continue further with more vigorous.numbers. Bagging ensembles, KNN, and SVM can handle decent marks as well. What stands out? IMFOHDL-IDclimbs highest, getting maximum scores: accuracy lands at 98.31, precision at 92.09, recall hits 91.75, F-score rests on 91.90%, while MCC holds firm at 90.96%. That peak matters.Lastly, comparison computational time (CT) analysis of IMFOHDL-ID method to existing methods.Procedures on the intrusion detection in techniques.



Average of IMFOHDL-ID system with 80:20 of TRAPS/TESPS (New Run).

V. CONCLUSION

This study presents the IMFOHDL-ID method, which is aimed at detecting threats and enhancing security within.IoT systems. The structure consists of scaling data, feature selection using improved.algorithm, detecting anomalies with a deep learning configuration, and optimizing settings with another smart.process. At the very beginning it converts raw information into a standardized format and then carries on working with it. On top of that, key attributes are selected by an adapted optimization strategy designed to be much more relevant.tracking. IMFOHDL-ID employed LSTM-dSSAE in the detection of intrusions. Eventually, DTOA came to the rescue.Select the most appropriate settings of that model. To demonstrate the effectiveness of it, a series of simulations followed each other. This new arrangement did better by far when contrasted with older methods.Future work could consider cleaning up of the odd data points in order to make the system even more efficient.

In the future, one of the directions is to check the effectiveness of the model when subjected to various IoT networks -the ones that are different in type and size. It can be run in a variety of setups that may demonstrate how it copes with all that.to intelligent homes to serious industrial setups. The difference here is to see whether the model retains or not.Good performances and detecting risks in changing situations and asymmetrical flows of data. Another angleis based on broader experiments: they may reveal problems which are related to definite environment, which can provide hints to.hone the system to be able to withstand live environments.

[1] L. Reynaud and F. T. B. de Oliveira, ''Federated learning for IoT cybersecurity: Concepts, architecture, and applications,'' IEEE Internet Things J., vol. 9, no. 18, pp. 16813–16827, Sep. 2022.

[2] J. Wang, L. Zhao, and J. Huang, ''A comprehensive review of artificial intelligence applications in IoT cybersecurity,'' IEEE Access, vol. 10, pp. 93678–93695, 2022.

[3] A. O. Omolara et al., ''The Internet of Things security: A survey encompassing unexplored areas and new insights,'' Comput. Secur., vol. 112, Jan. 2022, Art. no. 102494.

[4] Z. Abbas and S. Myeong, ''Forecasting ML application in Industrial Cloud focusing on privacy and trust issues,'' Appl. Sci., vol. 13, no. 8, p. 4725, Apr. 2023.

[5] M. M. Alani et al., ''DeepIIoT: An explainable deep learning-based intrusion detection system for industrial IoT,'' Comput. Secur., vol. 132, Sep. 2023, Art. no. 103328.

[6] S. N. Kumar et al., ''IoT-based intrusion detection system using new hybrid deep learning algorithm,'' Electronics, vol. 13, no. 6, p. 1053, Mar. 2024.

[7] B. S. Sharmila and R. Nagapadma, ''Quantized autoencoder (QAE) intrusion detection system for anomaly detection in resource-constrained IoT devices using RT-IoT2022 dataset,'' Cybersecurity, vol. 6, no. 1, p. 28, Dec. 2023.

[8] M. U. Tariq et al., ''A novel deep learning-based intrusion detection system for IoT networks,'' Computers, vol. 12, no. 2, p. 34, Feb. 2023.

[9] R. M. A. Ute, R. B. A. M. Ali, and S. M. Darwish, ''Feature selection using hybrid ant harris algorithm for IoT network security enhancement,'' Int. J. Comput. Netw. Commun., vol. 17, no. 1, pp. 125–140, Jan. 2024.

[10] A. K. M. N. Islam et al., ''Feature-optimized intrusion detection based on a hybrid spiking neural network for the Internet of Things,'' J. Adv. Inf. Technol., vol. 15, no. 4, pp. 848–858, 2024.

[11] S. A. Alghawli et al., ''An intelligent feature selection method for IoT botnet attack detection based on meta-heuristic algorithms,'' IEEE Access, vol. 11, pp. 4567–4580, 2023.

[12] E. C. P. Neto et al., ''CICIoT2023 Dataset,'' Canadian Institute for Cybersecurity, 2023. [Online]. Available: https://www.unb.ca/cic/datasets/iotdataset-2023.html

[13] B. S. Sharmila and R. Nagapadma, ''RT-IoT2022 Dataset,'' UCI Machine Learning Repository, Jan. 2024. [Online]. Available: https://archive.ics.uci.edu/dataset/942/rt-iot2022