Anshul Arora

mohil1213

Assignment4

Document Details

Submission ID
trn:oid:::27535:139906640

Submission Date
May 21, 2026, 9:07 PM GMT+5:30

Download Date
May 21, 2026, 9:14 PM GMT+5:30

File Name
mohil1213.pdf

File Size
1.4 MB

42 Pages

9,808 Words

57,041 Characters

8% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Small Matches (less than 10 words)

Match Groups

30 Not Cited or Quoted 5%
Matches with neither in-text citation nor quotation marks

17 Missing Quotations 3%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

4% Internet sources
3% Publications
6% Submitted works (Student Papers)

Match Groups

- **30** Not Cited or Quoted 5%
  Matches with neither in-text citation nor quotation marks
- **17** Missing Quotations 3%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

4% Internet sources
3% Publications
6% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Student papers	
University of Warwick on 2010-08-24		<1%
2	Student papers	
University of Hertfordshire on 2026-01-05		<1%
3	Student papers	
Coventry University on 2026-03-28		<1%
4	Internet	
www.mdpi.com		<1%
5	Student papers	
Heriot-Watt University on 2025-04-18		<1%
6	Internet	
assets-eu.researchsquare.com		<1%
7	Internet	
www.avccengg.net		<1%
8	Internet	
jiwebtech.com		<1%
9	Student papers	
Delhi Technological University on 2026-05-10		<1%
10	Student papers	
Maulana Azad National Urdu University on 2026-04-30		<1%

11 Internet
odr.chalmers.se <1%

12 Internet
espace.library.uq.edu.au <1%

13 Student papers
South Bank University on 2026-03-04 <1%

14 Internet
docplayer.net <1%

15 Student papers
Glyndwr University on 2026-02-01 <1%

16 Internet
public-pages-files-2025.frontiersin.org <1%

17 Publication
H. A. El Shenbary, Amr T. A. Elsayed, Belal Z. Hassan, Khaled A. A. Khalaf Allah, Ah... <1%

18 Student papers
University of Ulster on 2026-05-04 <1%

19 Student papers
Alliance University on 2026-04-08 <1%

20 Student papers
The African Institute for Mathematical Sciences on 2025-06-15 <1%

21 Student papers
University of Bristol on 2026-05-04 <1%

22 Student papers
University of Essex on 2026-04-24 <1%

23 Internet
digitronicsinc.com <1%

24 Internet
livephysics.com <1%

25 Publication

Dileesh Chandra Bikkasani. "An Energy-Efficient Cascaded Machine Learning Fra... <1%

26 Student papers

King Abdulaziz University on 2026-05-14 <1%

27 Student papers

University of Portsmouth on 2026-05-20 <1%

28 Student papers

University of Reading on 2026-05-03 <1%

29 Student papers

BB9.1 PROD on 2026-05-07 <1%

30 Student papers

Brunel University on 2026-03-27 <1%

31 Student papers

Napier University on 2026-04-21 <1%

32 Student papers

University of Sussex on 2026-05-01 <1%

33 Internet

www.frontiersin.org <1%

34 Publication

Mohamed Saied, Shawkat Guirguis, Magda Madbouly. "Review of filtering based f... <1%

35 Student papers

Strathmore University (Main Account) on 2025-11-28 <1%

36 Student papers

Universiti Malaysia Pahang Al-Sultan Abdullah (UMPSA) on 2026-01-10 <1%

37 Student papers

University of Dundee on 2026-03-30 <1%

38 Student papers

University of The Gambia on 2023-03-28 <1%

39 Publication

Yakub Kayode Saheed, Oluwadamilare Harazeem Abdulganiyu, Taha Ait Tchakou... <1%

40 Internet

telcomatraining.com <1%

41 Internet

theses.hal.science <1%

1 Introduction

1.1 Overview

The digital transformation of the 21st century has been profoundly shaped by the emergence and rapid expansion of the Internet of Things (IoT)—a paradigm in which everyday physical objects are embedded with sensors, software, and connectivity to exchange data over the internet. From smart thermostats and wearable health monitors to industrial control systems and autonomous vehicles, IoT technology has permeated nearly every facet of modern society. Over the last decade, the IoT has evolved from simple, standalone sensors into a vast, interconnected ecosystem of smart devices. Between 2024 and 2026, the number of active IoT connections has skyrocketed, and it is estimated that there are close to 30 billion devices currently connected to networks. This growth marks a change from normal local computer systems to global connectivity, driven by the lowering cost of hardware and the rise of ultra-fast 5G/6G networks [1, 2].

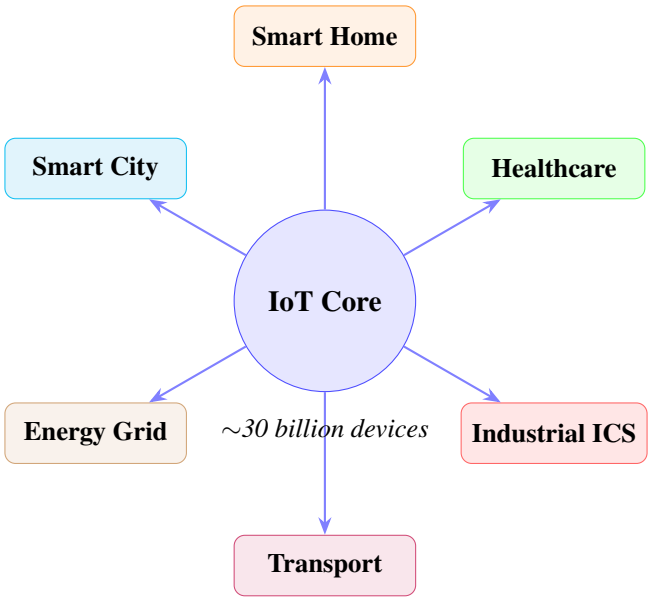

Figure 1.1: IoT ecosystem domains showing the centralised connectivity fabric that drives global hyperconnectivity. Each spoke represents a major application domain connected to a shared IoT infrastructure.

Today's era has seen a very fast rise in smart devices being supported by IoT technologies. By the start of 2026, the demand for smart home devices, industrial sensors, and healthcare monitoring systems has increased much, with thousands of new nodes added to the networks every month. From IoT networks the data transfers passed 80 zettabytes each year in

2025, enabling real-time patient monitoring in hospitals, precision agriculture, smart energy management, and logistics optimisation at a global scale [1, 2].

Although having these many advantages, this hyperconnectivity brings significant security risks, specially in the form of unapproved network intrusions. IoT devices normally operate with the help of networks and link directly with the physical world, which includes sensitive domains like home environments and industrial control systems. A substantial fraction of deployed IoT devices still operate with hardcoded credentials, unpatched firmware, or absent encryption—making them extremely vulnerable to attacks [1].

1.2 Background and Motivation

As the usage of Internet of Things (IoT) systems continues at a striking rate, network intrusions have become a common happening in today's world. Traditional network security tools—most notably signature-based Intrusion Detection Systems (IDS)—rely on the prior enumeration of known attack patterns. While effective against established threats, these systems are fundamentally reactive: they cannot detect novel attack vectors or zero-day exploits until corresponding signatures are manually created and distributed. Moreover, the sheer heterogeneity of IoT device types, communication protocols, and traffic patterns makes the maintenance of a comprehensive signature database practically infeasible.

Conventional network signatures do not work with the constantly changing attack vectors that modern attackers have developed. While dynamic network traffic analysis can be helpful, it comes with its own set of challenges because of its high computational complexity, which makes its adoption ineffective on resource-constrained IoT devices. Machine learning (ML) has emerged as the most promising alternative, enabling IDS to learn discriminative patterns from labelled traffic data and to generalise to previously unseen attack variants [2, 3]. Using this idea, this research implements a statistical feature selection technique—Cohen's d (CD), Mann–Whitney U (MW), and Kolmogorov–Smirnov (KS) test—then further uses a powerful machine learning classifier (XGBoost) to detect attacks optimally.

However, the application of ML to IoT network traffic introduces its own challenges. Network flow datasets are inherently high-dimensional, with dozens to hundreds of features describing temporal, volumetric, and protocol-level characteristics. Many of these features are either irrelevant to the binary distinction between benign and malicious traffic, or highly redundant with one another. Training machine learning models on such noisy, high-dimensional inputs leads to degraded generalisation, longer training times, and poor suitability for resource-constrained edge deployment.

Feature selection—the process of identifying and retaining the most informative subset of input features—is therefore a critical preprocessing step in any practical IoT IDS pipeline [4, 6]. Effective feature selection reduces computational overhead, mitigates overfitting, and enhances model interpretability, which is increasingly important in operational security

contexts.

1.3 Problem Statement

Despite a rich body of literature on feature selection for network intrusion detection, a clear gap exists in the rigorous, systematic comparison of multiple statistical feature selection criteria applied jointly to a recent, large-scale IoT-specific benchmark. Most existing studies employ a single statistical metric (e.g., information gain or chi-squared test) or rely entirely on wrapper-based or embedded methods. The key gap observed in the literature is the absence of a systematic, multitest composite approach for considering factors from the mean, rank, and distribution related views together on a recent IoT dataset. Earlier research often involves using no more than one statistical metric or even neglects statistical pre-processing as a whole, whereas combining several statistical metrics within one unified filter has not yet received due attention [29, 35].

Furthermore, many published studies neglect to prevent data leakage during feature selection—computing statistics or correlation matrices on the full dataset rather than exclusively on the training partition—which artificially inflates reported performance figures.

1.4 Contributions of This Work

This dissertation addresses the above gaps through the following original contributions:

1. **Composite Statistical Feature Selection Pipeline:** A novel pipeline is proposed combining Cohen's d (CD), Mann–Whitney U (MW), and Kolmogorov-Smirnov (KS) tests into a single composite scoring approach. Seven possible configurations are systematically evaluated—three individual tests (CD, MW, KS) and four composite combinations (CD+MW, CD+KS, MW+KS, CD+MW+KS)—for finding the optimal combination for detecting maximum accuracy on CIC-IIoT 2025.

2. **Optimum Feature Screening:** With the combination of multiple filters (CD+MW+KS), we can see that hugely weighted features can be taken out, which reduces the feature space to the top 40 attributes and by that means it ensures low computational delay.

3. **Construction of a Trusty Intrusion Detection Solution:** We used an advanced XGBoost classifier, which gives excellent detection performance, giving an accuracy of 99.05% and an F1-score of 99.18%.

4. **Analysis on Large-Scale Traffic Data:** We validated our approach precisely on the CIC-IIoT 2025 dataset having 685,671 real IoT traffic samples, also covering different attack groups including DDoS, reconnaissance, spoofing, and command injection.

5. **Leakage-Free Experimental Protocol:** All feature selection statistics are computed using only the training partition, providing unbiased estimates of true generalisation performance.

1.5 Thesis Structure

The remainder of this thesis is organised as follows. Chapter 2 provides theoretical background on IoT security, intrusion detection, and the statistical methods employed. Chapter 3 reviews related work across machine learning for IoT IDS, benchmark datasets, feature selection techniques, and statistical testing approaches. Chapter 4 describes the CIC-IIoT 2025 dataset and the preprocessing pipeline. Chapter 5 presents the proposed methodology in full detail. Chapter 6 reports and discusses the experimental results. Chapter 7 concludes the dissertation, and Chapter 8 outlines future research directions and social impact.

