Bidirectional Feature-Subset Transfer between Binary Intrusion Detection and Multiclass Attack Attribution in IoT Communication Networks

[1]Ahmed Sabri Ghazi Behadili*
[1]College of Engineering, Al-Karkh University of Science, Baghdad, Iraq
*e-mail: eng.ahmed.sabri@kus.edu.iq

(***received*:** 18 October 2026, ***revised*:** 4 November 2026, ***accepted*:** 5 November 2026)

Abstract

The set of traffic features used in a reduced binary intrusion detection and multiclass attack attribution may not work equally well in IoT communication networks. This study separates out the impact of granularity of the label selection by moving subsets both ways, keeping the data partitions, classifier settings, and feature budgets constant. The contribution is a paired evaluation of re-use loss complemented with protocol shortcut audits and comparing shared subsets with subsets selected for each task. Four selectors, two tree ensemble classifiers, and 30 matched splits were used for evaluating Edge-IIoTset, the network table and CICIoT2023. Absolute regret is the magnitude of a loss and normalized regret is the loss magnitude relative to the leeway in the goal task. Practically relevant harm was detected in 17, 15 and 22 configurations with the inference with dependence correction on the common grid k = 5-20, respectively. In all of the cases, there was no direction of transfer that was better. Each testbed had a unique shortcut that was associated with MQTT coverage, DNS applicability or a fixed mix of TCP and UDP. They were removed, reducing a few effects but not the primary ones. Recall loss was not always correlated with the rarity of an attack family. Borda fusion tuned on validation data frequently produced a parent subset, and new subsets did not reliably lead to a decrease in loss for the worse task. This validation should therefore be performed on both label views, before shared feature pipelines are used instead of task-specific subsets, against an explicit loss tolerance.

Keywords: feature selection, IoT communication networks, IoT intrusion detection, multiclass classification, network traffic analysis

1 Introduction

Internet of Things (IoT) and Industrial IoT (IIoT) constrained devices interact with networks that span various types. To safeguard these endpoints, network intrusion-detection systems (IDSs) study flow or packet-level traffic [1]. Each of the predictors that a gateway or edge detector holds must be removed from live traffic. Use of feature count then is a design constraint, not a detail [2] [3].

In the binary and multiclass ToN-IoT label setting, Li et al. [4] set the selection against extraction. Sezgin et al. [5] conducted a comparison of optimisers under the same protocol, using accuracy, subset size as well as runtime. These comparisons do not, however, separate the loss due to the reuse of a subset of rows whose label view is ranked below that of the other.

There are predictors specific to each category, which are maintained by class-aware selection [6, 7] and stage-specific pipelines use different subsets at each decision point [8]. This is a line of research that has shown that subsets are not always freely reusable. It doesn't indicate if they can be traded or not, or even if they can be traded and for what reason. In this case, the traffic is benign in all classes; Therefore, it is not a second stage with attacks only.

The open problem is the cost of changing the labels in the ranking while the evaluation conditions remain unchanged. A binary ranking can overlook the distinctions between attacks, and a multiclass ranking can be overly lenient on distinguishing between benign and attack traffic. A good score for each of the view scores does not imply compatibility of the subsets. This can only be seen by comparing two things directly. The closed world benchmarks can also provide misleading results, as they can be different from how an intrusion detector is deployed [9, 10]. Measurable shortcuts can be introduced both into the construction of the data sets and during processing [11] [12] [13]. An audit can help remove artefacts from the traffic generator from the ranking labels.

 ISSN:2302-8149
e-ISSN:2540-9719

In this study, a controlled evaluation for subset reuse is performed across different level of granularity for the labels. It is different from the comparisons within a single task reviewed above in four ways. First, the two transfer directions are tested with reference sets chosen on the target task with the same sampled rows, splits, budgets and the same classes that are frozen. In this way the ranking labels are not discovered with the changes in the evaluation data. Second, absolute loss, a score ceiling guard and a score headroom normalization are combined with dependent-corrected inference to distinguish meaningful harm from amplification at the score ceiling. Third, predictor-value audits reveal three protocol "short-cuts". Transfers that are matched with sensitivity are then estimated and their contribution to transfer loss is found. Finally, Borda fusion is validated and contrasted with both endpoints. This helps to distinguish parent reproduction from true subsets and to evaluate the value of common selection.
The goal is to determine when the detection or attribution of reuse is degraded, by either side, when there is a consistent degradation of the same, and when the selection of worst task loss occurs, so that features used in the communication networks for IoT can be selected accordingly.

