An Efficient Feature Selection Technique for Intelligent IDS using Metaheuristics Consensus Ensemble Aggregation.

S.Vijayalakshmi[1] and Dr.V.Prasanna Venkatesan[2]

[1]Research Scholar, Department of Banking Technology, Pondicherry University, Pondicherry, India.
[2]Professor, Department of Banking Technology, Pondicherry University, Pondicherry, India

ABSTRACT

The mammoth proliferation of digital data across diverse computing platforms, devices and social web portals have dictated the need for efficient pruning and trimming of data to a concise representation. This concise format facilitates the security detection engine to be a winner in any scenario amidst avalanche of impending security threats. This mandates the effective institutionalization of security infrastructure viz. Intrusion Detection System (IDS), Firewalls and Application proxy. Dimensionality Reduction techniques viz. Feature Selection and Feature Extraction mechanism echoes the same sentiment. The inherent bias, variance and weakness exhibited in a single feature selection technique is annulled with the ensemble of feature selection techniques empowered with collective crowd intelligence. This problem effectively translates into optimization problem that recommends the deployment of metaheuristics search algorithm to effectively tackle this NP hard problem with the exponential growth of the problem space and complexity. This proposed model leverages the application of MH algorithms as powerful Ensemble Attribute Aggregator/Fusion agent of the feature selector ensemble outcomes to generate optimal feature combinations that amplifies the classification performance and as well the timing efficiency of IDS. This model encourages the adoption of Ensemble of Filter Feature selectors as it is not tied to any classification algorithm and fast to generate intermediate optimized feature subsets that is fed to the IDS model. This model promises to deliver quick real time decisions with high TPR and minimal detection latency. Using MH as aggregators helps to avoid myopic decisions engineered by Filter Ensemble. The data deluge property in Ultra High Dimensional Dataset (UHDD) is effectively countered with MH algorithms that optimizes the feature aggregation to address the scalability and stability issues with increased execution speed up and convergence rate. Various evaluation metrics have been proposed to comprehend the efficacy of this model with NSL-KDD dataset comprising different attacks.

KEYWORDS

Feature Selection, IDS, Metaheuristics, Genetic Algorithm, Meta-Aggregator.

1. INTRODUCTION

As the quote goes, “We are drowning in Information Flood (load) but starving for Knowledge” is the apt situation existing in current scenario where big data era herald’s transformative shifts in the way the data is analyzed and interpreted to derive meaningful insights and perceptions. Today’s enterprise has to confront the challenges posed by massive generation of data from multi modal sources viz. transaction logs, IoT sensors, social media, Customer Interaction and network traffic [1]. Curse of dimensionality/Dimensionality debacle poses a significant challenge to the efficient data processing and assimilation. This unfolds heavy information load on the inference model/engine thereby staggering its performance. This may lead to unnecessary confusion, delayed decisions, computation and storage overhead in the data pre-processing layer. Corporates and enterprises need to be on constant vigil against adversaries playing spoilsport against the network connection and communication established between the peers disrupting the security policies and infrastructure hosted to ensure holistic and comprehensive security solutions [2]. This poses a grave threat to security models and infrastructure viz. Intrusion Detection System (IDS), Firewalls and Application proxy where in real time live attack detection, even millisecond matters [3][4].

Millions of features exist in ultra-high dimensional dataset where not all features either equally contribute to the target class variable or have a direct impact on the IDS Classification model. Extraneous features may slow down the functioning of inference engine. IDS touted as latency sensitive application where only informative and useful features need to be provided. This mandate the application of feature engineering paradigm where feature selection and feature extraction are an integral part of it [5]. Feature Preprocessing Phase is the time heavy component in the IDS classification pipeline. The performance of the IDS falters and takes high time for detecting the attackers when it is shrouded with cluttered data. Attacker may silently creep in to the network and cause damage. Optimizing this stage using metaheuristics search algorithms yields the biggest performance gain. This research work inspires from the "Optimized Ensemble feature selection using Metaheuristics Search Algorithms for IDS" [6][7].

As network threats evolve, adaptivity and robustness in feature selection are crucial. Static aggregation (mean, voting) cannot handle dynamic traffic patterns. Metaheuristic or learning-based aggregators can adaptively tune the fusion process—leading to:

- more compact feature sets,
- better attack detection,
- and reduced false alarms.

This makes aggregation strategy not just a technical detail, but a core determinant of IDS performance and reliability [8]. Quality and smart features are seeded to the IDS model where holistic combination and recombination of potential feature subsets are facilitated through deployment of Meta-aggregator. Intelligent, adaptive, context-aware MA capture dynamic complementary insights from the base feature selectors in pre-empting myopic decisions propagated by standalone filters. This method also promises to improve the stability and robustness of the selected consensus driven features [9][10].

Research Questions/Challenges:

- How does a metaheuristic-based aggregator improve over conventional ensemble aggregation in adjudicating IDS performance metrics?
- How do we substantiate the usage of Genetic Algorithm as a promising Meta-Aggregator and compare it against CAS.
- What trade-offs exist between accuracy, feature selection time, and scalability across different optimization strategies?

The organization of the paper is as follows. Section 2 underscores the importance of dimensionality reduction techniques boosting IDS performance metrics. Section 3 highlights the importance of ensemble aggregation strategy as a key determinant of IDS success and its failures in arriving at an optimum solution. This gap has mooted the idea of incorporation of Metaheuristics acting as intelligent aggregation agent (Meta-Aggregator) is deliberated in detail in Section 4. This section also exemplifies the adaptability of Genetic Algorithm as MA and its associated merits. Proposed model with GA as MA is elaborately discussed in Section 5 backed with algorithms and conceptual workflow. Section 6 discusses the experimental setup, results and discussion. Section 7 concludes the paper.

2. LITERATURE SURVEY/BACKGROUND STUDY

2.1. Intrusion Detection System (IDS)

Intrusion: Any inadvertent activity that compromise the three security pillars viz. Confidentiality, Integrity and Availability (CIA) of the System in use. The major classes of attacks are DoS (Denial of Service), R2L (Root to Local), U2R (User to Root), Probe and Normal profile. A set of techniques, methods, tools and procedures used to analyze and detect an abnormal/anomalous behavior in the inward data traffic entering the network [11][12].

Ideal Characteristics to be possessed by IDS are as follows:

- High Detection Accuracy
- Real Time Detection Capability
- Scalability
- Adaptability & Evolvability
- Efficiency & Parsimony
- Robustness & Reliability
- Interpretability & Explainability
- Deployment Flexibility
- Interoperability
- Security of the IDS itself

Figure 1 shows the general taxonomy of IDS where the classification is based on two parameters viz. source of data and detection methods [13]. Depending on the origin of network traffic the IDS can be classed into Host-based IDS and Network-based IDS. Based on the behaviour of the attacker and their activity profiling yields two classes viz. Anomaly detection (suitable for zero-day threats, rare and fresh attacks) and Misuse detection methods (suitable for Labelled attacks) [14][15].

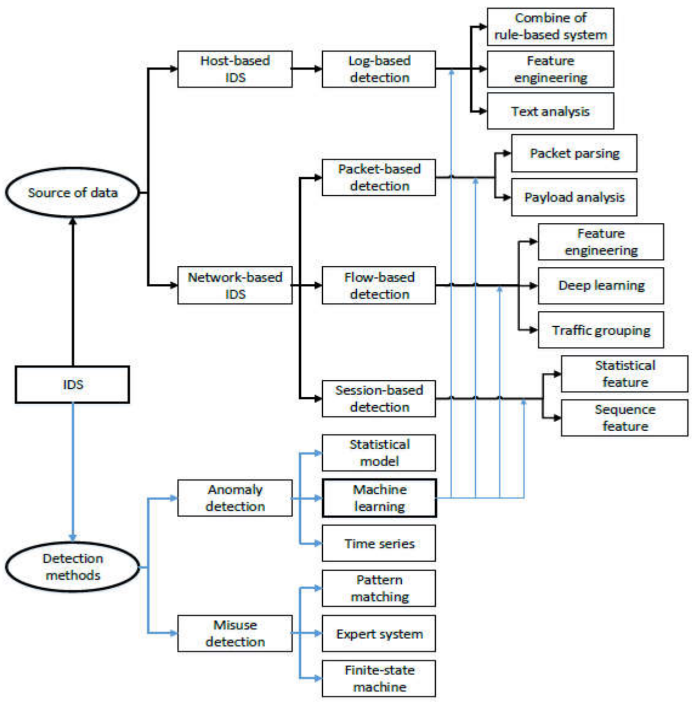

Figure 1: General Taxonomy of IDS

2.2 Feature Selection (FS)

The process of identifying and retaining the most informative attributes with high predictive power in a dataset while removing irrelevant, redundant, or noisy ones to enable the classification model detect attacks accurately, efficiently, and with minimal redundancy [16]. Benefits of Feature selection in IDS is shown in Fig. 2. High-dimensional datasets (like NSL-KDD, CICIDS2017, UNSW-NB15) make feature selection crucial for performance, scalability, and interpretability [17].