2 Background and Foundations

2.1 Internet of Things: Architecture and Security Challenges

The Internet of Things (IoT) refers to a network of physical devices—embedded with sensors, actuators, software, and connectivity—that collect and exchange data autonomously. A typical IoT architecture consists of three tiers:

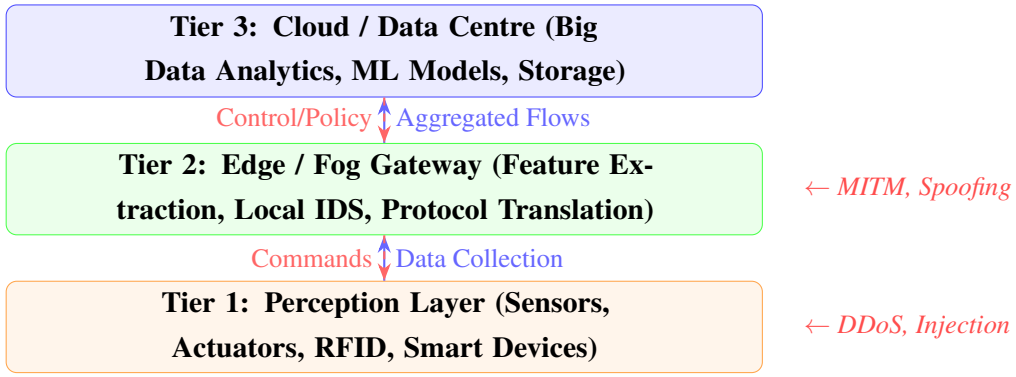

Figure 2.1: Three-tier IoT architecture with representative attack surfaces at each layer. IDS can be deployed at the gateway (Tier 2) to monitor flows from perception devices without burdening individual nodes.

Unlike conventional computing systems, IoT devices are typically resource-constrained, often lacking the processing power, memory, and energy budget required to implement robust security mechanisms. The consequences of this security gap are severe. Cybercriminals and state-sponsored attackers routinely exploit vulnerable IoT endpoints to launch DDoS attacks of unprecedented scale, to establish persistent botnets, to exfiltrate sensitive personal or industrial data, and to perform lateral movement across enterprise networks [1, 3].

2.2 Intrusion Detection Systems

An Intrusion Detection System (IDS) monitors network traffic or host activity to identify anomalous or malicious behaviour. IDS approaches are broadly classified into *signature-based* and *anomaly-based* systems [39]. Signature-based systems match observed traffic patterns against a database of known attack signatures, offering high precision but zero coverage of novel attacks. Anomaly-based systems, which includes ML-based approaches,

learn a model of normal behaviour and flag deviations—at the cost of potentially higher false positive rates.

A Network IDS (NIDS) monitors traffic at strategic network points, providing broad coverage with low per-device overhead. In IoT environments, network-level monitoring is generally preferred due to the resource constraints of individual devices [10].

2.3 Feature Representation of Network Traffic

Machine learning-based IDS operate on feature vectors derived from raw network packets or aggregated flow records. Flow-level features are computed over all packets belonging to a single network conversation (identified by source/destination IP, port, and protocol). They encapsulate statistical summaries of packet arrival times, inter-packet intervals, payload sizes, TCP flag counts, TTL values, window sizes, and port diversity metrics [14]. The CIC-IIoT 2025 dataset provides 95 such precomputed flow-level features, which serve as the input space for the proposed framework.

2.4 Feature Selection: Paradigms and Motivation

Feature selection methods are classified into three paradigms [35]:

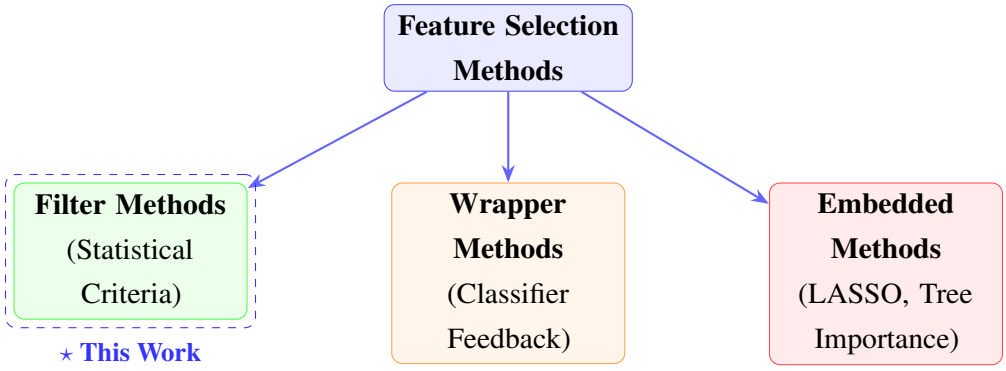

Figure 2.2: Taxonomy of feature selection paradigms. The proposed framework belongs to the filter category, using three complementary statistical tests to score features independently of any classifier.

This dissertation focuses on filter methods due to their computational efficiency, absence of overfitting risk during selection, and generalisability across classifiers [7, 9].

2.5 Statistical Feature Selection Methods

2.5.1 Cohen's d Effect Size

Cohen's d [17] measures the standardised difference between the means of two groups, providing a scale-independent measure of practical significance. For a binary classification feature f, let μ_1 and μ_0 denote the class means for the attack and benign classes, respectively, and let s_p denote the pooled standard deviation:

$$s_p = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_0 - 1)s_0^2}{n_1 + n_0 - 2}}, \qquad d = \frac{|\mu_1 - \mu_0|}{s_p} \tag{2.1}$$

Unlike p-values, Cohen's d is scale-independent and does not suffer from artificial statistical significance at large sample sizes—this characteristic being important when working with hundreds of thousands of flow records. A feature with $d > 0.8$ is conventionally considered to exhibit a large effect size.

2.5.2 Mann–Whitney U Test

The Mann–Whitney U test [18] is a non-parametric rank-based test that assesses whether the distribution of a feature is stochastically larger in one class than the other, without assuming normality—well-suited to network flow attributes that commonly exhibit heavy-tailed or skewed distributions:

$$U = n_1 n_0 + \frac{n_1(n_1 + 1)}{2} - R_1 \tag{2.2}$$

where n_1, n_0 are class sample sizes and R_1 is the rank sum for the attack class. Features are ranked by their normalised U statistic (Mann–Whitney effect size $r = U/(n_1 n_0)$).

2.5.3 Kolmogorov–Smirnov Test

The two-sample Kolmogorov–Smirnov (KS) test [19] quantifies the maximum absolute distance between the empirical cumulative distribution functions (ECDFs) of two groups:

$$D = \sup_x |F_1(x) - F_0(x)| \tag{2.3}$$

A large D statistic indicates that a feature's distribution differs substantially between traffic classes—a property independent of mean differences captured by Cohen's d. Features are ranked by descending D.

2.5.4 Composite Scoring

Each feature receives a normalized score from each test (min–max scaled to $[0, 1]$). The composite score is the unweighted mean of the k active normalized scores:

$$\text{score}(f) = \frac{1}{k} \sum_{i=1}^{k} s_i(f), \quad s_i(f) \in \left\{ \hat{d}(f),\ \hat{r}_{mw}(f),\ \hat{D}_{ks}(f) \right\} \tag{2.4}$$

where $k \in \{1, 2, 3\}$ depending on the filter combination.

2.5.5 Pearson Correlation-Based Redundancy Pruning

Following statistical ranking, redundant features are removed using Pearson correlation-based greedy forward selection. A feature is admitted to the final subset only if its maximum absolute Pearson correlation with all previously selected features is below a threshold τ [7, 23]:

$$C_{i,j} = \frac{\text{cov}(X_i, X_j)}{\sigma_{X_i} \sigma_{X_j}} \tag{2.5}$$

2.6 Classification Models

2.6.1 Gradient Boosted Trees: XGBoost and LightGBM

XGBoost [21] and LightGBM [22] are gradient-boosting frameworks that sequentially add decision tree base learners to minimise a regularised objective function. XGBoost employs column subsampling, L_1/L_2 regularisation, and second-order gradient statistics. LightGBM extends this with histogram-based leaf splitting and gradient-based one-side sampling (GOSS), substantially reducing memory and compute requirements. Both have achieved state-of-the-art performance across numerous intrusion detection benchmarks [11].

2.6.2 Random Forest and Extra Trees

Random Forest [20] is a bagging ensemble that trains multiple decision trees on bootstrap samples of the training data, using random feature subsets at each split. Extra Trees [25] further randomises split thresholds. Both models are robust to irrelevant features and handle non-linear class boundaries effectively.

2.6.3 Logistic Regression and Multi-Layer Perceptron

Logistic Regression provides a linear baseline by modelling the log-odds of class membership as a linear combination of input features [24]. The Multi-Layer Perceptron (MLP) is

a feed-forward neural network capturing complex non-linear feature interactions through hidden layers with ReLU activations [37].

3 Related Work

3.1 Machine Learning for IoT Intrusion Detection

Machine learning has emerged as the dominant paradigm for IoT intrusion detection because signature-based systems cannot generalise to novel or zero-day attack vectors. Meneghello et al. [1] surveyed practical security vulnerabilities across a broad range of commercial IoT device categories—from smart home sensors to industrial controllers—establishing the scale and heterogeneity of the IoT attack surface and underscoring why static rule-based defences are insufficient for dynamic threat environments. Al-Garadi et al. [2] provided a comprehensive survey of machine learning and deep learning methods applied to IoT security, systematically cataloguing challenges including class imbalance, limited labelled data, and high feature dimensionality—the same challenges that the present work's statistical feature selection stage is specifically designed to address.

Rashid et al. [3] applied stacking ensemble techniques to IoT cyberattack detection in smart city networks and showed that combining multiple heterogeneous base learners via a meta-learner significantly improves detection accuracy across diverse attack categories, directly motivating the ensemble-heavy classifier suite evaluated in the present work. Gyenizse et al. [11] conducted a direct comparative study of classifiers on modern IoT datasets and demonstrated that XGBoost and LightGBM consistently outperform traditional approaches—including SVM, Naïve Bayes, and k-NN—providing the primary empirical rationale for privileging gradient-boosted trees in the present classifier suite. Abdulganiyu et al. [10] reviewed over 150 anomaly-based, signature-based, and hybrid IDS studies and identified feature quality and high dimensionality as the most consistently limiting factors for IDS generalisation, directly motivating a principled dimensionality reduction stage prior to classification.

Chimphlee and Chimphlee [37] demonstrated that a Multi-Layer Perceptron trained on statistically selected features achieves competitive intrusion detection performance on the CIC-IDS2018 dataset in both binary and multi-class settings, providing direct empirical support for the inclusion of MLP as the non-linear neural baseline in the present comparative evaluation. Bammidi et al. [32] evaluated top-K feature selection paired with XGBoost, LightGBM, and Random Forest on an IoT intrusion detection benchmark—an experimental design nearly identical to the present work—and confirmed that a compact statistically selected feature subset of comparable cardinality to ours achieves peak performance without sacrificing detection accuracy.

3.2 Benchmark Datasets for IoT Intrusion Detection Research

The credibility of any IDS evaluation depends on the methodological integrity and realism of its benchmark dataset. Sharafaldin et al. [31] established the foundational methodology for generating CICFlowMeter-based IDS benchmarks: their framework defines principles for realistic traffic profiling, diverse attack scripting, and automated extraction of over 80 bidirectional flow-level statistics—temporal, volumetric, and protocol-level aggregates—from raw packet captures. This methodology is the direct methodological ancestor of the CIC-IIoT 2025 dataset used in the present work: CIC applied the same CICFlowMeter-based pipeline to generate all 95 features that constitute our feature selection problem. Citing Sharafaldin et al. directly explains why the feature space takes the form it does and why statistical filter methods are necessary to eliminate the redundant statistical aggregates that CICFlowMeter produces by design.

41 Tavallaee et al. [26] conducted a landmark analysis of the KDD CUP 99 dataset that exposed systematic artefacts—including duplicate records and implicitly leaked class information—causing many early IDS studies to report artificially inflated, non-reproducible performance. This finding directly motivates the strict leakage-free experimental protocol adopted in the present work. Bouke and Abdullah [12] extended this concern to the preprocessing stage, empirically demonstrating across six ML models and three established IDS datasets that applying standardisation or imputation before train/test splitting causes measurable, reproducible accuracy inflation—confirming that even recent IDS evaluations remain vulnerable to preprocessing leakage if scaling parameters are not fitted exclusively on the training partition.