2 Literature Review

Prior studies can be sorted by what they optimise and what they hold fixed, and this distinguishes evidence for compact within-task models from evidence that selected features stay useful once the ranking labels change.
How useful subsets are obtained is the focus of statistical relevance measures [14], selection–normalisation analysis [15] and ensembles that pair correlation filters with embedded methods [16]. Scores obtained within one task cannot establish reuse loss. A criterion may perform well locally yet prioritise information that the destination task does not need. Reduction has also been tied to efficiency and model relevance through ensemble selection for constrained devices [2], common-feature selection with autoencoders [3] and SHAP-guided ensembles [17]. Even so, compatibility between detection and attribution has to be tested, because a small feature count is not, on its own, evidence of deployability.
Classifier-guided search has been explored through recursive elimination with a decision tree [18], RF ranking followed by genetic optimisation [19] and grey wolf optimisation inside voting ensembles [20]. The resulting subsets depend on the optimisation objective. A strong local solution must therefore still be evaluated once the destination task changes.
Subset search has been widened further by grey wolf optimisation with random forests [21], particle swarm optimisation [22] and metaheuristic ensembles [23]. Here, fixed rankings are used instead. They isolate sensitivity to the label view and complement these optimisation studies, although the selector set is deliberately restricted. Li et al. [4] did evaluate both ToN-IoT label views, but their comparison of selection and extraction stayed within each task; Tseng et al. [24] studied multiclass modelling on CICIoT2023. Reporting results under each label view is not the same as isolating the transfer effect. That requires transferred and target-local subsets to be compared on identical test observations. Sezgin et al. [5] ran a controlled optimiser comparison with 30 independent runs, using one dataset and one binary decision tree. The overlapping splits used here capture a different kind of variability, which is why the dependence correction in Section 3.7 is needed.
A vehicular intrusion detector has combined feature selection with a multistage design [25]. Sakthivel and Kumarasamy [8] tailor subsets to a cascade of a confidence-gated SVM and a neuro-fuzzy stage. Both architectures acknowledge that different stages need different information, yet their routing is unlike a multiclass target that keeps benign traffic. Gains at the architecture level do not, therefore, establish that subsets are compatible in both directions.
Earlier work has mapped system level features to attack categories [7] and preserved class coverage during selection [6]. Other studies combine selection conditioned on labels with coreset budgets set for each class [26], or combine balancing, selection and ensembles [27]. All of these account for how classes are represented, which justifies checking the recall of each family attack after reuse. They do not imply that class size decides the damage, so H3 tests whether support is associated with loss. In this study, the same capped samples are used in every condition, and changes in class composition cannot confound the paired subset comparisons. The altered prevailing does, however, limit what can be said about deployment.

 ISSN:2302-8149
e-ISSN:2540-9719

Transfer across environments has been studied through selection guided by SHAP across datasets [28], standard NetFlow features [29] and detection guided by domain knowledge [30]. In the present design, the rows stay fixed and only the ranking labels change. This separates label granularity from distribution shift. It does not establish generalization across sites or schemas.

Studies of operational systems and of machine learning for security show that benchmark accuracy depends on how the evaluation is designed [9], [10], [13]. A protocol field might reveal how a testbed generated a particular class. Pairing the label views controls the comparison, but it cannot remove a shortcut that both views inherit from the same data. Benchmark audits have traced bias to errors in construction and processing [11], [12]. Reviews of deployment also report that evidence on resources and generalization is often incomplete [31]. The predictor-value audit carried out here carries these concerns over to protocol applicability. It is diagnostic in purpose, and the predictors that remain are not certified free of artefacts.

None of these studies measures the paired cost of moving a subset between binary and multiclass labels while rows, budgets and classifiers stay fixed; that cost is what this study evaluates.

3 Research Method

Figure 1 shows the paired evaluation and the diagnostic sensitivity path, and the subsections below define the frozen settings and protocol amendments. Where the dashed arrows denote diagnostic sensitivities using the same rows and splits.

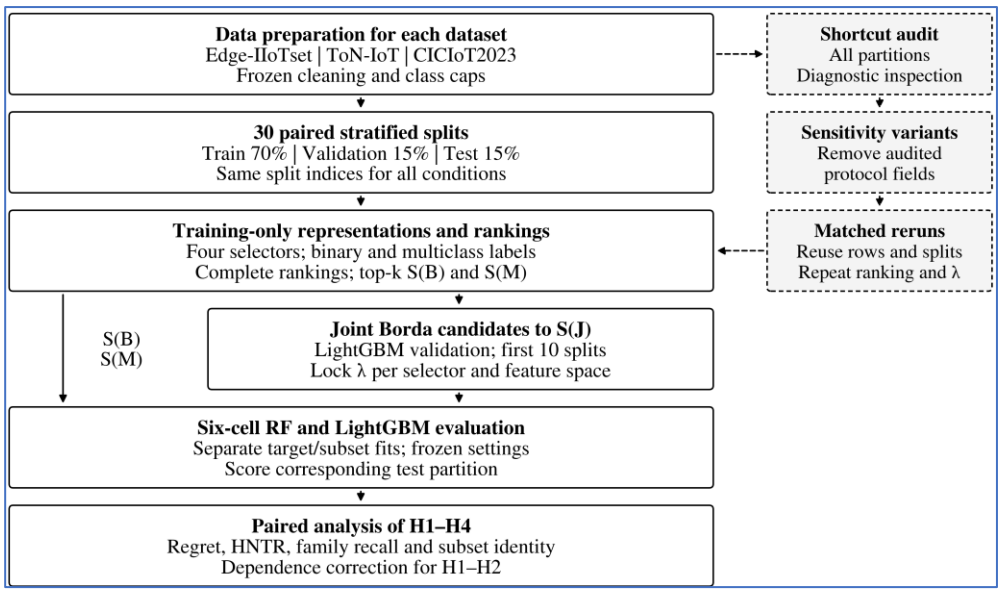


Figure 1 Workflow of the Paired Feature-Subset Transfer Experiment. S(B), S(M) and S(J) denote binary-ranked, multiclass-ranked and joint subsets.

Algorithm 1 specifies the paired evaluation procedure for the documented dataset variants.

Algorithm 1 Paired Feature-Subset Transfer Evaluation

Input: Prepared primary and diagnostic variants; 30 frozen paired splits; registered selectors, budgets and classifier settings; Borda weights from Section 3.6.

Output: Paired evaluation records and H1–H4 summaries.

For each dataset variant d do
  Reuse its prepared rows and 30 frozen training/validation/test splits (70%/15%/15%).
  For each split s and selector r, fit training-only representations and store both complete rankings.
  For each selector r, select and lock the minimax λ before testing using LightGBM validation across the first ten splits and all registered budgets (Section 3.6).
  For each split s, selector r, budget k and classifier c do
    Form S(B,k), S(M,k) and S(J,k,λ) from the stored rankings; use the locked λ for both classifiers.
    Fit the six target/subset cells separately on training data with frozen settings (Section 3.5).
    Score the corresponding test partition; store predictors, metrics, timing, class recall and support.