Figures 2, 3, 4, 5 are showcased to enhance the understanding of basics of feature selection, need for it, types of FS and classification of FS techniques based on different perspectives.

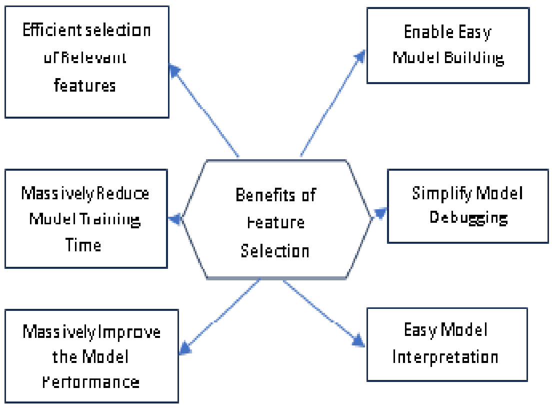

Figure 2: Benefits of Feature Selection in IDS

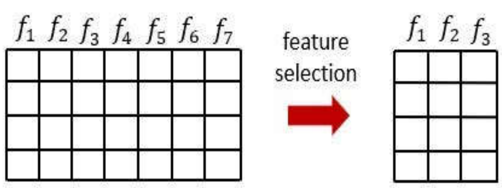

Figure 3: Feature Selection – Pictorial Representation

Figure 4: Classification of FS techniques based on different Perspectives

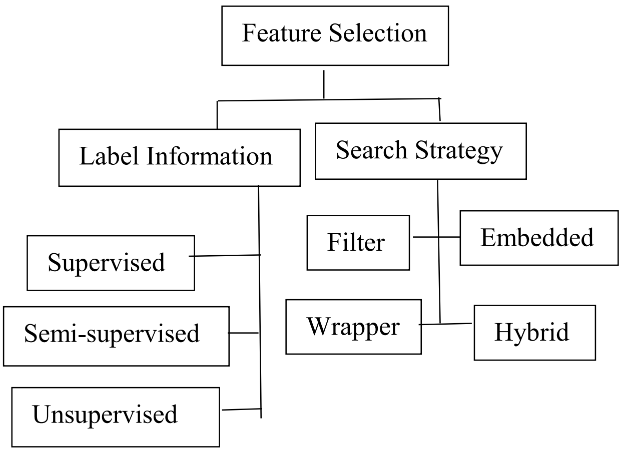

Figure 5: Types of FS

2.3 Failures Encountered by Single Feature Selection Method

Deploying a single (baseline) feature selection may be beneficial for several reasons in resource constrained environments but it has its own share of drawbacks as follows,

Feature Selection (FS) based on supervision strategy	Unsupervised Supervised Semi supervised
Feature selection based on evaluation function	Filter Wrapper Hybrid
Feature selection from a data perspective	Conventional data based Structured data based FS with heterogenous data FS on data streams
Feature selection based on search strategy	Exponential Sequential Metaheuristic Based

- No one method works best across all scenarios.
- Poor Generalization Across Traffic Types
- Bias Toward Certain Feature Types
- Vulnerability to Noisy or Redundant Features
- High Computational Cost
- Inflexibility in Evolving Threat Landscapes
- Missed Complementary Insights
- Incomplete feature sets

These failures have mooted the idea of adoption of Ensemble Feature Selection (EFS) in IDS [25][26].

3. Ensemble Feature Selection (EFS)

3.1 Description

The process of combining the outputs of multiple feature selection methods to produce a more stable, accurate, and robust set of features for a machine learning model. No single feature selection method is perfect. Each base selector may capture different aspects of data relevance, redundancy, or noise [27]. Combining multiple methods can capture complementary strengths and reduce weaknesses. It is a Mixture of Experts/ Wisdom of Crowd/Collective Intelligence. The need for EFS in IDS is attributed to several reasons viz.

- Stability
- Better Generalization
- Reduced Method Bias
- Improved Detection Rates
- Resilience to Evolving Threats
- Balanced, stable, and higher-performing feature subset
- Valuable in imbalanced attack datasets

Thus, EFS plays a pivotal role in IDS model quality. Homogenous and heterogenous ensemble architecture exists to validate/substantiate the diverse nature required for effective ensemble functioning supplemented with the various levels/layers that can be tweaked for performance gain [28][29]. Various levels in a EFS design architecture that can be tweaked for IDS performance gains is highlighted in Figure 6.

Variations/changes instigated in dataset level and learner method level such as data perturbation, function perturbation and hybrid perturbation has been extensively addressed in the literature leaving a void/vacuum in the combination (aggregation) level that has elicited research interest in tweaking the aggregation methods to evolve and embrace nature inspired search strategies viz. Metaheuristics search algorithms as intelligent aggregation agent [30].

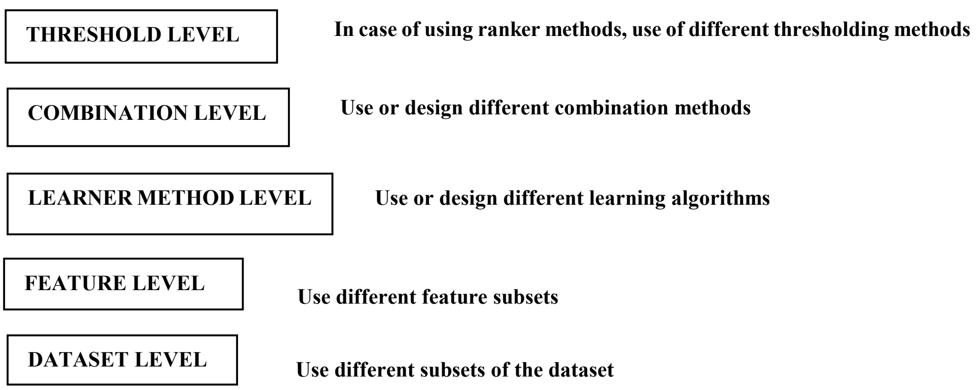

Figure 6: The Different Levels that can be varied in an Ensemble Feature Selection Design

3.2 Merits of Minimal Feature Preprocessing/Selection Time on IDS

The feature selection itself can be a computationally heavy step, especially with large, high-dimensional network datasets. IDS face strict real-time constraints and detection requirements where balancing speed vs. accuracy becomes a cornerstone for its usability [18]. Scalability Concerns also arise in this voluminous nature of dataset demanding intelligent feature search (selection) methods to handpick useful features in minimal time. Longer feature selection times consume more

CPU/GPU cycles and memory causing undue delay in detection of potential attackers plaguing network resources and users. Measuring FST helps in choosing lightweight algorithms that fit within the available resources in continuous Learning Scenarios [19]. A marginal reduction in FST in the early preprocessing phase culminating in selection of high-valued features has a resounding impact in timely and live detection of attackers at the point of entry itself [20][21].

Key Factors affecting Feature Selection Time (FST) are dimensionality, data Complexity, Feature Representation, Volume of Data and Algorithm Type. An in-depth analysis performed on the requirement of FST for various types of data viz. text, audio, video, image and multimedia exhibit similar range values for multimedia and UHDD [22]. So, this mandates the adoption of intelligent feature search technique viz. Metaheuristics Enabled Feature Optimization and search for high dimensional IDS dataset. Recommended feature selection strategy for diverse IDS context do exist distinctly for Text-based IDS, Audio-based IDS, Image-Based IDS, Video-based IDS and Multimedia-based IDS [23]. Extensive deliberation on the current research has highlighted the stigma faced by individual FS method in constraining the feature space where the potential and useful features may escape and its resilience to class-oriented features (major and minor attacks) [24].