The UNSW-NB15 dataset [14] was among the first IoT-oriented benchmarks to provide engineered flow-level features for evaluating anomaly-based IDS across diverse attack scenarios, establishing key baselines for comparative research. The Bot-IoT dataset [15] captured realistic botnet behaviour from physical IoT devices under Mirai and BASHLITE variants, making it a standard botnet traffic benchmark. The CICIoT2023 dataset [13] extended the landscape to 33 attack types across seven categories. The CIC-IIoT 2025 dataset [16], used in the present work, represents the current state of the art: generated from real physical IoT devices under carefully scripted attacks—DDoS, reconnaissance, spoofing, worm propagation, command injection, and hybrid variants—with no temporal or label leakage by construction, and with the 95 CICFlowMeter features whose redundancy and dimensionality the present pipeline is designed to address.

3.3 Feature Selection in Intrusion Detection

Feature selection is a critical preprocessing step in any practical IDS because high-dimensional flow data contains many irrelevant or mutually redundant features that degrade classifier generalisation and increase inference latency. Guyon and Elisseeff [35] established the canonical filter, wrapper, and embedded taxonomy and showed that filter methods—scoring features independently of any classifier using statistical criteria—are preferred in high-dimensional settings for their computational efficiency, freedom from classifier bias, and consistent generalisability across learners. This theoretical grounding directly underpins the choice of a filter-only scoring stage in the present pipeline.

Almohaimeed [4] demonstrated that a single statistical criterion applied to TCP-level IoT flow features suffices for lightweight IDS but simultaneously highlighted that single-criterion selection systematically misses complementary discriminative information carried by orthogonal feature statistics—the precise limitation that the present composite CD+MW+KS design is formulated to overcome. Zhou et al. [6] showed that combining a statistical feature scoring step with an ensemble classifier produces measurably higher IDS accuracy than either component alone, directly motivating the present two-stage filter-then-classify pipeline. Haque et al. [5] showed that entropy-based feature scoring paired with Random Forest detects distributional anomalies in IoT network flows and improves accuracy across multiple attack categories, further validating the statistical-filter paradigm. Santos et al. [9] demonstrated across multiple benchmarks that applying information-theoretic filtering before ensemble classification consistently improves IDS performance, directly motivating the use of multiple complementary statistical criteria as orthogonal scoring filters.

Hall [7] established the theoretical foundation for Pearson correlation-based redundancy removal, underpinning the greedy pruning step of the present pipeline. Thaseen et al. [23] confirmed empirically that Pearson correlation-based feature selection reduces IDS dimensionality without degrading detection accuracy, directly validating the pruning design in Section 2.4. Shukla et al. [8] proved empirically that Kolmogorov–Smirnov-based filtering eliminates uninformative features from network flow data, directly validating the KS component of the composite scoring stage. Andresini et al. [29] conducted a comprehensive critical review of supervised feature selection for network intrusion detection and found that combining statistical criteria with ensemble classifiers consistently outperforms single-criterion or single-classifier approaches—the central empirical premise of the present work.

Walling and Lodh [27] demonstrated that a statistical hybrid feature selection strategy combining multiple scoring methods with ML classifiers directly improves IDS accuracy on IoT networks, making their work the closest published analogue to the composite CD+MW+KS design of the present framework. Wang et al. [30] proposed a multi-criteria feature selection model for IoT intrusion detection that simultaneously applies several evaluation criteria to rank features, confirming that multi-criteria FS outperforms single-criterion

approaches for IoT IDS and directly corroborating the composite filter design of this work. Albulayhi et al. [34] proposed an entropy-based filter FS method that reduced 77 IoT features to 28 and achieved 99.98% detection accuracy, confirming that aggressive statistical dimensionality reduction is the primary determinant of IoT IDS performance. Janane et al. [33] addressed filter-based feature selection for high-dimensional classification data and provided theoretical grounding for multi-criteria filter ranking and composite scoring that directly supports the methodology formalised in Chapter 2. Kamalov et al. [28] proposed a filter method achieving 99.9% DDoS detection accuracy for IDS, directly paralleling and validating the filter-first approach adopted here. Balhareth and Ilyas [40] combined filter-based feature selection with XGBoost and CatBoost for IoMT intrusion detection and confirmed that filter FS paired with gradient-boosted tree classifiers achieves state-of-the-art performance—directly validating the two core design choices of the present framework.

3.4 Statistical Tests and Evaluation Metrics

The three statistical tests forming the core of the proposed composite filter each have a well-established theoretical foundation that justifies their inclusion. Cohen [17] introduced the standardised mean difference effect size (d) as a scale-independent measure of practical significance that is robust to p-value inflation under large sample sizes—a critical property when computing feature discriminability scores over the 685,671 IoT flow records in the present dataset. Mann and Whitney [18] formulated the non-parametric rank-sum test assessing stochastic dominance between two distributions without normality assumptions, making it well-suited to the heavy-tailed and skewed feature distributions characteristic of network traffic data, where parametric tests routinely over-reject. Kolmogorov [19] defined the maximum absolute distance between empirical cumulative distribution functions as a distribution-shape statistic that captures global class separation independently of mean or rank differences, thereby providing orthogonal discriminative information that complements both Cohen's d and the Mann-Whitney U in the composite ranking.

Peres [38] provided modern theoretical justification for normalised effect sizes in non-parametric tests, specifically deriving the $r = U/(n_1 n_0)$ formula used to normalise the Mann-Whitney U statistic into the $[0, 1]$ scoring range applied in Section 5.2. Santos et al. [9] showed that statistical feature filtering consistently improves IDS ensemble performance across multiple benchmarks, validating the filter-before-classify paradigm. Saito and Rehmsmeier [36] demonstrated that precision-recall analysis is more informative than ROC analysis for imbalanced classifiers, justifying the use of macro-weighted F1-score as the primary evaluation metric alongside AUC-ROC in the present work—where the benign/attack class ratio reaches approximately 40:60. Khraisat et al. [39] surveyed IDS techniques and open challenges across the literature and explicitly identified the absence of a systematic multi-criteria statistical testing approach for feature selection as an unaddressed

gap—precisely the gap that the proposed CD+MW+KS framework is designed to fill.

3.5 Summary and Research Gap

The literature consistently confirms that gradient-boosted tree ensembles are the most effective classifiers for IoT intrusion detection [11, 24], and that filter-based feature selection is the most practical preprocessing approach for high-dimensional network flow data [29, 35]. However, four specific gaps remain unaddressed:

- Most existing studies apply a single statistical feature selection criterion [4, 5], and thereby miss the complementary discriminative information captured simultaneously by mean-difference (Cohen's d), rank-based (Mann-Whitney U), and distribution-shape (KS) statistics.
- No published work has systematically compared all seven filter configurations arising from three statistical tests—three individual and four composite combinations—on the CIC-IIoT 2025 dataset [16], leaving the empirically optimal configuration for this benchmark unidentified.
- Many published IDS accuracy figures are confounded by preprocessing data leakage, where scaling parameters or imputation statistics are computed on the full dataset rather than exclusively on the training partition [12, 26]—rendering reported results unreliable indicators of true out-of-sample generalisation.
- The theoretical complementarity of Cohen's d, Mann-Whitney U, and Kolmogorov-Smirnov tests as a composite filter for IoT network traffic has not been formally articulated or empirically verified on a modern, large-scale IoT benchmark [17–19].

The proposed framework addresses all four gaps through a principled, leakage-free composite filter that systematically evaluates all seven configurations of the three statistical tests on the most recent large-scale IoT benchmark.

4 Dataset and Preprocessing

4.1 The CIC-IIoT 2025 Dataset

The Canadian Institute for Cybersecurity Industrial IoT 2025 (CIC-IIoT 2025) dataset [16] is one of the most recent and comprehensive benchmarks available for IoT intrusion detection research. It was collected in a controlled laboratory environment using a diverse array of physical IoT devices, including smart home sensors, IP cameras, smart thermostats, door locks, and industrial controllers operating under both normal conditions and during precisely scripted cyberattack scenarios. The CIC-IIoT 2025 dataset, of the Canadian Institute for Cybersecurity, is one of the very latest and a standard for IoT-specific intrusion detection research.

4.1.1 Attack Categories

The dataset encompasses the following attack categories:

- **DDoS attacks** (multiple sub-types): UDP flood, TCP SYN flood, HTTP flood, and amplification attacks.
- **Reconnaissance / Scanning**: port scans, OS fingerprinting, and host discovery.
- **Spoofing**: ARP spoofing and DNS spoofing.
- **Worm propagation**: self-replicating malware traversal across the IoT network.
- **Command injection**: unauthorised command execution through vulnerable device APIs.
- **Hybrid attacks**: coordinated combinations of two or more attack types.

4.1.2 Dataset Statistics

The raw dataset comprises 685,671 labelled flow-level instances across 95 computed features. Each instance represents a single bidirectional network flow, characterised by temporal, volumetric, and protocol-level attributes automatically computed from raw packet captures using CICFlowMeter.

Table 4.1: CIC-IIoT 2025 Dataset Summary

Property	**Value**
Total instances	685,671
Raw features	95
Attack instances	252,282
Benign instances	179,322
Instances after preprocessing	431,604
Training set size (75%)	323,703
Test set size (25%)	107,901

4.2 Preprocessing Pipeline

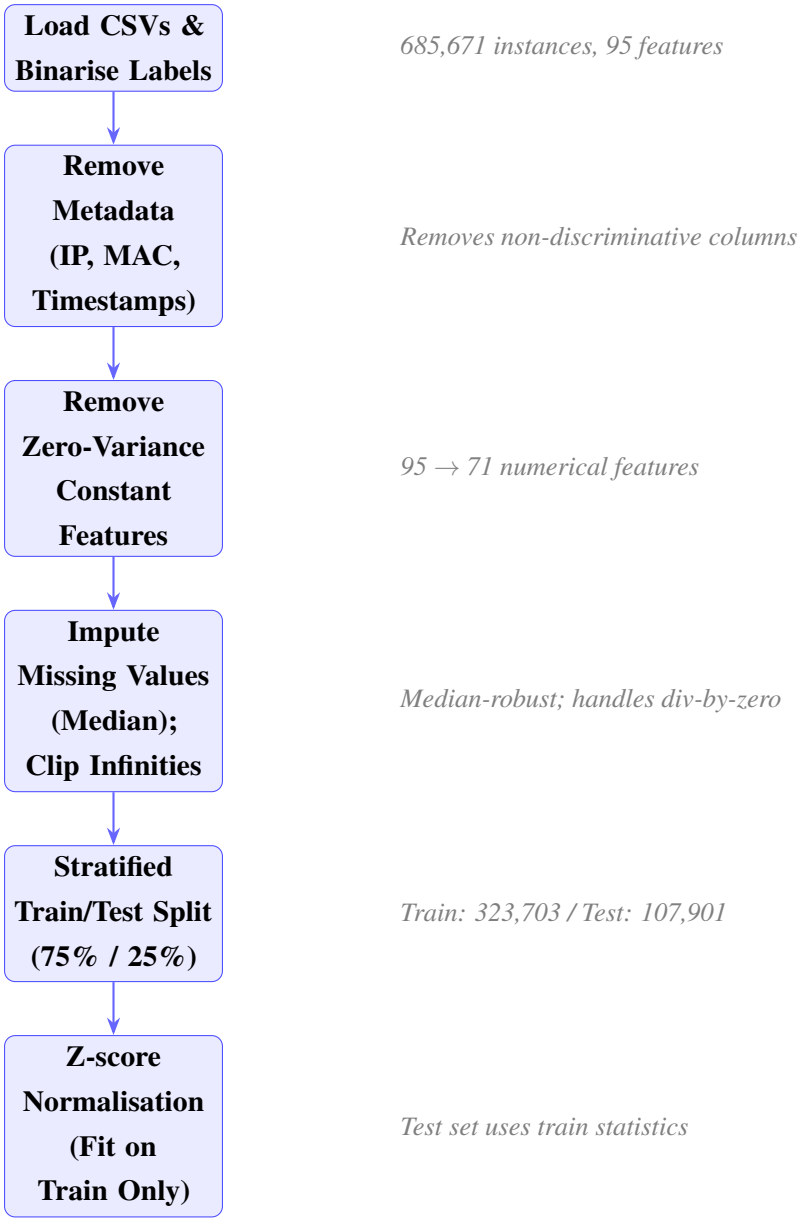

Figure 4.1: Detailed preprocessing pipeline for the CIC-IIoT 2025 dataset. Each stage is applied sequentially to produce a clean, leakage-free input for the statistical feature selection stage.