 ISSN:2302-8149
e-ISSN:2540-9719

9 *End for*
10 Compute paired regret and HNTR; apply the H2 ceiling guard and H1–H4 analyses, including dependence correction for H1–H2 (Section 3.7).
11 *End for*
12 Compare primary and diagnostic sensitivity outputs on matched rows, splits and common budgets.

3.1 Study Design and Protocol Amendments

The binary and multiclass labels in each split are two different rankings, the resulting subsets are tested on the same rows (for training, validation and testing), the same classifier settings and the same feature budgets, and only the label used for selection differs for each subset. Matched configuration is the one where the variant of the dataset, the seed for the split, the selector is fixed, the budget k is fixed, the classifier is fixed, all comparisons are matched by seed, and finally, the source and target task are fixed and identical. S(B,k) is the set of top k predictors when ranked by binary labels and S(M,k) is the set of top k predictors when ranked by multiclass labels. S(J,k,λ) is the merging of the two complete rankings, with a weight chosen on a validation data set. All changes to Protocol Version 1.2 were made prior to the experiments, and subsequent changes are described below. Any analyzes conducted after looking at the test results are considered post hoc. The confirmatory hypotheses are:

H1: transferred subsets underperform target-local subsets on average.

H2: M→B transfer has smaller headroom-normalised regret than B→M transfer.

H3: lower-support attack families suffer larger B→M recall losses, implying a negative support–loss association (amended test: Section 3.7).

H4: S(J) lowers worst task-normalised loss below the better task-specific endpoint. Only this component of Version 1.2 is tested; its statistical-competitiveness clause is untested.

Code and validation metadata fixed k = 5, the dataset-feasible budgets, the classifiers, λ selection and the ceiling guard before each affected test evaluation; split seeds were not changed. Removal of TCP/UDP from CICIoT2023 was defined only after the primary results were known. It was frozen before its own evaluation but remains post hoc.

After the Edge-IIoTset test results were inspected, the planned Cliff's delta and bootstrap intervals were replaced by rank biserial correlation and 95% t intervals. H3 was evaluated through the Spearman association between class support and recall regret, and its structural similarity part was not tested. H4 retained the comparison of worst loss but dropped the competitiveness clause. Both the comparison of low-support families and the breakdown of H4 by distinct subsets are likewise post hoc.

3.2 Datasets and Data Preparation

Each of Edge-IIoTset [32], the ToN-IoT network table [33] and CICIoT2023 [34] is evaluated on its own (Table 1). All three carry the benign and attack-family information needed to derive both label views from the same rows. Paired reuse comparisons are therefore possible without joining unrelated samples.

The datasets were picked because they represent IoT communication traffic in different ways. Edge-IIoTset uses packet-level protocol fields, ToN-IoT uses Zeek connection fields, and CICIoT2023 uses statistics computed over aggregation windows. With different feature spaces and taxonomies, it becomes possible to check whether a directional finding holds across distinct testbeds. Schemas are not harmonised. Nor are models transferred between datasets: the comparison concerns scope, and it is not an external-validation experiment.

For ToN-IoT, only "Train_Test_Network.csv" is used; the Windows, Linux and telemetry tables are left out.

Comparisons across datasets use $k \in \{5, 10, 15, 20\}$. On Edge-IIoTset, k = 30 is added as a near-saturation control, which keeps 30 of the 36 predictors. Sensitivity analysis on CICIoT2023 is restricted to $k \in \{5, 10\}$, so robustness to TCP/UDP removal at k = 15 or 20 has not been tested.

 ISSN:2302-8149
e-ISSN:2540-9719

Table 1 Frozen dataset scope and feature budgets

Dataset	Retained rows	Classes	Primary predictors	Primary k values	Sensitivity analysis
Edge-IIoTset	156,957	15	36	5, 10, 15, 20, 30	Remove ten MQTT predictors; 26 remain; k = 5, 10, 15, 20
ToN-IoT network table	178,434	10	35	5, 10, 15, 20	Remove four DNS-flag predictors; 31 remain
CICIoT2023	812,272	34	39	5, 10, 15, 20	Remove TCP and UDP; 37 remain; k = 5, 10

Source files are identified by SHA-256 fingerprints before any ranking takes place. The audits keep a record of schemas, mappings, exclusions, malformed values, class counts, sampling decisions and splits. No preprocessing step is fitted to test data.

The file ML-EdgeIIoT-dataset.csv held 157,800 rows and 63 columns. Exact deduplication dropped 814 rows. Fifteen predictors judged in advance to be prone to leakage were excluded, among them hosts, ports, raw TCP payload, MQTT message content and HTTP URIs.

Mixed text was canonicalised so that numerically equivalent strings (e.g., "0" and "0.0") matched, while valid text categories were kept distinct. Constant predictors were dropped before checking for malformed rows. In 29 Normal rows, DNS query-name length held domain names; these rows were removed under the malformed-value policy. The final matrix contained 156,957 rows and 36 predictors, with no contradictions between binary and multiclass labels.

ToN-IoT held 211,043 rows and 44 columns, and deduplication removed 20,569 of them. Excluded fields were addresses, high-cardinality laboratory identifiers and textual content: DNS queries, SSL subjects and issuers, HTTP URIs and user agents.

Zeek marks inapplicable values with the token "-". This was mapped to zero in numeric fields and to an NA category in text fields. Applicability information was kept and no rows were deleted, although numeric zero may now be confused with inapplicability. With Normal capped at 30,000, the result was 178,434 rows and 35 predictors.

CICIoT2023 consisted of 309 CSV files with 39 predictors. Labels came from the top-level directories, and Benign_Final was mapped to BenignTraffic. Conversion to finite numeric values removed 1,040 of the 46,776,700 CICIoT2023 rows. The worst malformed fraction, 0.0022% in Rate, stayed under the 1% limit. Duplicates were controlled by deterministic within class 128 bit bottom hash sampling, which kept no more than 30,000 distinct rows per class and produced 812,272 rows with 39 predictors. Before ranking, the file and retained-row hashes, label mappings, malformed value reports and sampling seed were all frozen.