The increase in computational time and Feature Selection Time in EFS compared to standalone selectors is annulled by meta-level aggregation/integration (especially using metaheuristics) leading to a net improvement in IDS detection performance and feature selection efficiency. Adding of tinge of parallelism flavor to the meta-aggregation significantly mitigates the time overhead resulting in a balanced trade-off between accuracy, speed and scalability that is critical for real-time IDS. The co-ordination of ensemble crowd members/base feature selectors contributes to the disparity in FST. The optimization of aggregation and computational layers through parallel metaheuristics encourages EFS frameworks in outperforming individual selectors in both time and classification efficiency. This results in a more responsive (low latency), more accurate (through feature diversity) and more scalable (suitable for big data intrusion logs.)

EFS architecture is endowed with the privilege of tweaking/modifying/optimizing specific design levels for improving FST while maintaining or enhancing classification performance. The various design levels are as follows.

- Base Selector Level
- Ensemble Construction Level
- Aggregation Level (Metaheuristic/Voting mechanism)
- Feature Evaluation and Feedback level
- Infrastructure and Computation Level

The table below mentions the design levels that can be tweaked and the impact it has on efficiency and accuracy.

Table 1: Tweaking Ensemble Design Levels for IDS performance amelioration.

Design Level	**Tweaks**	**Impact**
Base Selector	Replace slow, complex filter methods (like	Reduces redundant

Level	ReliefF) with fast ranking-based filters (e.g., Chi-Square, Information Gain). Parallelize independent feature evaluations (using multiprocessing or GPU-based filtering). Use feature pre-screening (remove low-variance or redundant features before full ensemble processing).	computations. Decreases selection time by 20–40%. Minimal loss in classification accuracy when filters are complementary.
Ensemble Construction Level	Use subset-based ensemble formation instead of evaluating all filters on full data. Introduce dynamic weighting — only use high-confidence feature selectors in each iteration. Apply incremental or streaming ensemble updating for online IDS datasets.	Reduces the number of active filter computations. Maintains accuracy while cutting feature selection time by 30–50%. Improves adaptability for real-time IDS systems.
Aggregation Level (Metaheuristic / Voting Mechanism)	Replace fixed voting (mean, majority) with metaheuristic-driven aggregation (Genetic Algorithm, PSO, DE). Adopt parallel genetic algorithms (PGA) or distributed evolutionary frameworks to optimize aggregation faster. Use multi-objective fitness (accuracy, FPR, and time) for balanced convergence.	Optimizes feature subset selection globally rather than heuristically. Reduces aggregation time while improving overall accuracy. Demonstrated performance gain: ~10–15% higher accuracy ~35% faster feature convergence time
Feature Evaluation & Feedback Level	Introduce cross-validation caching to avoid repeated evaluations on the same subsets. Implement adaptive termination criteria (stop early when convergence threshold is met). Use surrogate modeling (e.g., regression-based prediction of fitness) to approximate evaluations.	Reduces repetitive classifier training. Decreases feature selection time by 25–60%. Maintains high detection stability.
Infrastructure / Computational Level	Deploy on multi-core or distributed clusters (e.g., Spark, Dask, or MPI-based parallel computing). Use asynchronous population update in genetic algorithms to avoid idle CPU cores.	Massive improvement in scalability and throughput. Up to 3–5× speed-up in end-to-end feature selection

	Optimize data structures with in-memory caching and GPU acceleration for large-scale IDS data (NSL-KDD, CICIDS)	

Incorporating a twist/tweak in aggregation level through metaheuristic guided ensemble aggregation witness a substantial performance gain with approximately a noticeable spike in accuracy values from 10 to 15% and 35% faster feature convergence time augmented with global optimization of feature subsets rather than the heuristic approach. The results have propelled us to adopt changes in Aggregation Layer and appreciate its importance and impact on IDS performance.

3.3 Importance of Aggregation Strategy in EFS for IDS

Aggregation plays a pivotal role in Ensemble Feature Selection (EFS), as it determines how the outputs of multiple feature selection techniques are fused into a final, stable subset of features that drives the performance of Intrusion Detection Systems (IDS)[31]. This step critically affects which features are selected, how consistent selections are across folds or runs and ultimately, how well the IDS generalizes [32]. Since different feature selection methods capture distinct statistical or structural properties of network traffic data, an effective aggregation strategy ensures that the final subset balances feature relevance, redundancy reduction, and robustness against noise.

The choice of aggregation method—whether simple rank averaging, voting, weighted fusion, or metaheuristic optimization—directly influences the IDS's classification accuracy, false positive rate, and generalization capability [33][34]. Poorly designed aggregation may overemphasize redundant or irrelevant features, leading to degraded detection performance, while adaptive and optimization-based aggregation can dynamically identify the most discriminative features for evolving attack patterns. Consequently, the aggregation strategy is not merely a post-processing step but a central component that governs the stability, interpretability, and overall efficiency of EFS-based IDS frameworks. Recent studies have demonstrated that adaptive, metaheuristic-driven aggregation approaches yield superior detection accuracy and lower false alarm rates compared to conventional static fusion methods, highlighting the strategic importance of aggregation design in modern IDS research [35].

3.4 Conventional Aggregation Strategies (CAS)

The Table 2 below shows select CAS adopted in EFS.

3.5 Deficiencies encountered by CAS

The flops embraced by the CAS has signaled the inquisitive incorporation of Metaheuristics enabled feature aggregation/integration or as a Meta-Aggregator in IDS framework [36].

- Equal Weighting Ignores Method Quality
- Loss of Rare but Critical Features
- Sensitivity to Rank Scale and Position
- Redundancy and Correlation Blindness
- Static and Non-Adaptive Aggregation Fails with Concept Drift
- Bias Toward High-Frequency Features
- Lack of Context-Aware Aggregation
- Dataset & Attack Diversity Ignored

Deployment of MH as a preliminary feature subset optimizer has been extensively deliberated leading a lacuna in its enrolment as Meta-aggregator [37]. This when judiciously employed leads to incremental performance gains than those accrued through CAS.

Table 2: Conventional Aggregation Strategies

Strategy	Description	Type	Common Use Cases
Rank Aggregation	Combine ranked lists of features from each selector using methods like Borda, average rank, or reciprocal rank fusion.	Ranking-based	When selectors produce full ranked lists.
Score Averaging	Compute the mean of normalized feature scores from each selector.	Score-based	When feature selectors output importance scores.
Score Voting (Hard)	A feature gets 1 vote from each selector that selects it; features with most votes are selected.	Voting-based	Simple consensus across binary selectors.
Weighted Voting	Like score voting, but each selector is given a weight based on accuracy, reliability, or prior knowledge.	Voting-based	When selector performance varies across datasets.
Thresholding	Select features that meet a predefined threshold in a fixed number or percentage of selectors.	Binary/Hybrid	To ensure consistency across selectors.
Intersection	Only keep features selected by **all** selectors.	Binary	High precision, but risks low recall.
Union	Combine features selected by **any** of the selectors.	Binary	High recall, may include irrelevant features.
Majority Voting	Select features chosen by **more than half** of the selectors.	Voting-based	Balanced between precision and recall.
Meta-ranking	Learn weights for selectors or ranks using a validation set or meta-learning.	Learning-based	Adaptive to dataset, useful in stacked ensembles.

4. METAHEURISTICS (MH)

4.1 Definition

MH is a high-level Problem-independent general purpose optimization frameworks designed to efficiently explore large and complex search spaces by combining randomization (stochastic), adaptation, and guiding strategies without guaranteeing an exact optimal solution but aiming for near-optimal and practical solutions within reasonable time and with incomplete information. Application of exact methods for the above said purpose is computationally impossible for large IDS datasets where complexity is 2^n where n is the number of features. This method attempts to perform an exhaustive exploration of feature space where all possible feature subsets and their combinations are tested in an exponential time [38]. Heuristic methods advocating correlation-based feature selection are considered fast and problem-specific but may miss global relevance and better convergence. The Metaheuristic search methods apply algorithms viz. Genetic Algorithm or Particle Swarm Optimization to search diverse rich, compact, high discriminative feature subsets aiming at discovering attack signatures with ease and minimal latency. It helps to find good trade-off feature sets efficiently, without being tied to a specific dataset [39].