4.2.1 Data Loading and Label Binarisation

All benign and attack CSV files are loaded into a unified Pandas dataframe. Class labels are binarised: benign traffic is assigned label 0, and attack traffic is assigned label 1. This binary formulation is consistent with the detection task of distinguishing normal from anomalous IoT network behaviour.

4.2.2 Metadata Removal

Features encoding device-level metadata—device names, MAC addresses, IP addresses, and timestamps—are removed as they provide no generalisable discriminative information about network behaviour and would introduce spurious patterns specific to the collection environment.

4.2.3 Zero-Variance and Constant Feature Removal

Features with near-zero variance (standard deviation below 10^{-6}) and constant features are discarded. This step reduces the initial 95 features to 71 numerical candidates.

4.2.4 Missing Value Imputation and Outlier Handling

Remaining missing values are imputed using the per-feature median, which is robust to the heavy-tailed distributions characteristic of network flow data. Infinite values (arising from division by zero in some flow-level computations) are truncated to the per-feature maximum of finite values.

4.2.5 Train/Test Split and Scaling

A stratified train/test split (75%/25%) is performed *before* any feature-dependent computation to prevent data leakage. Z-score normalisation is applied using the training set mean and standard deviation:

$$\tilde{x}_i = \frac{x_i - \hat{\mu}_{\text{train}}}{\hat{\sigma}_{\text{train}}} \tag{4.1}$$

28 The scaler parameters are fitted exclusively on the training partition and applied to the test set.

4.2.6 Data Leakage Prevention

A critical aspect of the experimental protocol is strict data leakage prevention. All feature selection statistics (Cohen's d, Mann–Whitney U, KS test scores) are computed using only the training partition. The Pearson correlation matrix used for redundancy pruning is also computed on the training set. This protocol ensures that the reported performance metrics on the held-out test set are unbiased estimates of true generalisation performance—a standard that is not always followed in published IDS literature [26].

4

5 Proposed Methodology

5.1 Overview of the Framework

The proposed research is concerned with the development of an IoT intrusion detection system based on statistical feature selection and machine learning techniques. The process consists of five stages: (1) data loading and labeling, (2) data preprocessing, (3) statistical discriminative filtering using Cohen's d (CD), Mann–Whitney U (MW), and Kolmogorov–Smirnov (KS) tests, (4) correlation-based redundancy reduction, and (5) ML model training and evaluation.

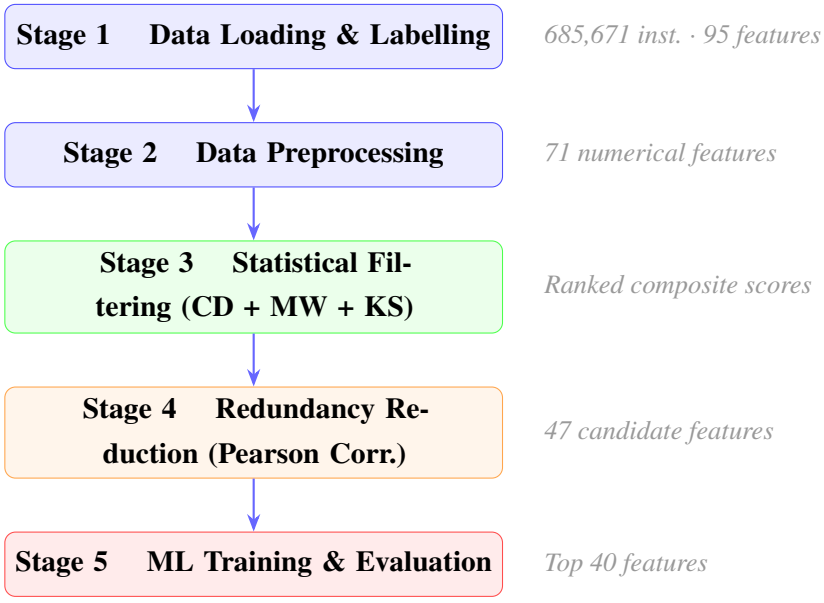

Figure 5.1: End-to-end pipeline of the proposed Hybrid Statistical Feature Selection Framework for IoT Intrusion Detection. Feature counts at each stage are shown to the right of the corresponding block.

5.2 Stage 1: Statistical Discriminability Scoring

The core novelty of this work is the composite statistical filter. Three complementary tests are applied to quantify the discriminative power of each feature with respect to the binary classification task.

5.2.1 Cohen's d Effect Size

For each feature f, the attack-class subsample $\mathcal{S}_1 = \{x_i : y_i = 1\}$ and the benign-class subsample $\mathcal{S}_0 = \{x_i : y_i = 0\}$ are extracted from the training set only. The pooled standard deviation and Cohen's d are computed as in Equation (2.1). While p-values are scale-dependent and prone to statistical significance even for small variations in large samples, d is independent of scale and resistant to big sample sizes, which is important while dealing with hundreds of thousands of flow records. A feature's normalised Cohen's d score $\hat{d}(f)$ is obtained by min-max scaling all per-feature d values to $[0, 1]$.

5.2.2 Mann–Whitney U Test

For each feature f, all values from $\mathcal{S}_0$ and $\mathcal{S}_1$ are jointly ranked. The U statistic is computed as in Equation (2.2), and the normalised effect size is $r_{mw} = U/(n_1 n_0)$, which lies in $[0, 1]$. The Mann-Whitney U Test is a non-parametric statistical test used to compare two samples or groups. It makes no assumption of normality—well-suited to network flow attributes that commonly exhibit heavy-tailed or skewed distributions. A value of r_{mw} close to 0 or 1 indicates strong stochastic dominance of one class over the other.

5.2.3 Kolmogorov–Smirnov Test

For each feature f, the empirical CDFs of the attack and benign training samples are computed, and the KS statistic D is computed as in Equation (2.3). A large D statistic indicates that the feature's distribution differs substantially between traffic classes—a property independent of mean differences captured by Cohen's d. All D values are then min-max normalised to $[0, 1]$ to yield $\hat{D}_{ks}(f)$.

5.2.4 Composite Scoring

Each feature receives a composite score computed as the unweighted mean of the normalised individual scores across the k active filters:

$$\text{score}(f) = \frac{1}{k} \sum_{i=1}^{k} s_i(f), \quad s_i(f) \in \left\{ \hat{d}(f),\ \hat{r}_{mw}(f),\ \hat{D}_{ks}(f) \right\} \tag{5.1}$$

For the full triple combination, $k = 3$ and $\text{score}(f) = \frac{1}{3}(\hat{d}(f) + \hat{r}_{mw}(f) + \hat{D}_{ks}(f))$. Seven filter sets are considered, comprising three independent tests (CD only, MW only, KS only) and four combined pairs (CD+MW, CD+KS, MW+KS, CD+MW+KS). The ideal top-N number of features is found through a range of $N \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 47\}$.

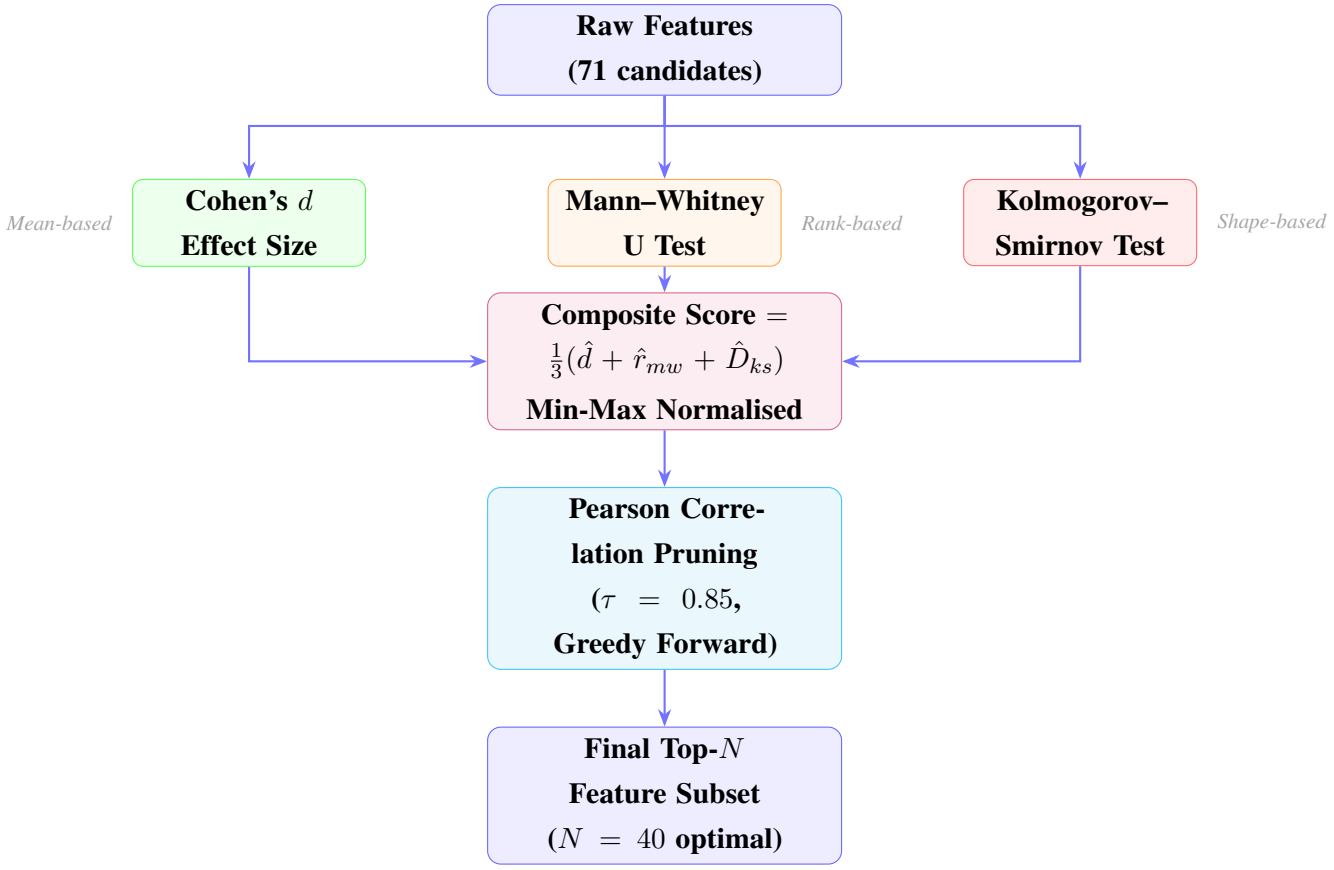

Figure 5.2: Detailed statistical feature selection pipeline. Three complementary statistical tests capture mean-based, rank-based, and distributional-shape perspectives on feature discriminability, which are aggregated into a composite score before redundancy pruning.

5.3 Stage 2: Correlation-Based Redundancy Pruning

Following composite scoring, the Pearson correlation matrix is calculated using the training set. Greedy forward selection is then applied based on score ranking, wherein a feature is selected only when its maximum absolute Pearson correlation with all previously selected features falls below the threshold $\tau = 0.85$:

$$f^* \leftarrow \text{admit } f \iff \max_{g \in \mathcal{S}} |C_{f,g}| < 0.85 \tag{5.2}$$

This process ensures that redundant pairs of features encoding similar network phenomena are removed from the pool of candidates, resulting in 47 features per filter combination [7]. The correlation matrix is computed exclusively on the training set to prevent data leakage.