After canonicalisation, constant columns are removed. Any remaining malformed rows may be deleted only if every predictor's malformed fraction is ≤1%. If not, processing halts until a documented amendment is made. The class distribution of removed rows is recorded, and no imputation rule is learned from validation or test data.

3.3 Class Eligibility and Paired Splits

A class is eligible if it has at least 300 valid rows before sampling. Classes with 300–1,499 rows are kept whole; for the rest, n(c) = min(N(c), 30,000). No class is enlarged. SMOTE, oversampling and random undersampling are all excluded. Because the cap alters class prevalence, binary precision and predicted attack prevalence are not estimates of deployment prevalence. The paired subset comparisons are still appropriate, since every condition uses the same rows, and class recall is reported together with test support. Thirty multiclass-stratified splits are drawn with seeds 20260913–20260942, assigning 70%, 15% and 15% of the data to training, validation and test, respectively. The same indices serve every label view, selector, budget, classifier and subset source. Every class appears in every partition.

 ISSN:2302-8149
e-ISSN:2540-9719

3.4 Feature Ranking

One ranking is generated for each training split and each view for the label. It can be used without any refitting of its top k slices, and ties are broken by the name of the predictor.
The four selectors behave differently when the view of the label is updated, with each having its own common budget. Mutual information reflects overall dependency between feature and label. After discretisation, chi-square tests categorical association and ANOVA F compares classes to the variation within them. For additional layers of importance, add a ranking from a model (RF importance). These are well known and the comparisons can be controlled without the need for wrapper searches with various budgets. They cannot be used as a definitive list of selection methods.
Label views share labels, and both label views learn vocalabularies and bin boundaries from training data (Table 2). In the case of chi-square, the contributions of one-hot columns of the same contingency table are added together. The importances of the encoded columns are also added together to get the Random Forest (RF) importance. Whereas, analysis of variance (ANOVA) F takes the largest child score because a sum of F statistics does not have a test interpretation. If the same quantity is used again and again, then there may be fewer than 10 bins. If no categories are provided, a reserved ranking code will be assigned, and a one-hot block of zeros will be assigned to the classifier. Budgets are in original predictors and not in encoded columns.
Split seed controls both the mutual information and RF selection. The RF selector uses the RF settings provided in Section 3.5, and follows [35]. For the downstream RF and LightGBM [36] models, the seed of the classifiers will remain fixed when performing paired comparisons, and will use the seed of the upstream classifiers for seed zero.

Table 2 Feature-ranking representations and score aggregation.

Selector	Training representation	Aggregation to original predictor
Mutual information	Numeric predictors discretised into up to ten equal-frequency bins; categorical predictors as training-vocabulary codes; all inputs passed to the estimator as discrete	One score per original predictor
Chi-square	One-hot expansion of the training-fitted discrete codes	Sum across bins or categories
ANOVA F	Raw numeric predictors and one-hot categorical predictors	Maximum score among the encoded children
RF importance	Raw numeric predictors and one-hot categorical predictors	Sum of impurity importance across encoded children

3.5 Bidirectional Transfer Matrix and Classifiers

For every combination of dataset, split, selector, budget and classifier, six cells are evaluated (Table 3). A fitted classifier is used for each target task, one for each subset condition. No models or predictions are re-labeled for each task. The same frozen specifications (Table 4) are used for all datasets and conditions, allowing for the study of subset effects without the need to tune the classifiers. The reason why we picked RF and LightGBM as the algorithms to compare is that they are complementary tree ensembles. RF is used for randomising trees, and LightGBM is used for boosted trees. These two can process the mixed traffic representations after the encoding is documented and contrasting them reveals if a transfer effect is part of one direction of building an ensemble. Fixed settings will mask the subset loss. No superiority over deep models or other families of classifiers is claimed.
Seeds affect only the split and the seeded selector routines. The classifiers receive original numeric values, with categories encoded through vocabularies fitted on training data alone. When a categorical predictor is selected, all of its encoded columns come with it.

 ISSN:2302-8149
e-ISSN:2540-9719

Table 3 Bidirectional six-cell transfer matrix.

Cell	Subset	Target task	Role
B1	S(B)	Binary	Binary local reference
M1	S(B)	Multiclass	Binary-to-multiclass transfer
M2	S(M)	Multiclass	Multiclass local reference
B2	S(M)	Binary	Multiclass-to-binary transfer
B3	S(J)	Binary	Joint subset on binary detection
M3	S(J)	Multiclass	Joint subset on multiclass attribution

Table 4 Frozen classifier specifications.

Classifier	Frozen parameters
Random Forest	200 trees; maximum depth 20; minimum leaf size 2; square-root feature sampling; balanced class weights; all CPU threads; random seed 0
LightGBM	300 estimators; 63 leaves; learning rate 0.05; minimum child samples 20; balanced class weights; all CPU threads; random seed 0; verbosity −1

3.6 Joint Subset Construction

Among p eligible predictors, rank 1 is the best. Each rank r(f) is turned into a Borda utility by Equation (1). Equation (2) then merges the binary and multiclass utilities using $\lambda \in \{0, 0.25, 0.50, 0.75, 1.00\}$. At the endpoints, $\lambda = 1$ and $\lambda = 0$ reproduce the binary and multiclass rankings, respectively.

$$U(f) = \frac{p - r(f)}{p - 1} \quad (1)$$

$$J(\lambda, f) = \lambda \cdot U(B, f) + (1 - \lambda) \cdot U(M, f) \quad (2)$$

Candidate subsets are scored by LightGBM on validation data. For each task, loss is taken relative to that task's local subset and normalised by the performance headroom that remains. The minimax objective is the larger of the two losses.