4.2 Unique Traits of MH as Intelligent Feature Aggregation Agent.

- Problem-Independence & Flexibility
- Stochastic Nature (Randomization)
- Scalability to High-Dimensional Data
- Adaptivity / Self-Tuning/Self-Learning Capability
- Multi-objective Optimization Capability
- Robustness to Noise and Redundancy
- Exploration vs. Exploitation Balance
- Approximate but Scalable Solutions
- Global Search Ability

Dynamic Feature Weight Learning:
Instead of static rank averaging, metaheuristics assign adaptive weights to feature scores from different selectors based on their contribution to accuracy and robustness.

Global Optimization:
They explore the global search space, avoiding local minima that plague static aggregation methods [40].

Multi-objective Capability:
Metaheuristics can jointly optimize accuracy, false positive rate (FPR), and feature selection time, leading to a more balanced IDS model.

4.3 Meta-Aggregator (MA)

A MA in a filter feature selector ensemble for IDS is a higher-level optimization mechanism that intelligently combines the outputs of multiple filter-based feature selectors (e.g., Chi-square, Information Gain, ReliefF, Correlation) into a unified, optimized feature subset, ensuring that the resulting set maximizes intrusion detection accuracy, efficiency, and robustness across diverse attack categories. Rather than using a simple rule like Majority voting/Average Ranking, a MA applies optimization learning/adaptive strategies to synthesize the diverse insights from different selectors [41]. A MA would

- Encode different feature combinations.
- Evaluate each subset based on classifier accuracy (e.g with RF, SVM)
- Evolve toward the most performant subset, considering all filter insights.

4.4 MH as MA in EFS – Underexplored.

Considerable research effort has been directed to incorporate several MH algorithms viz. GA, PSO and ACO as meta-aggregators for EFS where it tries to optimize the aggregation strategy rather than performing direct feature selection. This idea has been well received in reality and GA-based Meta-Aggregator (MA) improved detection accuracy by 4% while reducing selected features by 30%

compared to conventional rank aggregation [42]. Parallel implementations of GA further reduced feature selection time by ~40%, making them more viable for real-time IDS.

4.5 Special Purpose of MA in IDS

1. **Resolve Conflicts Across Filters**
   Different modus-operandi adopted by each filter method (e.g., Chi-square may favour categorical features, ReliefF may emphasize local dependencies) rank features differently which in turn is reconciled effectively by incorporating meta-aggregator to form a balanced subset.
2. **Preserve Rare but Critical Features**
   A meta-aggregator ensures to retain rare yet critical features important for detecting rare attacks e.g., U2R, R2L unlike conventional aggregators that tends to drop such features [43].
3. **Optimize Multi-Objective Trade-offs**
   MA is entrusted with responsibility of simultaneously optimizing diverse performance parameters viz. accuracy, false positive rate, computational speed, feature subset size.
4. **Enhance Robustness Across Datasets**
   A meta-aggregator ensures consistent performance across heterogeneous environments and different IDS datasets hosting diverse feature types and relevance [44].
5. **Pre-empts Myopic Feature Selector Decisions**
   A myopic feature selection strategy might evaluate and select features individually based on a single criterion (e.g, correlation with the target variable.) without considering how features interact when used together. Reliance on myopic filters (selecting top-k features individually) can miss inter-feature dependencies critical for detecting sophisticated attacks. Using MA helps avoid myopic decisions by evaluating feature combinations holistically to enable smart feature seeding using filter ensemble outputs [45].

4.6 MH Based Aggregator vs Conventional Ensemble Aggregation in IDS

Conventional methods assume equal relevance among individual feature selectors, whereas metaheuristic aggregators learn which selector or feature contributes most to intrusion detection effectiveness as highlighted in Table 3 [46].

4.7 Preference of Genetic Algorithm (GA) as MA in IDS

A GA is a sequential, population-based metaheuristic optimization method inspired by natural selection, where candidate solutions (chromosomes) evolve over generations using selection, crossover, and mutation operators to search for near-optimal solutions in complex spaces [47]. GA works on a single population and evolves step by step in sequence limited by the computational capacity of a single processor/core. GA endorses optimized feature selection but may be slow when traffic datasets are huge (CICIDS2017, UNSW-NB15) rendering evolutionary search on a single population in sequence [48].

Table 3: Difference Between Meta-Aggregator and Conventional Ensemble Aggregator

Aspect	Conventional Aggregator	Metaheuristic-Based Aggregator
Aggregation Mechanism	Uses static rules (e.g., majority voting, rank averaging, mean weighting)	Uses adaptive search algorithms (e.g., Genetic Algorithm, Particle Swarm Optimization, Ant Colony Optimization)
Adaptivity	Fixed weighting, not responsive to data variations	Dynamic weighting based on fitness and feedback from IDS performance
Exploration–Exploitation	No optimization process	Actively explores the search space to find optimal feature weight combinations
Computational Intelligence	Rule-based, deterministic	Stochastic, self-learning, and adaptive
Optimization Goal	Combine selector outputs directly	Optimize performance metrics (accuracy, FPR, time) simultaneously

Basic Steps in GA:

1. Initialize population randomly
2. Evaluate fitness of each individual
3. Select individuals for reproduction
4. Crossover (combine genes)
5. Mutation (introduce variability)
6. Replace and repeat for multiple generations

Using a Genetic Algorithm (GA) as a fusion agent for Intermediate Feature Subset (IFS) generated by EFS offers several advantages, including the ability to find optimal or near-optimal combinations of features, handle complex interactions between features, and provide robustness due to its evolutionary nature [49]. GA allows randomness enhanced FS and its parallelization reduces overall data preprocessing and allows large population count which in turn leads to better feature selection [41]. GA is touted as a natural choice for optimal aggregation agent because they are well suited for searching complex, high-dimensional and combinatorial spaces [50].

4.8 Key Traits of GA supporting its participation as MA in IDS

Population-Based Search (Parallel Exploration)
Crossover Operator (Natural Aggregation Mechanism)
Mutation Operator (Diversity & Rare Feature Preservation)
Fitness Function Flexibility (Multi-Objective Optimization)
Robustness & Scalability
Global Exploration + Local Exploitation Balance
Stability of Feature Subset Selection

5 Proposed System

The proposed model's feature selection pipeline consists of four stages. They are sample distribution stage, filter ensemble stage, metaheuristic aggregation stage and classifier.

- **Distribution phase**

(Subsampling and random sampling) (several chunks of disjoint training set/records)

- **Filter Ensemble Stage**
  This stage combines scores from multiple filter methods to rank features and the top ranked features are passed to Metaheuristic Aggregation Stage (MHAS) where it adopts Genetic algorithm to select optimal feature subset through optimization of fitness function (F1-score) using a wrapper classifier (Random Forest (RF)).
- **Metaheuristic Aggregation Stage**
  A **Genetic Algorithm** is deployed as a **meta-aggregator**, treating feature ranking positions as chromosomes and applying crossover, mutation, and selection to optimize aggregate rankings. The fitness function maximizes classification accuracy while minimizing feature redundancy.
- **Testing phase (Classifier)**.
  The resulting aggregated optimal feature subset is then used by an Intrusion Detection System (IDS) model/classifier in GA, which benefits from improved detection accuracy, reduced false positives, faster detection, less computational cost and robustness to noise.

The Ensemble Learning Infused Feature Selection Method (ELIFSM) for efficient IDS comprises GA with filter feature selector ensemble (bagging). The work here focuses on ameliorating the Feature Aggregation Mechanism (FAM) through adoption of metaheuristic search in the enlarged feature search space. Sifting through this deep/enormous IFS and selecting an optimal solution is a daunting task. Leveraging the metaheuristic technique as a solution combiner/aggregator to optimize the feature combination process faced with 2 conflicting objectives of minimizing the size and number of the features subset, maximizing the classification accuracy associated with minimal feature selection time and feature gathering cost.