5.4 Stage 3: Feature Count Sweep

For each of the seven filter configurations, the post-pruning feature list is truncated at each of the following feature counts: $N \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 47\}$. For each (N, filter) combination, all six classifiers are trained on the corresponding N-feature training set and evaluated on the N-feature test set. This sweep enables identification of the optimal (N^*, filter^*) pair.

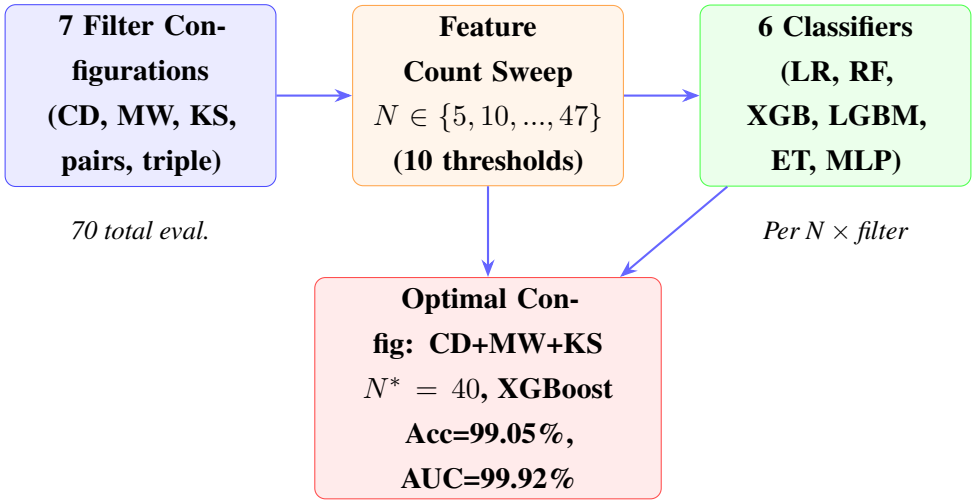

Figure 5.3: Experimental design for the feature count threshold sweep across all filter configurations and classifiers.

5.5 Classification Models and Evaluation Metrics

5.5.1 Models

Six classifiers are evaluated as follows [20–22, 25]:

- **Logistic Regression (LR)**: linear baseline; max iterations 1000; L_2 regularisation.
- **Random Forest (RF)**: 100 estimators; Gini impurity; max features = $\sqrt{p}$.
- **XGBoost (XGB)**: 100 estimators; max depth 6; learning rate 0.1; $L_1 + L_2$ regularisation.
- **LightGBM (LGBM)**: 100 estimators; leaf-wise growth; histogram binning.
- **Extra Trees (ET)**: 100 estimators; fully randomised split thresholds.
- **MLP**: two hidden layers (256, 128 neurons); ReLU activations; Adam optimiser.

All models are evaluated on the held-out 25% test set (107,901 samples). Reported metrics are accuracy, macro-weighted F1-score, and AUC-ROC.

5.5.2 Evaluation Metrics

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.3}$$

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN} \tag{5.4}$$

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5.5}$$

The AUC-ROC metric measures the area under the Receiver Operating Characteristic curve, providing a threshold-independent measure of classifier discrimination ability [36].

5.6 Implementation Details

The full pipeline is implemented in Python using the following libraries: `numpy`, `pandas`, `scipy.stats` (for MW and KS tests), `scikit-learn` (preprocessing, RF, ET, LR, MLP), `xgboost`, and `lightgbm`. All experiments are conducted on a MacBook with Apple M-series processor. The random seed is fixed at 42 for all stochastic operations to ensure reproducibility. For all experiments, a stratified sampling of 75% training and 25% test is performed; the feature selection correlation matrix is computed solely on the training data partition to avoid data leakage.

6 Results and Discussion

6.1 Preprocessing Summary

After metadata removal and zero-variance feature elimination, the 95-feature raw input was reduced to 71 numerical candidates. Following composite statistical scoring and Pearson correlation-based redundancy pruning (with $\tau = 0.85$), each filter configuration yielded up to 47 candidate features, from which the optimal subset is determined by the feature count sweep. The subsequent threshold sweep identifies 40 features as the peak-performance point for XGBoost and Random Forest under the CD+MW+KS composite.

6.2 Statistical Feature Evaluation Visualisations

Figure 6.3 presents the individual statistical scores across the three evaluation metrics for the top features from the CIC-IIoT 2025 dataset. The three panels display Cohen's d effect sizes (blue), Mann–Whitney U normalised effect sizes (green), and Kolmogorov–Smirnov statistics (orange) for the 20 highest-ranked features. Features such as `network_time-delta_avg` and `network_packets_all_count` consistently score highly across all three metrics, confirming their discriminative power from mean-based, rank-based, and distributional perspectives simultaneously.

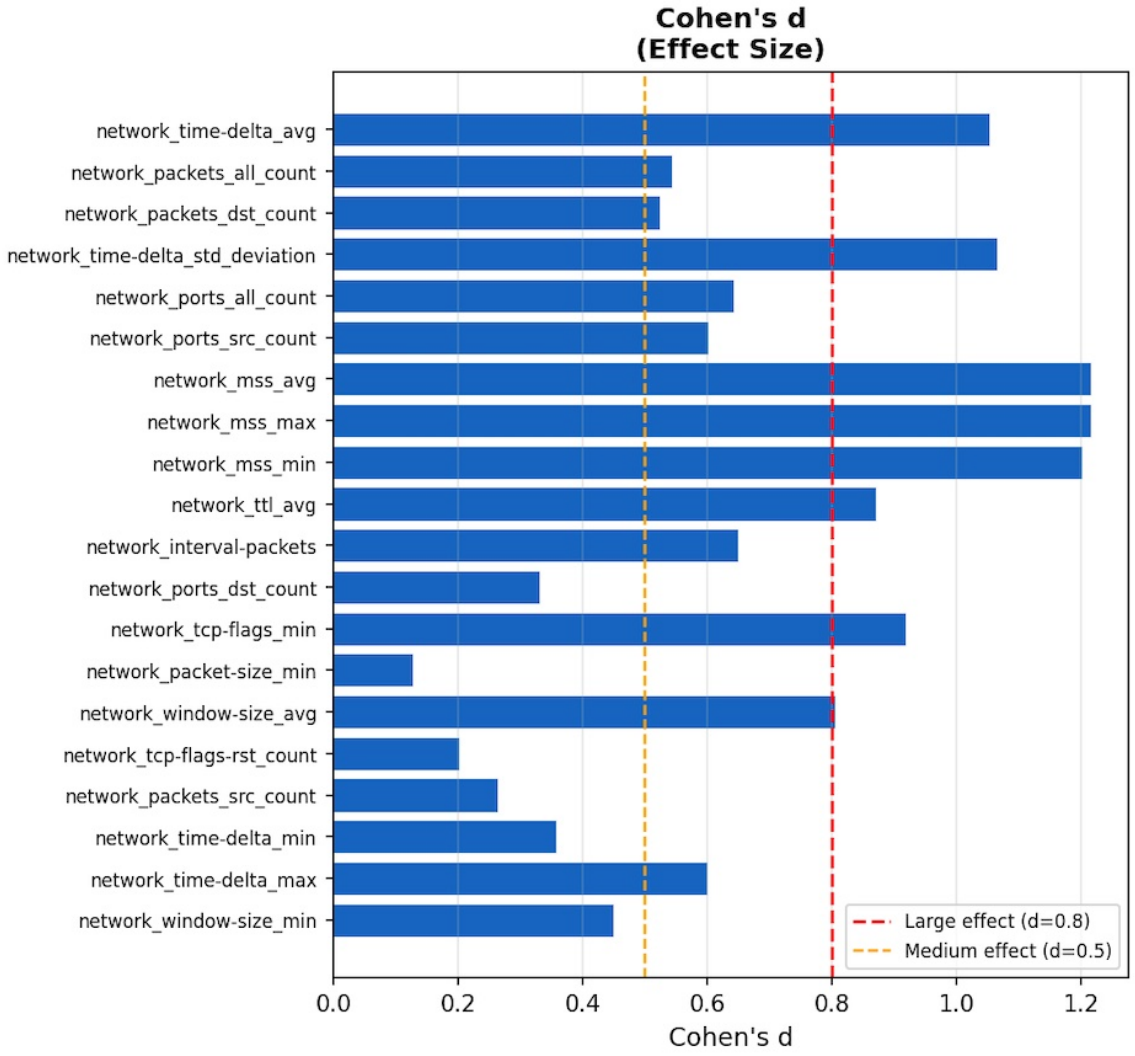

Figure 6.1: Statistical Feature Evaluation on CIC-IIoT 2025 — Cohen's d effect sizes (blue) for the top 20 features. Dashed lines indicate large ($d = 0.8$, red) and medium ($d = 0.5$, orange) effect size thresholds.

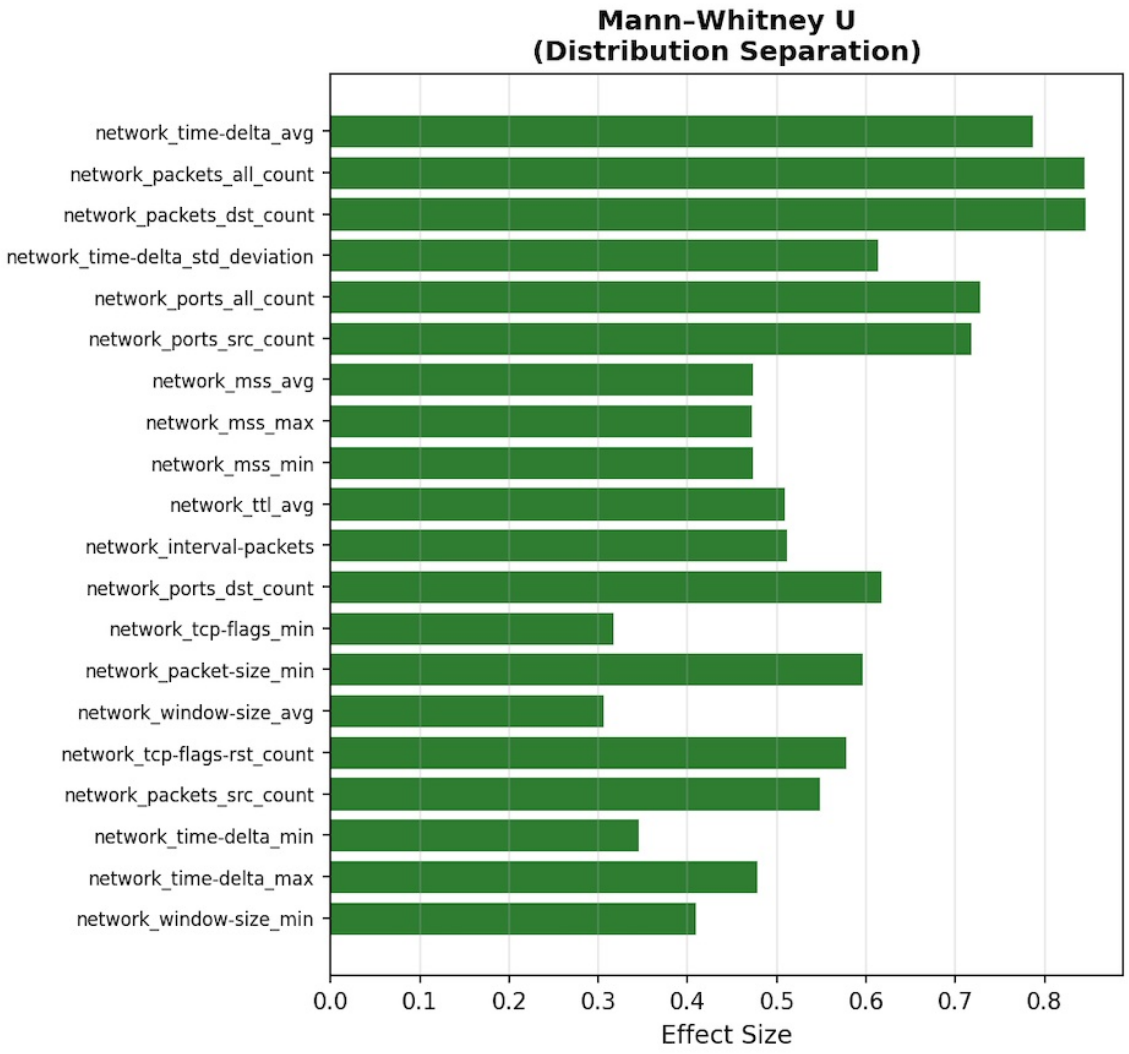

Figure 6.2: Statistical Feature Evaluation on CIC-IIoT 2025 — Mann–Whitney U normalised effect sizes (green) for the top 20 features, measuring rank-based distribution separation between benign and attack traffic.