For each dataset variant and selector, a single λ is chosen to minimise this objective, averaged over the first ten frozen seeds and all registered budgets. Exact ties go to 0.50 first and then to the smaller λ. Once locked before testing, the weight is applied to both classifiers.

On CICIoT2023, validation-based selection uses deterministic caps of 1,500 training and 500 validation observations for each multiclass class. Final fitting and testing, however, use the complete frozen partitions. Edge-IIoTset and ToN-IoT are validated on all their validation data.

3.7 Outcome Measures and Statistical Analysis

The main metrics are binary F1 (Attack as positive class) and multiclass Macro-F1. The Matthews correlation coefficient (MCC) and balanced accuracy are used for secondary checks. H1 and H2 are rerun post hoc with these secondary metrics, as the "everything is an attack" rule would inflate the F1 score in so many cases. Class recall and test support are also reported for multiclass outputs.

Fitting times and prediction times are only those of the actual computing environment. These are not to be interpreted as estimates of edge latency or energy consumption without accounting for hardware-specific considerations.

The concepts of transfer regret in the target task, and recall regret for each class are defined by Equations (3)–(5). A positive regret will be a harm. A value of zero indicates equal performance and a value less than zero indicates improvement.

$$TR(B \to M) = MacroF1(M)(S(M)) - MacroF1(M)(S(B)) \quad (3)$$

$$TR(M \to B) = F1(B)(S(B)) - F1(B)(S(M)) \quad (4)$$

 ISSN:2302-8149
e-ISSN:2540-9719

$$RecallRegret(c) = Recall(c)(S(M)) - Recall(c)(S(B)) \quad (5)$$

Absolute regrets in binary F1 and multiclass Macro-F1 are not directly comparable across directions. For this reason H2 relies on headroom-normalised transfer regret (HNTR; Equation (6)), in which P(local,t) is the score of the local reference in target task t.

$$HNTR(t) = \frac{TR(t)}{max(1 - P(local,t),10^{-6})} \quad (6)$$

If local headroom falls below 0.01, the observation is treated as ceiling-limited. Its absolute regret is kept, but it does not enter H2. A negative paired difference D(s), as defined in Equation (7), supports H2. A positive one points the other way.

$$D(s) = HNTR(M \to B, s) - HNTR(B \to M, s) \quad (7)$$

Equation (8) gives the joint worst-task loss. The endpoint losses are W(B) = max(0, HNTR(B→M)) and W(M) = max(HNTR(M→B), 0). Equations (9) and (10) define the better endpoint and the joint improvement. When Δ(J) is positive, S(J) is favoured; when it is negative, the endpoint is favoured; and $|\Delta(J)| \leq 1 \times 10^{-12}$ is treated as a computational tie.

$$W(J) = max(HNTR(J,B), HNTR(J,M)) \quad (8)$$

$$W(E) = min(W(B), W(M)) \quad (9)$$

$$\Delta(J) = W(E) - W(J) \quad (10)$$

Whether two subsets are identical is decided by comparing unordered sets of original predictor names. Four categories result: all subsets identical, S(J) = S(B) only, S(J) = S(M) only, and S(J) distinct from both.

For each configuration the 30 differences are summarized in terms of the mean, median, deviation standard, harm rate, 95% mean interval and rank biserial correlation. The frozen analysis is preserved by two sided Wilcoxon signed rank tests [37] [38] and split overlap is adjusted for by a ratio of test to training size of 0.15/0.70, using Nadeau-Bengio corrected resampled t tests [39]. Level of significance: α = 0.05. Statistical significance is not interpreted in terms of practical importance. Losses of less than 0.01 are practically small, losses of 0.01 or more are practically relevant, and losses of 0.05 or more are significant.

For H1, the regret is evaluated against zero for each selector, budget and classifier, and Holm correction [40] is individually applied to each data set variation and direction. The significance levels for comparison families in this section is determined by the Holm adjusted p values. After conducting the full grid analysis, the H1 summary across datasets was defined to give a common denominator. It includes the basic variables for k = 5, 10, 15 and 20, resulting in 64 comparisons (one way) per dataset. The sensitivity results are presented separately.

H2: a seed is added to D(s) if both directional observations have headroom ≥0.01. For those eligible pairs these Wilcoxon tests are performed and Holm correction is used for the different eligible configurations in every variant. A negative median which remains significant after Holm correction is in favor of H2 and a positive one is against H2. If the configuration does not win then it is labeled nonsignificant. These three counts are reported for each dataset verdict and descriptive HNTR means do not replace the tests. Absolute regret and harm rate complement HNTR and direction checks with MCC and balanced accuracy are reported in Section 4.2.

Under H3, class-recall regret is first averaged within each seed across the registered selectors, budgets and classifiers. Paired Wilcoxon tests then compare the 30 class-level losses with zero. Holm correction is applied across classes within each variant.

Spearman correlation links mean regret with the size of each attack class, benign traffic excluded. H3 predicts a negative correlation. The valid counts before capping are used for a post hoc analysis, as 23 of the 33 attack classes are capped at the same test support in CICIoT2023. Support is reported for each class result for recalling changes in large steps for small classes. Families of less than 1,500 eligible rows continue to be post hoc and descriptive comparisons.

H4 displays the rates of improvement, tie and deterioration in all cells and, post hoc, in sub-sets that are different from both parents for Δ(J). This separation will ensure that no reproduced endpoint at λ= 0 or 1 will be considered a benefit of fusion.