Proposed Architecture Diagram and schematic flow of the proposed model is highlighted in Figure 7 and Figure 8 (the following page)

The overall workflow of the proposed model be like:

1. **Generating Intermediate Subsets:**
   Ensemble Feature subset selection (EFS) – This phase attempts to combine the outputs of multiple base feature selection methods (Mutual Information, ReliefF, Chi-square, Correlation Coefficient, ANOVA) applied on the NSL KDD dataset and obtain diverse subsets. Each method generates a ranked list or subset of features. These features termed as IFS may substantially differ in the representation depending on the base algorithm adopted and tries to capture different aspects of the feature relevance towards the target variable.
2. **Encoding for Genetic Algorithm**
   Represent the union of features from all subsets as chromosomes, where each gene indicates the inclusion or exclusion of a feature.
3. **Genetic Algorithm Execution**
   Evaluating the fitness of each chromosome based on criteria such as classification accuracy or other performance metrics. Genetic Algorithm is introduced to aggregate or to powerfully explore the optimal combination of features from these intermediate subsets. Refine/Re-Refine the combination/aggregation of features subsets towards improving a target criterion (e.g. Classifier Accuracy, AUC, etc.).
4. **Aggregating Results**
   Select the feature subset(s) with the highest fitness scores for model training and evaluation and aggregate the results to identify an optimal or near-optimal feature subset that enhances model performance.

5. The optimized feature subset is evaluated using classifiers such as **Random Forest (RF)**, **Support Vector Machine (SVM)**, and **XGBoost**. Metrics include **accuracy**, **false positive rate (FPR)**, **feature selection time**, and **number of selected features**. Comparative benchmarking is performed against conventional aggregation and standalone selectors.

Conceptual Impact Flow

Multiple FS methods → Aggregation Strategy → Selected Feature Subset
↓
Feature Quality (relevance, redundancy)
↓
IDS Classifier Performance
↓
Detection Accuracy, False Positives, Generalization

Thus, aggregation acts as the "bridge" between diverse selectors and IDS performance outcomes. Figure 9 highlights the usage of Genetic Algorithm as Meta-Aggregator for IDS performance enhancement.

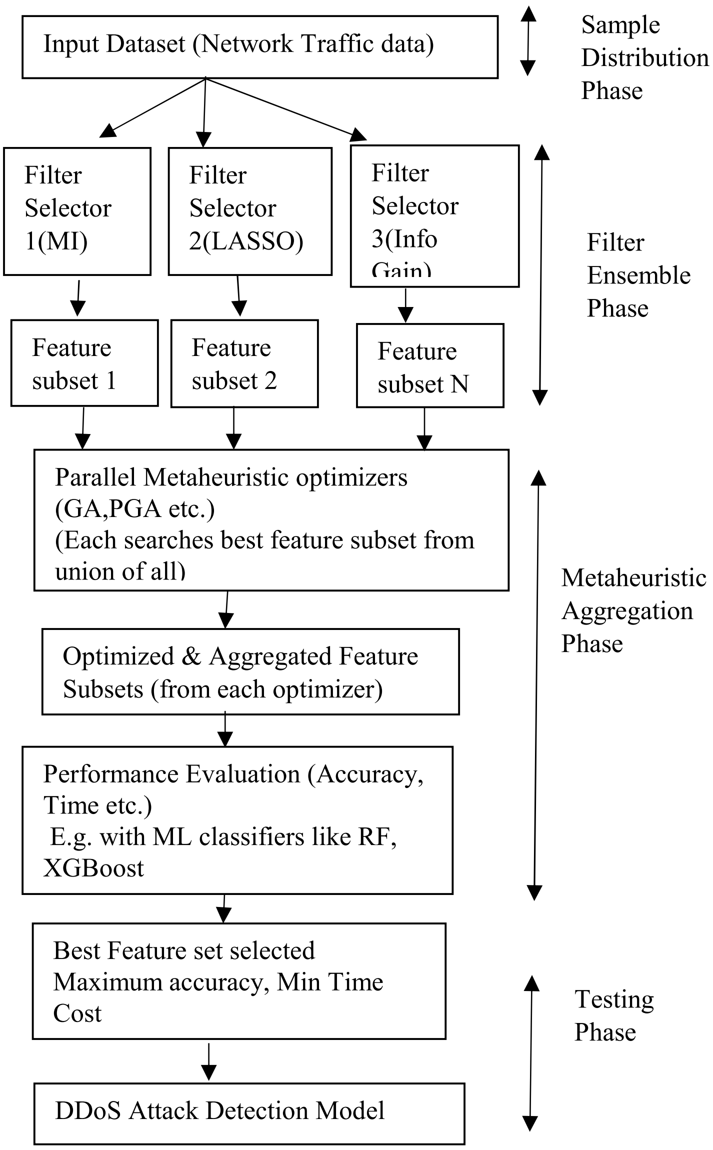

Figure 7: Proposed Architecture Diagram

Algorithm: GA-Based Meta-Aggregator for Ensemble Feature Selection

// Input:

- D: Dataset with N features
- F = {F1, F2, ..., Fm}: Set of m filter-based feature selectors
- k: Top-k features selected by each filter
- pop_size: Genetic algorithm population size
- G: Number of generations
- CR: Crossover probability
- MR: Mutation probability
- fitness_fn: Function evaluating a feature subset (e.g., classifier accuracy)

// Output:

- Optimal aggregated feature subset F_opt

Step 1: Run Filter Feature Selectors

```
        Initialize F_partial ← empty list
        for each filter Fi in F do
            Fi_k ← Top-k features ranked by Fi
            Add Fi_k to F_partial
        end for
        F_union ← Union of all features in F_partial  // Reduced feature space
```

Step 2: Initialize Genetic Algorithm

```
        Let L ← length(F_union)
        Initialize population P with pop_size binary chromosomes of length L
        Each chromosome Ci ∈ P encodes a subset of F_union
```

Step 3: Evaluate Fitness

```
        for each chromosome Ci in population P do
            Subset_i ← decode(Ci, F_union)
            Fitness[Ci] ← fitness_fn(D using Subset_i)
        end for
```

Step 4: Run Evolutionary Loop

```
    for gen from 1 to G do
        // Selection (Tournament or Roulette Wheel)
            P_selected ← select parents based on Fitness
        // Crossover
          for i from 1 to pop_size/2 do
            if random() < CR then
            (offspring1, offspring2) ← crossover(parent1, parent2)
            else
            (offspring1, offspring2) ← (parent1, parent2)
            end if
            Add offspring1, offspring2 to new population
          end for
        // Mutation
            for each chromosome C in new population do
              for each gene g in C do
                if random() < MR then
                flip g
                end if
              end for
            end for
        // Replace old population
```

P ← new population
// **Re-evaluate Fitness**
for each Ci in P do
Subset_i ← decode(Ci, F_union)
Fitness[Ci] ← fitness_fn(D using Subset_i)
end for
end for

Step 5: Output the Best Solution

C_best ← chromosome with max(Fitness)
F_opt ← decode(C_best, F_union)
return F_opt