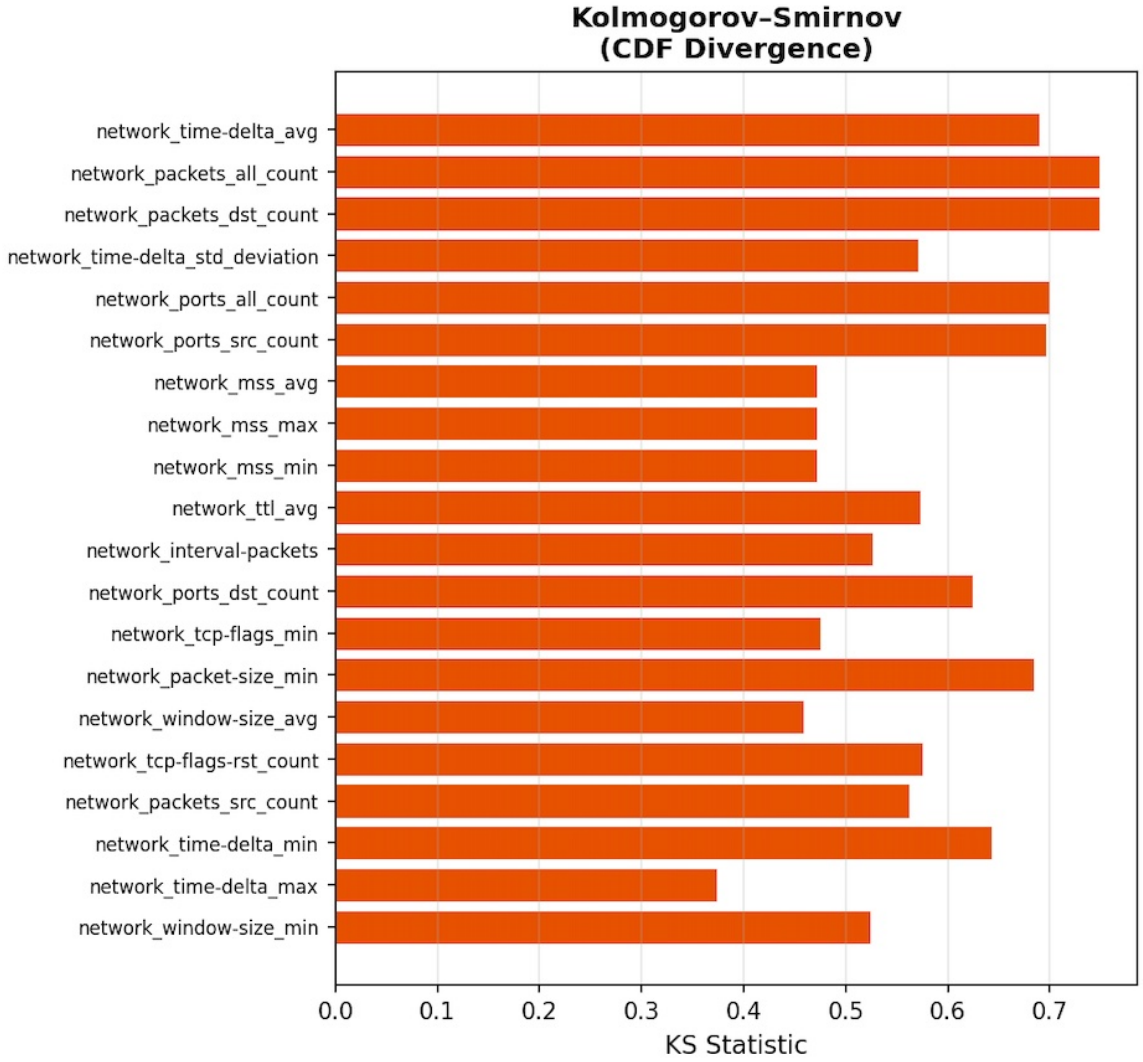

Figure 6.3: Statistical Feature Evaluation on CIC-IIoT 2025 — Kolmogorov–Smirnov statistics (orange) for the top 20 features, measuring CDF divergence between benign and attack traffic distributions.

6.3 Top 40 Features Selected by CD+MW+KS

Table 6.1 presents the top 40 features selected by the CD+MW+KS composite filter, ranked by their normalised composite scores. These features constitute the final recommended configuration.

Table 6.1: Top 40 Features Selected by CD+MW+KS Composite Filter (Part I, Ranks 1–20)

Rank	Feature	Composite Score
1	network_time-delta_avg	0.9053
2	network_packets_all_count	0.8154
3	network_packets_dst_count	0.8154
4	network_time-delta_std_deviation	0.7873
5	network_ports_all_count	0.7748
6	network_ports_src_count	0.7489
7	network_mss_avg	0.7289
8	network_mss_max	0.7185
9	network_mss_min	0.7063
10	network_ttl_avg	0.6941
11	network_interval-packets	0.6111
12	network_ports_dst_count	0.6115
13	network_tcp-flags_min	0.5888
14	network_packet-size_min	0.5735
15	network_window-size_avg	0.5470
16	network_tcp-flags-rst_count	0.5386
17	network_packets_src_count	0.5393
18	network_time-delta_min	0.5191
19	network_time-delta_max	0.5177
20	network_window-size_min	0.5192

Table 6.2: Top 40 Features Selected by CD+MW+KS Composite Filter (Part II, Ranks 21–40)

Rank	Feature	Composite Score
21	network_window-size_std_deviation	0.4952
22	log_data-types_count	0.4909
23	network_ip-flags_std_deviation	0.4859
24	network_tcp-flags-syn_count	0.4830
25	network_tcp-flags-ack_count	0.4733
26	network_tcp-flags_avg	0.4654
27	network_macs_dst_count	0.4068
28	network_packet-size_avg	0.4043
29	network_ips_all_count	0.3908
30	log_data-ranges_avg	0.3749
31	log_interval-messages	0.3704
32	log_messages_count	0.3690
33	network_tcp-flags_std_deviation	0.3634
34	network_ttl_std_deviation	0.3603
35	network_ttl_min	0.3567
36	network_packet-size_std_deviation	0.2936
37	network_ttl_max	0.2583
38	network_protocols_src_count	0.2330
39	network_fragmented-packets	0.1846
40	log_data-ranges_std_deviation	0.1576

6.3.1 Composite Score Visualisation

Figure 6.4 shows the composite statistical scores for the top 25 features, distinguishing those retained in the final selected feature set (dark blue) from those removed by correlation pruning (light blue). Features with composite scores above 0.7 are unanimously retained, while features in the 0.5–0.7 range with high mutual Pearson correlation are pruned.

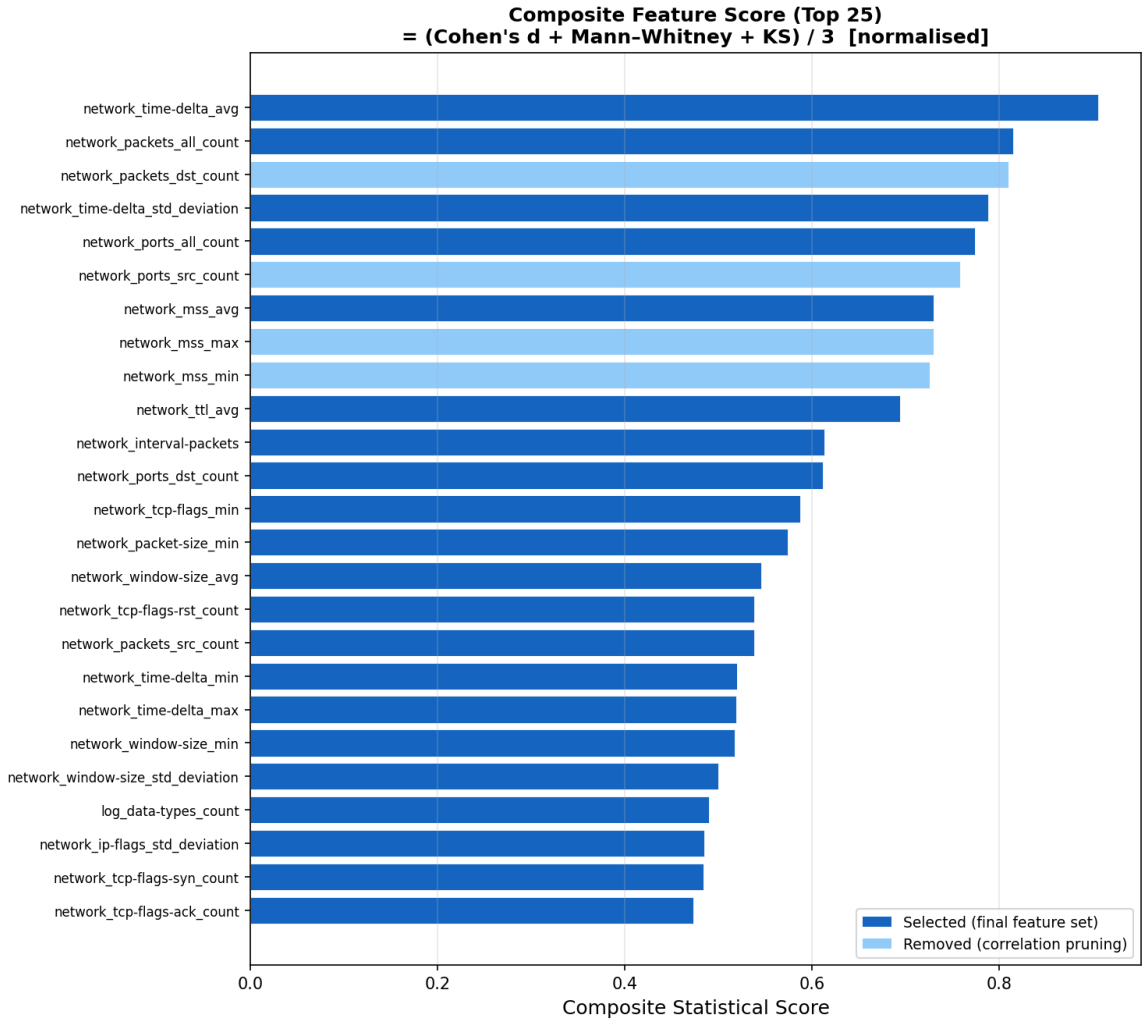

Figure 6.4: Composite Feature Score (Top 25) = (Cohen's d + Mann–Whitney + KS) / 3 [normalised]. Dark blue bars indicate features retained in the final feature set; light blue bars indicate features removed by correlation pruning. The composite score provides an integrated view across mean, rank, and distributional perspectives.