 ISSN:2302-8149
e-ISSN:2540-9719

3.8 Exploratory Subset Overlap Analysis

Jaccard overlap (Equation (11)) and its Spearman association with regret are examined descriptively only. Neither can stand in for an evaluation of predictive transfer.

$$J(S(B), S(M)) = \frac{|S(B) \cap S(M)|}{|S(B) \cup S(M)|} \tag{11}$$

As k/p rises, expected overlap rises mechanically. A post hoc chance reference is therefore built from two independently sampled uniform k-subsets. Their intersection X is distributed as in Equation (12), where C(a,b) denotes a binomial coefficient and x = max(0, 2k − p), …, k. The expected random Jaccard is given by Equation (13).

$$P(X = x) = \frac{C(k,x) \cdot C(p-k, k-x)}{C(p,k)} \tag{12}$$

$$E[J(random)] = \Sigma_x \frac{x}{2k-x} \cdot P(X = x) \tag{13}$$

The reference depends on p and k alone and draws on no model outcomes. It is not a prespecified hypothesis. Its purpose is to separate overlap that reflects the tasks from overlap produced simply by large budgets.

3.9 Shortcut Audit and Sensitivity Evaluation

Diagnostic sensitivity variants are defined by screening each value v in audits of predictor values. When these audits took place relative to the primary and affected sensitivity evaluations is set out in Section 3.1 and below. Support is the number of rows that carry v. Precision is max_c P(C=c|v), and coverage is P(v|C=c*) for the dominant class; the zero-class count records the classes in which v never occurs. A sensitivity analysis becomes mandatory when support ≥300, precision ≥95% and coverage ≥10%, provided the value appears in every partition of all 30 splits.

Bias-corrected Cramér's V and cardinality are reported descriptively and cannot on their own trigger a sensitivity analysis. The correction and the resulting V are defined in Equation (14), with $\varphi^2 = \chi^2/n$, n the number of observations, and r and c the dimensions of the contingency table. The subscript corr marks the bias correction. V = 0 is assigned to degenerate tables, meaning those with n ≤ 1 or with a corrected denominator that is not positive.

$$\begin{gathered} \varphi^2_{\text{corr}} = max\left(0, \varphi^2 - \frac{(r-1)(c-1)}{n-1}\right) \\ r_{\text{corr}} = r - \frac{(r-1)^2}{n-1}, \quad c_{\text{corr}} = c - \frac{(c-1)^2}{n-1} \\ V_{\text{corr}} = \sqrt{\frac{\varphi^2_{\text{corr}}}{\min(r_{\text{corr}} - 1, c_{\text{corr}} - 1)}} \end{gathered} \tag{14}$$

This rule was in place before the ToN-IoT and CICIoT2023 audits. Inspection of the Edge-IIoTset MQTT fields came before testing, and those fields later satisfied the same rule. Section 4.3 presents the audit results.

The sensitivity variants use the same rows and frozen splits as the primary analysis. Dropping predictors alters the full ranking and the Borda scale. Rankings are therefore recomputed, and a new joint weight is chosen on validation data for every feature space. Only common budgets are compared.

For Edge-IIoTset, the change in HNTR from the primary to the no-MQTT variant is paired by seed within each selector, budget, classifier and direction. A reduction is identified when two-sided Wilcoxon tests, Holm-corrected by direction, show the median sensitivity-minus-primary change to be significantly negative. Comparisons cover common budgets only.

The CICIoT2023 sensitivity analysis comprises 30 × 4 × 2 × 2 × 6 = 2,880 logical evaluations. A primary result is reused only if the ordered predictors and the entire fitting context agree, namely seed, selector, budget, subset, target and classifier. On that basis, 1,112 cells qualify. A further 128 cells are covered by caching identical fits within the sensitivity run. That leaves 1,640 new fits.

3.10 Reproducibility and Test Governance

Algorithm 1 gives the order of execution. In all conditions the same frozen rows and split files are employed. No testing is performed until encoders/rankings are trained on training data and λ is

 ISSN:2302-8149
e-ISSN:2540-9719

fixed on validation data. There is an incremental record for each configuration which is not refitted when the metrics, timing, recall and support are changed.
Recorded software versions: Python 3.10.11, pandas 2.3.3 and NumPy 1.26.4. It is not possible to reproduce the rest of the environment and/or the timing of the past precisely, because there is no full dependency lock or hardware profile. No guarantee is provided that two software versions or two machines will be bitwise equal.

4 Results and Analysis

The findings are reported based on 30 paired splits. Results are reported across 30 pair splits unless otherwise stated. Inference follows Section 3.7. The Table 5 below summarizes the hypothetical beliefs. All of H1-H4 convey information crucial for decision making of shared pipelines of the IoT communication. With H1, harm is caused based on configuration which dependence on reuse. H2 shows that there is no transfer direction that is safer in all cases. However, under H3 rarity was not always a good predictor of the loss of recall in attack families and so each family should be checked. Testing shows that H4 is not suitable to decrease the loss on the worst task and does not improve over the better endpoint when it is just able to duplicate a parent subset. Therefore, this shared selection should be validated in both label views against an explicit loss tolerance.

Table 5 Hypothesis verdicts across datasets. H1 uses the common primary k = 5–20 grid.

Hypothesis	Edge-IIoTset	ToN-IoT	CICIoT2023
H1 — practically significant transfer loss	Wilcoxon 18/64; corrected 17/64	Wilcoxon 17/64; corrected 15/64	Wilcoxon 24/64; corrected 22/64
H2 — M→B lower HNTR than B→M	Mixed: 11 / 8 / 21; corrected 10 / 3 / 27	Eligible subset only: 8 / 0 / 0 under both tests	Mostly opposite: 3 / 23 / 6; corrected 1 / 17 / 14
H3 — rarer families lose more recall	Not supported	Not supported; low power	Not supported; pre-cap association positive
H4 (worst-loss component) — S(J) lowers worst-case loss	Not supported	Not supported	Not supported

4.1 Feature Subset Divergence Across Label Granularities

With k = 5, ANOVA F produced disjoint subsets on Edge-IIoTset but the highest overlap on ToN-IoT, whereas chi-square produced disjoint subsets on CICIoT2023 (Figure 2). The ordering of selectors thus shifted from one dataset to the next.