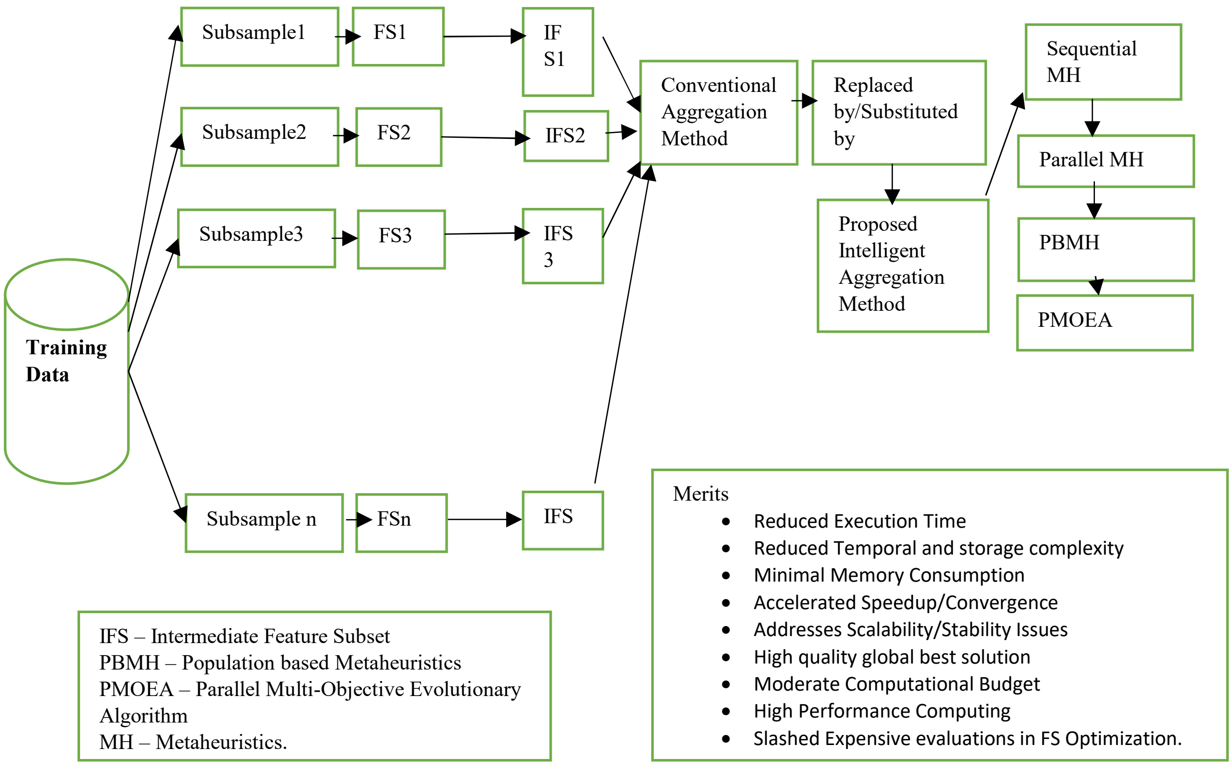

Figure 8: Design Flow of the Proposed Model.

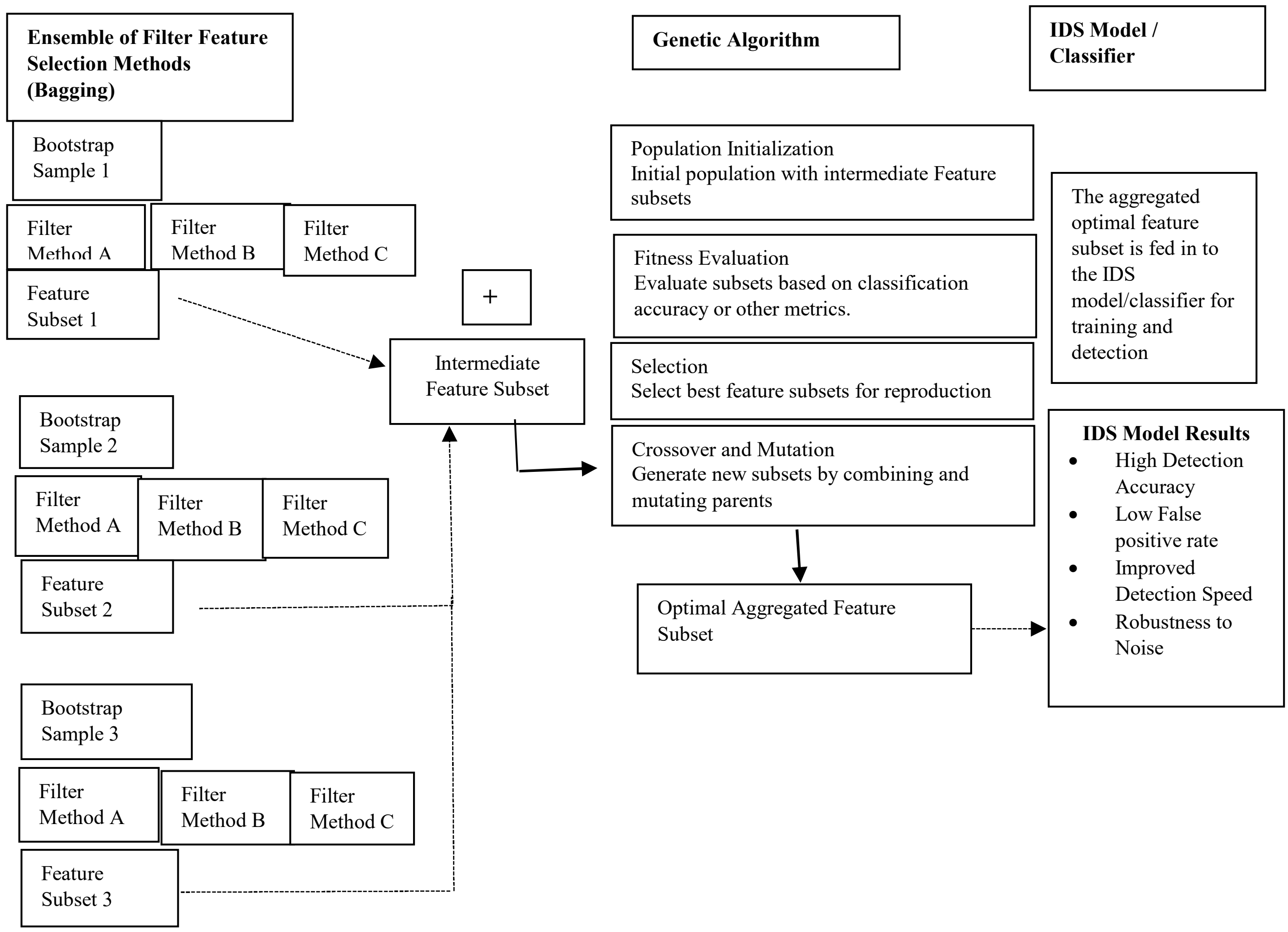

Figure 9: Genetic Algorithm as Meta-Aggregator.

6 EXPERIMENTAL SETUP

In this section, we deployed our proposed model both on the constrained dataset (ensemble pruned dataset - 17 features) created by an ensemble of five filter selection methods and on the full dataset (41 features). Attempts were made to construe genetic algorithm that innately supports feature optimization as meta-aggregator (ie aggregation/fusion agent) for the IFS rolled out by ensemble selectors. Experiments were conducted to assess the advantages reaped by the incorporation of GA as MA using metrics viz. FST, No. of features selected, Accuracy, FPR. Experimental setup required for this model is enlisted below.

Experimental Setup:

Dataset Used	**NSL-KDD** dataset A refined version of the KDD'99 dataset. Contains labeled connection records as normal or attack. Pre-processed to remove redundancy and class imbalance.

Feature Selection (Ensemble Stage) • **Five filter-based techniques** applied:	• ANOVA (F-test) • ReliefF • Mutual Information • Chi-Squared • Pearson Correlation Coefficient • The selected features from each method are aggregated. A **pruned ensemble subset** is generated using intersection/union or voting (CAS).
Meta-Aggregators **Genetic Algorithm (GA)**	Standard evolutionary optimization. Selection, crossover, mutation steps to optimize feature subset. Fitness function: maximize detection accuracy, minimize feature count.
Classifier Used	A consistent classifier (e.g., Random Forest, SVM, or XGBoost) was used to evaluate selected features. Performance evaluated using **5-fold cross-validation**.
Metrics Measured **Detection Accuracy (%)**: **Feature Selection Time (s)**: **FPR** **Number of Selected features:**	Ability to correctly classify attack and normal samples. Time taken by GA to converge on the final feature subset. True positive rate of attack detection. Final dimensionality of the feature set.
Tools and Environment	Programming Language: Python Libraries: scikit-learn, numpy,pandas,matplotlib,DEAP (for GA), joblib or multiprocessing (for PGA) System Specs: e.g., Intel i7 CPU, 16 GB RAM, Ubuntu 20.04

The Table 4 below presents the comparative results of Standalone filter, Filter Feature selector ensemble (Conventional Ensemble Aggregator(voting)), GA as Metaheuristic Aggregator on complete dataset as well on Ensemble pruned set. These results are subject to variations in every population iteration depending on GA randomness.

6.1 Results and Discussion:

The GA meta-aggregator outperforms standalone and conventional ensembles by producing a compact, high-quality leaner feature subset (15 selected features) with >12% higher accuracy and ~3% lower FPR compared to ReliefF. Its multi-objective nature enables balancing between relevance, diversity, and stability — crucial for real-time IDS deployment. When applied on ensemble-pruned data, the GA's search space is smaller and more informative, resulting in faster convergence and more consistent results. Using GA on the **ensemble-pruned dataset** yields slightly higher detection accuracy and lower computation time because the search space is smaller and more informative. However, even on the **complete NSL-KDD dataset**, GA remains superior to traditional EFS methods due to its global optimization capability. The higher feature selection time in GA meta-aggregator is attributed to iterative fitness evaluations of the complete 41 listed features as against the ensemble pruned features subset with feature count as 17. The high selection time is offset with substantial increase in detection accuracy with considerable reduction in FPR and the number of features selected ie 12.