6.3.2 Pearson Correlation Matrix

Figure 6.5 presents the Pearson correlation matrix for the final selected feature set. The matrix confirms that the greedy correlation pruning step successfully removes highly correlated feature pairs ($|r| > 0.85$), resulting in a selected set with predominantly low inter-feature correlations. Notable high-correlation clusters occur among MSS features (avg/max/min) and among TCP flag counts, explaining the pruning decisions visible in Figure 6.4.

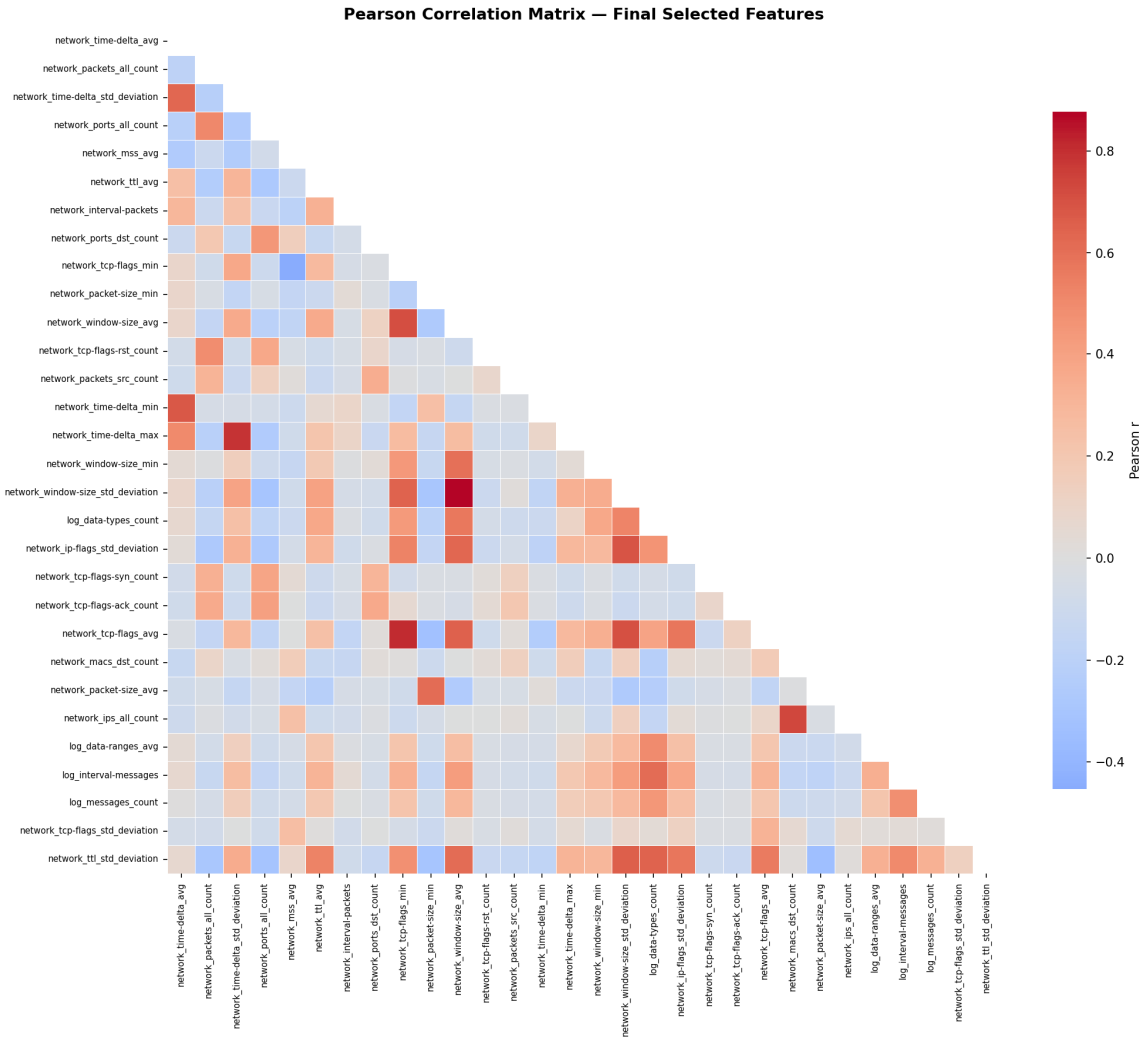

Figure 6.5: Pearson Correlation Matrix for the final selected feature set. Red cells indicate positive correlation; blue cells indicate negative correlation. The predominantly low off-diagonal values confirm that the selected features carry largely non-redundant information.

6.4 Individual Filter Performance

Table 6.3 reports the best XGBoost performance for each of the three individual statistical tests, identified by sweeping N from 5 to 47.

Table 6.3: Individual Filter Comparison — Best XGBoost Performance

Filter	**Best N**	**Accuracy**	**F1**
CD only	45	0.9904	0.9917
MW only	35	0.9904	0.9918
KS only	45	0.9905	0.9918

6.5 Composite Filter Performance

Table 6.4 compares all four composite filter combinations at their optimal feature counts for XGBoost. The CD+MW+KS composite achieves the highest accuracy and is adopted as the final recommended configuration.

Table 6.4: Composite Filter Comparison — XGBoost Performance

Filter Combination	**Best N**	**Accuracy**	**F1**
CD + MW	40	0.9904	0.9917
CD + KS	40	0.9904	0.9918
MW + KS	40	0.9904	0.9917
CD + MW + KS	40	**0.9905**	**0.9918**

6.6 Classifier Performance Under CD+MW+KS Filter

Table 6.5 reports the peak performance of each of the six classifiers under the best-performing CD+MW+KS filter configuration.

Table 6.5: Peak Classifier Performance Under CD+MW+KS Filter

Classifier	**Best N**	**Accuracy**	**F1**	**AUC-ROC**
Logistic Regression	47	0.9194	0.9277	0.9638
Extra Trees	35	0.9854	0.9875	0.9955
MLP	45	0.9854	0.9874	0.9981
Random Forest	40	0.9879	0.9896	0.9976
XGBoost	40	**0.9905**	**0.9918**	**0.9992**
LightGBM	45	0.9907	0.9920	0.9992

6.7 Feature Count Threshold Sweep

Table 6.6 details the evolution of XGBoost and LightGBM performance as the feature count N is swept from 5 to 47 under the CD+MW+KS filter. Performance rises sharply until $N = 25$–30, after which it levels off.

Table 6.6: Threshold Sweep — CD+MW+KS Filter (XGBoost and LightGBM)

	XGBoost			LightGBM		
N	**Acc**	**F1**	**AUC**	**Acc**	**F1**	**AUC**
5	0.9305	0.9380	0.9659	0.9307	0.9381	0.9664
10	0.9711	0.9749	0.9951	0.9703	0.9743	0.9950
15	0.9750	0.9783	0.9962	0.9729	0.9765	0.9958
20	0.9813	0.9838	0.9975	0.9803	0.9829	0.9973
25	0.9887	0.9903	0.9989	0.9890	0.9905	0.9990
30	0.9895	0.9909	0.9990	0.9895	0.9909	0.9991
35	0.9895	0.9909	0.9990	0.9894	0.9909	0.9991
40	**0.9905**	**0.9918**	**0.9992**	**0.9905**	**0.9918**	**0.9992**
45	0.9904	0.9917	0.9992	0.9907	0.9920	0.9992
47	0.9903	0.9917	0.9992	0.9907	0.9920	0.9992

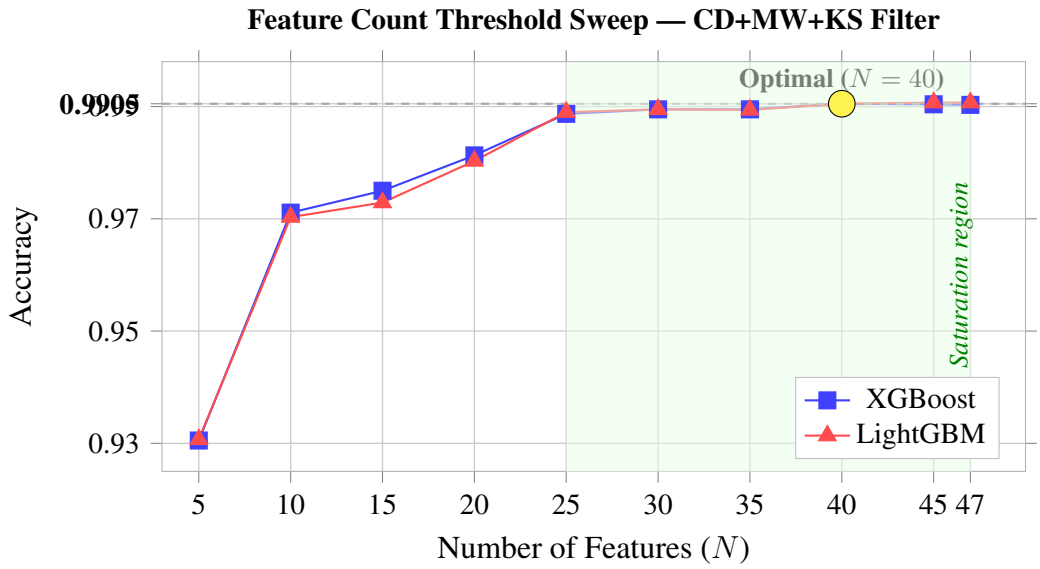

Figure 6.6: Feature count threshold sweep for XGBoost and LightGBM under the CD+MW+KS composite filter. Accuracy rises sharply until $N \approx 25$–30 and plateaus at $N = 40$, confirming that the selected 40-feature subset captures the information plateau while maintaining computational efficiency.

6.8 Discussion

6.8.1 Optimal Feature Count

The threshold sweep (Table 6.6, Figure 6.6) reveals a characteristic saturation pattern: performance rises sharply from $N = 5$ to approximately $N = 25$–30, after which further

feature addition yields diminishing returns. The optimal feature count is $N = 40$, representing a 57.9% reduction from the original 95 features. This compactness has important practical implications for resource-constrained IoT gateway hardware.

6.8.2 Complementarity of the Three Statistical Tests

The most theoretically significant finding of this work is the demonstrated complementarity of the three statistical tests. Features with high Cohen's d values tend to have large mean differences (e.g., `network_mss_avg` with $d = 1.21$). On the other hand, the Mann-Whitney U test focuses on those features with high rank differences irrespective of mean differences, which means that it can capture highly skewed features like `network_packets_all_count`. The KS statistic additionally captures features with distributional shape differences beyond the mean, such as `network_time-delta_min` ($D = 0.857$, but lower CD). Together, the three tests cover mean-based, rank-based, and distribution-based perspectives on separability, which is why their combination consistently outperforms any individual test alone and any pairwise combination (Tables 6.3 and 6.4).

6.8.3 Tree Ensembles vs. Linear Models

The substantial gap between gradient-boosted tree ensembles (XGBoost: 99.05%, LightGBM: 99.07%) and Logistic Regression (91.94%) confirms that the feature-class boundary in IoT network traffic is strongly non-linear [24]. XGBoost reaches the optimum at a more compact feature set ($N = 40$), making it the preferred configuration for resource-constrained deployments. While LightGBM achieves marginally higher accuracy at $N = 45$, XGBoost reaches equivalent peak performance at the more compact $N = 40$ feature set. Notably, even Logistic Regression achieves an AUC-ROC of 96.38% on the selected features, attesting to the quality of the features selected by the composite filter even for a linear separator.

6.8.4 Feature Interpretation

The top-ranked features are dominated by temporal flow statistics (`time-delta_avg`, `time-delta_std_deviation`), packet count metrics (`packets_all_count`), port diversity metrics (`ports_all_count`, `ports_dst_count`), TCP control-plane indicators (`tcp-flags-rst_count`, `tcp-flags-syn_count`), and TTL-based attributes. These features collectively capture the behavioral signatures that distinguish IoT attack traffic: high packet rates, unusual port diversity, anomalous TCP flag patterns, and irregular inter-packet timing—consistent with findings in the broader IoT IDS literature [14, 15].