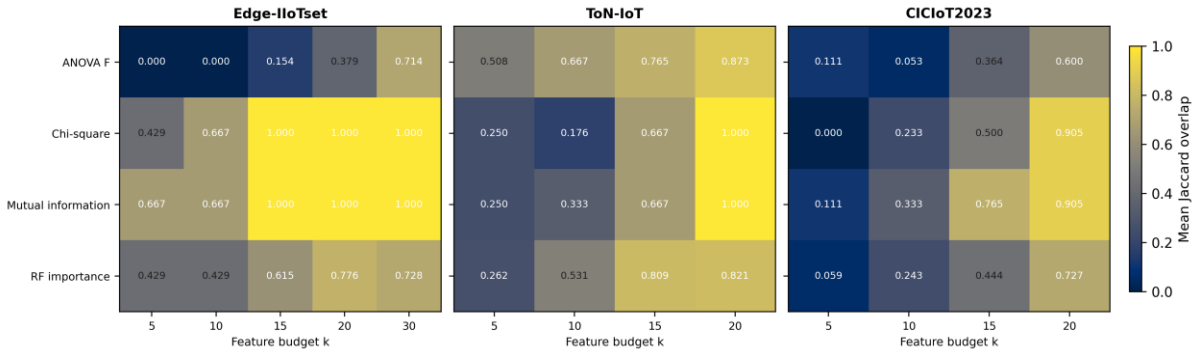


Figure 2 Mean Binary–multiclass jaccard overlap by selector and budget. edge-IIoTset k = 30 is a near-saturation control; comparisons use k ∈ {5, 10, 15, 20}.

For Edge-IIoTset, the separation produced by ANOVA F at k = 5 and 10 held in all 30 seeds, with a standard deviation of zero. Of the binary-selected predictors, MQTT fields made up four of five and nine of ten, respectively. tcp.seq supplied the rest. The multiclass subsets contained no MQTT fields and relied instead on ICMP, UDP, DNS and TCP fields.

 ISSN:2302-8149
e-ISSN:2540-9719

Chi-square and mutual information converged on identical subsets at k = 15 on Edge-IIoTset and at k = 20 on ToN-IoT, so transfer loss was zero by construction there. On CICIoT2023, identity was never reached, not even at the largest budget (Figure 2).
When the ToN-IoT DNS flags were removed, overlap increased in 9 of 16 selector–budget cells, decreased in 4 and stayed the same in 3. ANOVA F was involved in three of the decreases. The RF-importance decrease at k = 5 was negligible (−0.006). On Edge-IIoTset, removing MQTT lifted ANOVA F overlap at k = 5 from 0.000 to 0.111.
Stability was quantified as the mean, across seeds, of each feature's rank standard deviation. RF importance was the least stable selector in every binary view and in both CICIoT2023 views. In the multiclass view, though, ANOVA F was less stable than RF importance on Edge-IIoTset (0.793 versus 0.714) and on ToN-IoT (0.654 versus 0.593).
Measured against the post hoc chance reference in Figure 3, Edge-IIoTset ANOVA F stayed below chance at k = 5, 10 and 15, sat effectively at chance at 20, and matched it at 30.

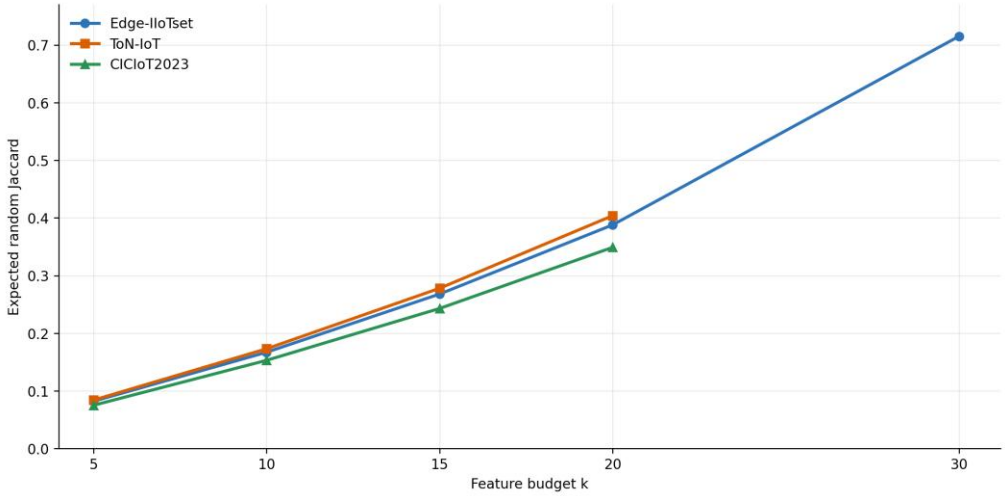


Figure 3 Post hoc expected jaccard overlap between two uniformly sampled random subsets of equal size.

At k = 5 on CICIoT2023, chi-square and RF importance dropped below the chance reference of 0.075. ANOVA F and mutual information exceeded it. At k = 10, ANOVA F reached roughly one third of the 0.153 reference. Figure 2 further reveals nonmonotonic overlap for chi-square on ToN-IoT, ANOVA F on CICIoT2023 and RF importance on Edge-IIoTset.
Li et al. [4] examined both label views but did not transfer subsets. The harm found here in both directions points to different losses of family-specific and benign–attack information. These mechanisms, however, remain a matter of interpretation.
Across ToN-IoT, Edge-IIoTset and CICIoT2023, M→B harm grew with class count (Table 1). There were also differences in prevalence, schemas and taxonomies between datasets. Thus, it is not possible to conclude that this pattern is due to class counts only. It would require a controlled experiment to separate the effect, in which the labels would be made coarser. This is also an extension of earlier work of comparing selectors based on a single dataset and a single classifier [5]. The rankings of selectors obtained under controlled conditions from one data set cannot be extrapolated to other data sets.
In Figures 2 and 3, the overlap that is actually observed is separated from the overlap that is only expected due to the large value of k/p. These numbers should be interpreted in conjunction with predictive regret. Information can be carried in two different subsets, and replacing just a few predictors can have a large loss. It is not enough to have an agreement on structure to share a feature pipeline.