Table 4: Comparative Study of Diverse Feature Selection, Optimization and Aggregation Techniques

Metric	ReliefF (Standalone Filter)	Voting (Conventional Ensemble)	GA on Complete Dataset	GA on Ensemble-Pruned Dataset as Meta-Aggregator (Proposed)	Improvement (Pruned vs. Complete)
Detection Accuracy	86.2 %	92.8 %	97.4 %	**98.6 %**	+1.2 %
Feature Selection Time	32.7 s	73.4 s	184.6 s	**142.3 s**	–22.9 %
No. of Features Selected	21	17	15	**12**	–3
False Positive Rate (FPR)	4.1 %	2.6 %	1.5 %	**1.1 %**	–0.4 %

7 Conclusion

The increasing sophistication of network attacks poses serious challenges to cyber security infrastructure (IDS). FS and FST plays a pivotal role in enhancing the IDS performance. This research work integrates ensemble learning with Metaheuristics algorithm to form robust feature selection pipelines to generate a refined, low-redundancy feature set with an aim to improve attack detection rates. MH based Ensemble Feature Selection Method (Ensemble Attribute Aggregator/Meta-aggregator) have emerged as effective strategies for optimizing feature subsets tailored for diverse attack detection with minimal latency and feature selection time. This proposed idea aims to shorten the FST in choosing high valued features that invigorates the classification potential with reduced detection latency and computational complexity.

Future work:

How parallelization of the MA could impact FST, scalability and other associated metrics. Accuracy vs performance trade off analysis to be conducted with diverse optimization strategies as MA on the dataset. ELIFSM can be considered with different ensemble design architectures viz. boosting and stacking and compared against base line feature selectors, conventional ensemble aggregator.

REFERENCES

1. Kumar, A., & Kumar, S. (2025). *Metaheuristic-Enhanced Feature Selection for High-Accuracy Intrusion Detection in Cloud Computing*. **Procedia Computer Science, 259**, 640–649. https://doi.org/10.1016/j.procs.2025.04.0144
2. Cheng, Z.-H., Shang, H., & Qian, C. (2024). *Detection-Rate-Emphasized Multi-objective Evolutionary Feature Selection for Network Intrusion Detection*. arXiv. https://doi.org/10.48550/arXiv.2406.09180
3. Mehanović, D., et al. (2021). *Feature selection using cloud-based parallel genetic algorithm for intrusion detection data classification*. **Neural Computing and Applications, 33**, 11861–11873. https://doi.org/10.1007/s00521-021-05871-5
4. Liu, X., & Du, Y. (2023). *Towards Effective Feature Selection for IoT Botnet Attack Detection Using a Genetic Algorithm*. **Electronics, 12**(5), Article 1260. https://doi.org/10.3390/electronics12051260
5. Maseno, E. M., & Wang, Z. (2024). *Hybrid wrapper feature selection method based on genetic algorithm and extreme learning machine for intrusion detection*. **Journal of Big Data, 11**, Article 24. https://doi.org/10.1186/s40537-024-00887-9
6. Asghari Varzaneh, Z., & Hosseini, S. (2024). *An improved equilibrium optimization algorithm for feature selection problem in network intrusion detection*. **Scientific Reports, 14**, Article 18696. https://doi.org/10.1038/s41598-024-67488-7
7. Saheed, Y. K., Kehinde, T. O., Raji, M. A., & Baba, U. A. (2023). *Feature selection in intrusion detection systems: A new hybrid fusion of Bat algorithm and Residue Number System*. **Journal of Information and Telecommunication, 8**(2), 189–207. https://doi.org/10.1080/24751839.2023.2272484
8. Bakro, M., Kumar, R. R., Alabrah, A. A., Ashraf, Z., Bisoy, S. K., Parveen, N., Khawatmi, S., & Abdelsalam, A. (2023). *Efficient Intrusion Detection System in the Cloud Using Fusion Feature Selection Approaches and an Ensemble Classifier*. **Electronics, 12**(11), Article 2427. https://doi.org/10.3390/electronics12112427
9. Zubair Khan, M., Ahmad Reshi, A., Shafi, S., Aljubayri, I., (2025). An adaptive hybrid framework for IIoT intrusion detection using neural networks and feature optimization using genetic algorithms. **Discover Sustainability**. Retrieved from https://doaj.org/article/414380eee4384de8b3d0d29caf6b6600
10. Laxmi Lydia, E., Sripada N. S. V. S. C. Ramesh, Veronika Denisovich, G. Jose Moses, Seongsoo Cho, Srijana Acharya, & Cheolhee Yoon. (2025). *African buffalo optimization with deep learning-based intrusion detection in cyber-physical systems. Scientific Reports, 15*(1), Article 10219. https://doi.org/10.1038/s41598-025-91500-3
11. Pramanick, N., Mathew, J., Selvarajan, S., & Agarwal, M. (2025). *Leveraging stacking machine learning models and optimization for improved cyberattack detection*. *Scientific Reports, 15*(1), Article 16757. https://doi.org/10.1038/s41598-025-01052-9
12. Waghmode, P., Kanumuri, M., El-Ocla, H., & Boyle, T. (2025). Intrusion detection system based on machine learning using least square support vector machine. *Scientific Reports, 15*(1), Article 12066. https://doi.org/10.1038/s41598-025-95621-7
13. Hindy H. Brosset D, Bayne E, Amar Seeam, Christos Tachtatzis, Robert Atkinson, and Xavier Bellekens. 2018. "A Taxonomy and Survey of Intrusion Detection System Design Techniques, Network Threats and Datasets". 1, 1 (June 2018), 35 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn
14. Torabi M, Udzir N.I, Abdullah M.T, Yaakob R, "A Review on Feature Selection and Ensemble Techniques for Intrusion Detection System", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 5, 2021

15. Umar M.A, Chen Z, Shuaib K, Liu Y, "Effects of feature selection and normalization on network intrusion detection, Data Science and Management, 2024, ISSN 2666-7649, https://doi.org/10.1016/j.dsm.2024.08.001.
16. Lin Y, Ren X, Wang S, " Ensemble Feature Selection based on Multiple Metrics and Improved Aggregation Strategies", ISCER '24: Proceedings of the 2024 3rd International Symposium on Control Engineering and Robotics, Pages 99 – 103, https://doi.org/10.1145/3679409.3679429
17. Haro A, Cerruela G, Pedrajasa N, "Ensembles of feature selectors for dealing with class-imbalanced datasets: A proposal and comparative study", Elsevier, Information Sciences, 2020, Spain.
18. Zhang Y, Zhang H, Zhang B, "An Effective Ensemble Automatic Feature Selection Method for Network Intrusion Detection. Information 2022, 13, 314. https:// doi.org/10.3390/info13070314
19. Can Q.T, Nguyen T.D, Pham M.B, Nguyen T, AnTran T.H, Dinh T.M, "An Innovative Hybrid Model for Effective DDoS Attack Detection in Software Defined Networks", International Journal of Computer Networks & Communications (IJCNC) vol 16, No 6, November 2024, DOI: 10.5121/ijcnc.2024.16607
20. Alkasassbeh M, Baddar A.H, "Intrusion Detection Systems: A State-of-the-Art Taxonomy and Survey", *Arab J Sci Eng* **48**, 10021–10064 (2023). https://doi.org/10.1007/s13369-022-07412-1
21. Ouyang, H., Lin, X., Li, S. *et al (2025).* Recent metaheuristic algorithms for multi-objective feature selection: review, applications, open issues and challenges. *Cluster Comput* 28, 467. https://doi.org/10.1007/s10586-024-04996-1
22. Haq N F, Onik A R, and Shah F M, "An ensemble framework of anomaly detection using hybridized feature selection approach (HFSA)," *2015 SAI Intelligent Systems Conference (IntelliSys)*, London, UK, 2015, pp. 989-995, doi: 10.1109/IntelliSys.2015.7361264.
23. Arora J, Agrawal U, Tiwari P, Gupta D, Khanna A, (2020), Ensemble Feature Selection Method Based on Recently Developed Nature-Inspired Algorithms, International Conference on Innovative Computing and Communications, Advances in Intelligent Systems and Computing 1087, https://doi.org/10.1007/978-981-15-1286-5_39
24. Soheili M, Amir Haeri M.A, "Distributed Ensemble Feature Selection Framework for High-Dimensional and High-Skewed Imbalanced Big Dataset," *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, Orlando, FL, USA, 2021, pp. 1-8, doi: 10.1109/SSCI50451.2021.9659937.
25. Jaw E, Wang X, "Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach", *Symmetry* **2021**, *13*, 1764. https://doi.org/10.3390/sym13101764
26. Pes B, Dessì N, Angioni M, "Exploiting the ensemble paradigm for stable feature selection: A case study on high-dimensional genomic data, Information Fusion, Volume 35, 2017, Pages 132-147, ISSN 1566-2535, https://doi.org/10.1016/j.inffus.2016.10.001.
27. Ouyang H, Lin X, Li S, Gao L, Houssein, E.H, (2025) "Recent metaheuristic algorithms for multi-objective feature selection: review, applications, open issues and challenges, Cluster Computing, Springer 28:467 https://doi.org/10.1007/s10586-024-04996-1 ,- vol V)
28. Abdullah M, Balamash A, Alshannaq A, Almabdy S, "Enhanced intrusion detection system using feature selection method and ensemble learning algorithms", International Journal of Computer Science and Information Security (IJCSIS), Vol. 16, No. 2, February 2018, pp. 48-55
29. Preethi D, Khare N, "EFS-LSTM (Ensemble-Based Feature Selection With LSTM) Classifier for Intrusion Detection System", International Journal of e-Collaboration Volume 16 • Issue 4 • October-December 2020
30. Akhiat Y, Touchanti K, Zinedine A, Chahhou M, "IDS-EFS: Ensemble feature selection-based method for intrusion detection system", Multimedia Tools and Applications (2024) 83:12917–12937 https://doi.org/10.1007/s11042-023-15977-8