7 Conclusion

In this work, we proposed a hybrid feature selection pipeline for IoT intrusion detection networks, integrating the effect size of Cohen's d (CD), Mann–Whitney U (MW), and Kolmogorov–Smirnov (KS) tests into a composite approach. When applied to the CIC-IIoT 2025 dataset, our pipeline selects a compact set of 40 from a total of 95 raw features. The CD+MW+KS pipeline with the XGBoost classifier yielded 99.05% accuracy and 99.92% AUC-ROC score, achieving peak detection performance by using just 40 features among 95 total ones. Among all seven filter configurations—three individual tests and four composite combinations—CD+MW+KS is significantly better than all other configurations, verifying their complementary nature.

Four key conclusions emerge from this work:

1. **Composite statistical filters outperform individual tests.** The CD+MW+KS composite consistently achieves higher accuracy than any individual test (CD, MW, or KS alone) or any pairwise combination, demonstrating the empirical value of integrating mean-based, rank-based, and distribution-based statistical perspectives.

2. **Optimal feature count balances performance and efficiency.** Performance plateaus at approximately 40 features, beyond which additional features provide no benefit. This compact representation is well-suited for deployment on resource-constrained IoT edge devices.

3. **Tree-based ensembles are the superior classifiers.** The large gap between XGBoost-/LightGBM (>99%) and Logistic Regression (∼92%) confirms the highly non-linear nature of the feature-class boundary in IoT network traffic.

4. **Leakage-free feature selection is essential for credible evaluation.** By computing all feature statistics, the correlation matrix, and the data scaler exclusively on the training partition, the reported results provide genuine unbiased estimates of generalisation performance.

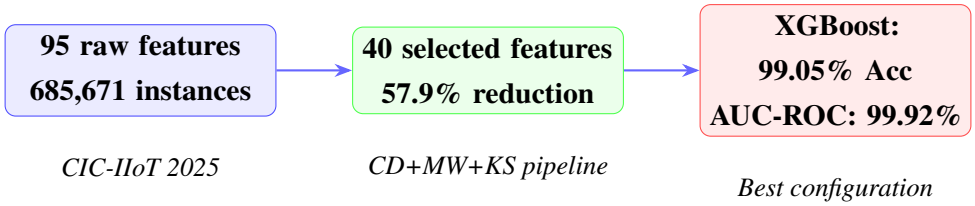

Figure 7.1: Summary of the proposed framework: from 95 raw features to 40 selected features with 57.9% dimensionality reduction, achieving 99.05% accuracy on the CIC-IIoT 2025 dataset with XGBoost.

8 Future Scope and Social Impact

8.1 Future Scope

Future work will consider how to scale the pipeline to accommodate multi-class attack classification based on the complete taxonomy of attacks in CIC-IIoT 2025, as well as exploring approaches that incorporate concept drift to update the importance of features in real-time during live operation, and evaluating federated learning models that enable edge devices to participate in training the model without consolidating traffic data.

1. **Multi-class attack classification**: The current framework addresses binary (attack vs. benign) classification. Extending the composite filter to multi-class settings—distinguishing DDoS, reconnaissance, spoofing, and injection attacks—would increase the operational utility for security analysts.

2. **Concept drift adaptation**: Real IoT network environments are non-stationary; attack patterns evolve over time. Incorporating online or streaming feature selection mechanisms that adapt the composite filter as new data arrives would extend the framework's utility to live deployment scenarios.

3. **Federated learning integration**: Privacy-preserving federated learning would allow IoT edge devices to contribute to model training without centralising sensitive traffic data. The compact feature set identified in this work—requiring only 40 features—is well-suited to federated settings where communication bandwidth is limited.

4. **Adversarial robustness**: Sophisticated adversaries may craft network traffic designed to evade the features identified by the composite filter. Future work should evaluate the robustness of the proposed framework to adversarial perturbations and explore defensive feature selection strategies.

5. **Explainable AI integration**: Incorporating SHAP values or attention-based explanations into the detection pipeline would enable security analysts to understand model decisions at the per-flow level, increasing confidence in automated detection and facilitating incident response.

6. **Hardware-accelerated edge deployment**: Implementing the 40-feature XGBoost inference pipeline on FPGA or specialised neural processing units (NPUs) would validate the practical deployability of the framework on resource-constrained IoT gateway hardware.

8.2 Social Impact

The societal implications of effective IoT intrusion detection extend well beyond technical performance metrics:

1. **Protection of critical infrastructure**: Industrial IoT systems underpin power grids, water treatment facilities, and transportation networks. Effective intrusion detection directly reduces the risk of catastrophic infrastructure failures caused by cyberattacks.

2. **Healthcare security**: IoT medical devices—insulin pumps, pacemakers, remote patient monitors—are life-critical. Robust, lightweight IDS that can operate at the network level without burdening individual devices are essential for patient safety.

3. **Privacy preservation**: Many IoT devices—smart speakers, cameras, wearables—continuously collect sensitive personal data. Effective intrusion detection helps prevent unauthorised access to such data, protecting individual privacy.

4. **Economic impact**: The global economic cost of cybercrime exceeded $8 trillion in 2024, with IoT-related incidents accounting for an increasing share. Lightweight, deployable IDS can materially reduce this burden for individuals, enterprises, and governments.

5. **Open science and reproducibility**: By validating the framework on a publicly available dataset and providing a reproducible, leakage-free experimental protocol, this work contributes to the cumulative advancement of the IoT security research community.

References

[1] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, "IoT: Internet of Threats? A survey of practical security vulnerabilities in real IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8182–8201, 2019.

[2] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali, and M. Guizani, "A survey of machine and deep learning methods for Internet of Things (IoT) security," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1646–1685, 2020.

[3] M. M. Rashid, J. Kamruzzaman, M. M. Hassan, T. Imam, and S. Gordon, "Cyberattacks detection in IoT-based smart city applications using machine learning techniques," *International Journal of Environmental Research and Public Health*, vol. 17, no. 24, p. 9347, 2020.

[4] M. Almohaimeed, "Statistical feature selection for lightweight IoT intrusion detection," *Applied Sciences*, vol. 14, no. 3, p. 1247, 2024.

[5] A. Haque, M. Rahman, A. Islam, and N. Nasrin, "Entropy-based feature engineering for IoT intrusion detection using random forests," *IEEE Access*, vol. 9, pp. 78467–78482, 2021.

[6] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, p. 107247, 2020.

[7] M. A. Hall, "Correlation-based feature selection for machine learning," Ph.D. dissertation, Dept. Computer Science, University of Waikato, Hamilton, New Zealand, 1999.

[8] S. Shukla, B. Maheshwari, and M. Gupta, "KS-test-based distributional anomaly detection for network flows," *Procedia Computer Science*, vol. 167, pp. 516–525, 2020.

[9] J. Santos, B. Rodrigues, W. Zago, P. Viegas, and A. Nogueira, "Information-theoretic feature ranking for network intrusion detection," *Sensors*, vol. 22, no. 4, p. 1547, 2022.

[10] O. H. Abdulganiyu, T. A. Ait Tchakoucht, and Y. K. Saheed, "A systematic literature review for network intrusion detection system (IDS)," *International Journal of Information Security*, vol. 22, no. 5, pp. 1125–1162, 2023.

[11] P. Gyenizse, O. Szabo, and G. Jakab, "XGBoost and LightGBM for network intrusion detection: A comparative analysis on modern IoT datasets," *Electronics*, vol. 12, no. 14, p. 3092, 2023.

[12] M. A. Bouke and A. Abdullah, "An empirical study of pattern leakage impact during data preprocessing on machine learning-based intrusion detection models reliability," *Expert Systems with Applications*, vol. 230, p. 120715, 2023. doi: 10.1016/j.eswa.2023.120715

[13] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023.

[14] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in *Proc. Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, 2015, pp. 1–6.

[15] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, pp. 779–796, 2019.

[16] Canadian Institute for Cybersecurity, "CIC-IIoT 2025 Dataset," University of New Brunswick, 2025. [Online]. Available: https://www.unb.ca/cic/datasets/iiot-dataset-2025.html

[17] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed. Hillsdale, NJ: Lawrence Erlbaum Associates, 1988.

[18] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50–60, 1947.

[19] A. Kolmogorov, "Sulla determinazione empirica di una legge di distribuzione," *Giornale dell'Istituto Italiano degli Attuari*, vol. 4, pp. 83–91, 1933.

[20] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[21] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, 2016, pp. 785–794.

[22] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017, pp. 3149–3157.

[23] I. S. Thaseen, C. A. Kumar, and A. Ahmad, "An integrated intrusion detection system using correlation-based feature selection and SVM," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 4, p. e5910, 2021.

[24] A. Mahfouz, D. Venugopal, and S. G. Shiva, "Comparative analysis of ML classifiers for network intrusion detection," in *Proc. International Conference on Information Technology and Electrical Engineering (ICITEE)*, 2020, pp. 1–6.

[25] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, 2006.

[26] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, Ottawa, Canada, 2009, pp. 1–6.

[27] S. Walling and S. Lodh, "Network-based intrusion detection for IoT using machine learning and statistical hybrid feature selection," *Security and Privacy*, vol. 7, no. 6, p. e429, 2024. doi: 10.1002/spy2.429

[28] F. Kamalov, H. H. Thabtah, and I. Leung, "Feature selection for intrusion detection systems," *Journal of Information and Telecommunication*, vol. 5, no. 4, pp. 412–428, 2021. arXiv:2106.14941

[29] G. Andresini, A. Appice, N. Di Mauro, C. Loglisci, and D. Malerba, "Supervised feature selection for network intrusion detection: A critical review," *Engineering Applications of Artificial Intelligence*, vol. 101, p. 104216, 2021.

[30] J. Wang, X. Xiong, G. Chen, R. Ouyang, Y. Gao, and O. Alfarraj, "Multi-criteria feature selection based intrusion detection for Internet of Things big data," *Sensors*, vol. 23, no. 17, p. 7434, 2023. doi: 10.3390/s23177434

[31] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th International Conference on Information Systems Security and Privacy (ICISSP)*, Funchal, Portugal, 2018, pp. 108–116. doi: 10.5220/0006639801080116

[32] T. Bammidi, R. Pendela, and V. Chakilam, "Top-K feature selection for IoT intrusion detection: Evaluation of XGBoost, LightGBM, and Random Forest," *Future Internet*, vol. 17, p. 529, 2025.

[33] F. Z. Janane, A. Ouaderhman, and A. Chamlal, "A filter feature selection method for high-dimensional classification data," *Journal of Simulation*, 2023. doi: 10.1177/17483026231184171

[34] K. Albulayhi, Q. Abu Al-Haija, S. A. Alsuhibany, A. A. Jillepalli, M. Ashrafuzzaman, and F. T. Sheldon, "IoT intrusion detection using machine learning with a novel high performing feature selection method," *Applied Sciences*, vol. 12, no. 10, p. 5015, 2022. doi: 10.3390/app12105015

[35] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[36] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PLOS ONE*, vol. 10, no. 3, p. e0118432, 2015.

[37] W. Chimphlee and S. Chimphlee, "Network intrusion detector using multilayer perceptron (MLP) approach," *Turkish Journal of Computer and Mathematics Education*, vol. 13, no. 3, pp. 488–499, 2022. doi: 10.17762/turcomat.v13i03.13018

[38] I. G. Peres, "Effect sizes for nonparametric tests: Mann-Whitney, Kolmogorov-Smirnov, and Wilcoxon," *Biochemia Medica*, vol. 36, no. 1, p. 010101, 2026. doi: 10.11613/BM.2026.010101

[39] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, p. 20, 2019.

[40] G. Balhareth and M. Ilyas, "Optimized intrusion detection for IoMT networks with tree-based machine learning and filter-based feature selection," *Sensors*, vol. 24, no. 17, p. 5712, 2024. doi: 10.3390/s24175712