31. Ahuja, J., & Ratnoo, S. (2023, February). Population-Based Meta-heuristics for Feature Selection: A Multi-objective Perspective. In *Proceedings of International Conference on Data Science and Applications: ICDSA 2022, Volume 1* (pp. 243-264). Singapore: Springer Nature Singapore.
32. Chanu U.S, JohnsonSingh K, Chanu Y.J, "An ensemble method for feature selection and an integrated approach for mitigation of distributed denial of service attacks", Concurrency ComputatPractExper. 2022; JohnWiley&Sons,Ltd., https://doi.org/10.1002/cpe.6919.
33. Roopak M, Tian G.Y, Chambers J, "Multi-objective-based feature selection for DDoS attack detection in IoT networks", IET, The Institution of Engineering and Technology Netw., 2020, Vol. 9 Iss. 3, pp. 120-127
34. Sahu, R. K., & Patel, S. (2021). Hybrid approach for feature selection based on genetic algorithm and recursive feature elimination. Expert Systems with Applications, 168, 114341. https://doi.org/10.1016/j.eswa.2020.114341 (2021)
35. Vijayalakshmi S, Venkatesan V.P, "A Survey on Application of Metaheuristics Techniques for Ensemble Feature Selection (EFS)", Proceedings of the International Conference on Automation, Computing and Renewable Systems (ICACRS 2022) IEEE Xplore Part Number: CFP22CB5-ART: ISBN: 978-1-6654-6084-2
36. Bolón-Canedo V, Sánchez-Maroño N, Betanzos A.A, "Recent advances and emerging challenges of feature selection in the context of Big Data", Knowledge-Based Systems (2015), doi: http://dx.doi.org/10.1016/j.knosys.2015.05.014
37. Huang, D., Liu, Z., & Wu, D. (2024). Research on Ensemble Learning-Based Feature Selection Method for Time-Series Prediction. *Applied Sciences*, *14*(1), 40. https://doi.org/10.3390/app14010040
38. Wenhao H, Li H, Li J, "Ensemble Feature Selection for Improving Intrusion Detection Classification Accuracy", AICS 2019, Association for Computing Machinery. ACM ISBN 978-1-4503-7150-6/19/07, https://doi.org/10.1145/3349341.3349364.
39. Nssibi M, , Manita G , Korbaa O, (2023), Advances in nature-inspired metaheuristic optimization for feature selection problem: A comprehensive survey, Computer science review 49(2023) 100559, Elsevier, **https://doi.org/10.1016/j.cosrev.2023.100559**
40. Seijo-Pardo B, Bolón-Canedo V, Alonso-Betanzos A, "Testing Different Ensemble Configurations for Feature Selection", Neural Process Lett DOI 10.1007/s11063-017-9619-1, Springer Science+Business Media New York 2017
41. Bolón-Canedo V, Alonso-Betanzos A. Ensembles for feature selection: A review and future trends, Elsevier, [J]. Information Fusion, 2019, 52: 1-12.
42. Arora J, Agrawal U, Tiwari P, Gupta D, Khanna A, (2020), Ensemble Feature Selection Method Based on Recently Developed Nature-Inspired Algorithms, International Conference on Innovative Computing and Communications, Advances in Intelligent Systems and Computing 1087, https://doi.org/10.1007/978-981-15-1286-5_39
43. Pes B (2020), Ensemble feature selection for high-dimensional data: a stability analysis across multiple domains, Neural Computing and Applications (2020) 32:5951–5973 https://doi.org/10.1007/s00521-019-04082-3 ,- vol V)
44. Ahsan, M.S., Islam, S., & Shatabda, S. (2025). A Systematic Review of Metaheuristics-Based and Machine Learning-Driven Intrusion Detection Systems in IoT. *ArXiv, abs/2506.00377*.
45. Vijayalakshmi S, Venkatesan V.P, "Ameliorated/Accelerated Intrusion Detection System (AIDS) Using Multiattribute Foveat Analysis with Recurrent Neural Network Augmented by Behavior Pattern Profile (BPP)", Dogo Rangsang Research Journal, UGC Care Group I Journal, ISSN: 2347-7180, Vol-10, Issue-07, No. 16, July 2020.
46. Tawhid, M. A., & Hossain, M. S. (2021). Recent advances and application of metaheuristic algorithms: A survey (2014–2020). Archives of Computational Methods in Engineering, 28(5), 3389–3421. https://doi.org/10.1007/s11831-021-09559-4

47. Hussien, A. G., & Amin, M. (2023). A systematic review on metaheuristic optimization techniques for feature selections in disease diagnosis: Open issues and challenges. Expert Systems with Applications, 219, 119663. https://doi.org/10.1016/j.eswa.2023.119663
48. Sharma, R., & Kumar, A. (2023). A metaheuristic-based ensemble feature selection framework for cyber threat detection in IoT-enabled networks. Computer Networks, 237, 110015. https://doi.org/10.1016/j.comnet.2023.110015
49. Rahman, M. M., & Alghamdi, N. S. (2023). Ensemble feature selection using Bonferroni, OWA and Induced OWA aggregation operators. Expert Systems with Applications, 227, 120301. https://doi.org/10.1016/j.eswa.2023.120301
50. Aljarah, I., Mafarja, M., Heidari, A. A., & Mirjalili, S. (2021). A parallel metaheuristic approach for ensemble feature selection based on multi-core architectures. Knowledge-Based Systems, 222, 106992. https://doi.org/10.1016/j.knosys.2021.106992

AUTHORS PROFILE

Mrs. S. Vijayalakshmi M.C.A., M.Phil. graduate currently pursuing Ph.D. in Dept. of Banking Technology, Pondicherry University. Her research interest includes Artificial Intelligence, Cyber security, Deep Learning and applications of DL models in security engineering mainly on domains such as Intrusion/Anomaly Detection System. I have 15 years of teaching and research experience and have scholarly publications in international repute conferences and erudite blind peer reviewed journals

Dr. V. Prasanna Venkatesan, Professor, Dept. of Banking Technology, Pondicherry University has research thrust on domains like software architecture, service-oriented architecture, Business Intelligence, Smart Banking, Banking Technology. Twenty One scholars have successfully earned their Ph.D degree under his able guidance and support. He has 33 years of teaching and research experience to his credit. He has meticulously completed one project and has 160 publications in peer-reviewed international conferences and journals.