4.2 Transfer Loss and Directional Asymmetry

The H1 verdicts follow Section 3.7 and Table 5. Significant improvements were seen in 10, 25 and 17 configurations on Edge-IIoTset, ToN-IoT and CICIoT2023, respectively.

 ISSN:2302-8149
e-ISSN:2540-9719

Support for H2 appeared only in the non-ceiling subset of ToN-IoT. Of 960 F1 configuration-seed observations, the guard excluded 750, removing every observation in 24 of 32 configurations. The eight eligible configurations, with between 7 and 30 contributing seeds, all supported H2 under both tests. On Edge-IIoTset and CICIoT2023, every configuration entered confirmatory inference.
Table 5 contrasts conventional and dependence-corrected H2 counts; Figure 4 shows the conventional counts and descriptive HNTR means. Corrected inference accounts for overlapping splits.

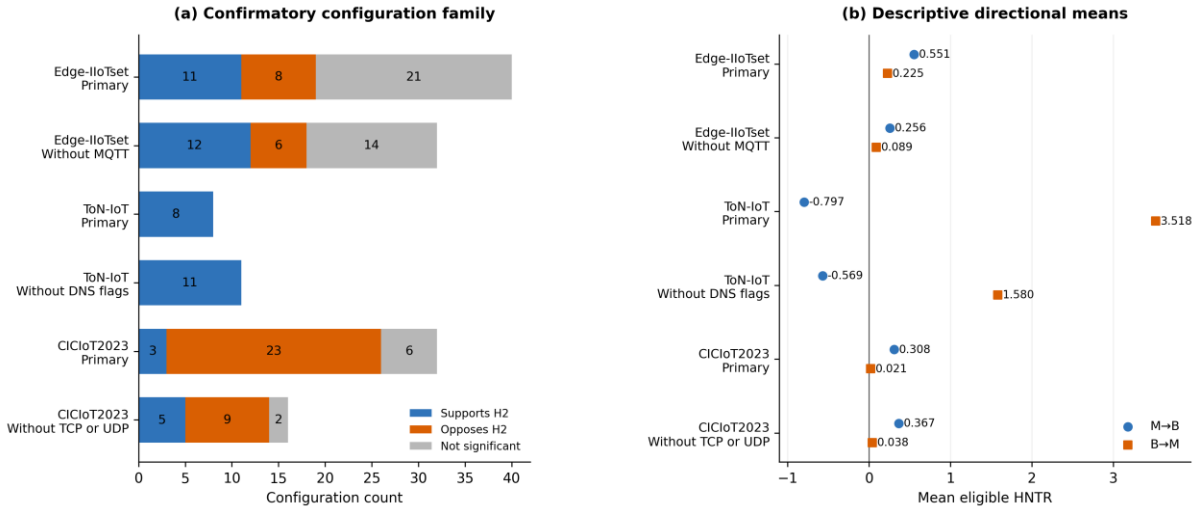


Figure 4 H2 Results: (a) Holm-adjusted wilcoxon counts; (b) eligible mean HNTR by direction. dependence-corrected counts are reported in Table 5.

Mean absolute regret for the M→B and B→M directions was 0.021 and 0.050 on Edge-IIoTset. On ToN-IoT the values were −0.005 and 0.069, and on CICIoT2023 they were 0.016 and −0.003. Harm rates, in the same directional order, were 41.3% and 54.5%, 5.7% and 83.5%, and 91.6% and 37.2%, respectively.
Under MCC (89.0%) and balanced accuracy (74.4%), M→B harm on CICIoT2023 was still frequent. For ToN-IoT, ceiling exclusions dropped from 78.1% under F1 to 44.6% under MCC and 65.5% under balanced accuracy (Figure 5).
The single largest mean loss came from ANOVA F at k = 5 on ToN-IoT under B→M transfer. Macro-F1 regret reached 0.446 with RF and 0.442 with LightGBM.
On Edge-IIoTset, the largest loss also involved ANOVA F at k = 5, this time under M→B transfer. Binary F1 regret was 0.329 with RF and 0.324 with LightGBM, and harm occurred in all 30 seeds. Without MQTT, these losses shrank to 0.294 and 0.284, respectively.
In the matching CICIoT2023 M→B configuration, losses were 0.109 with RF and 0.079 with LightGBM, in all 30 seeds. Since neither subset included TCP or UDP, removing them did not change these losses. The largest B→M loss on this dataset came from chi-square with RF at k = 5. It was 0.229 in all 30 seeds, with a range of 0.225 to 0.233. Next was RF importance with RF, at 0.164. ANOVA F with RF, in contrast, increased Macro-F1 by 0.184, bringing down the mean B→M regret to −0.0028, despite the frequent occurrence of harmful observations. On CICIoT2023, HNTR changes the landscape. For ANOVA F with RF = 5, local binary F1 = 0.950, meaning that there was approximately 0.050 of headroom, and a loss of 0.109 gave a mean HNTR of 2.17. Then for the chi-square with RF, the local Macro-F1 was 0.474, which leaves about 0.526. A loss of 0.229 there gave an HNTR of 0.44. That is, it was not the bigger loss, but rather the bigger loss normed.

 ISSN:2302-8149
e-ISSN:2540-9719

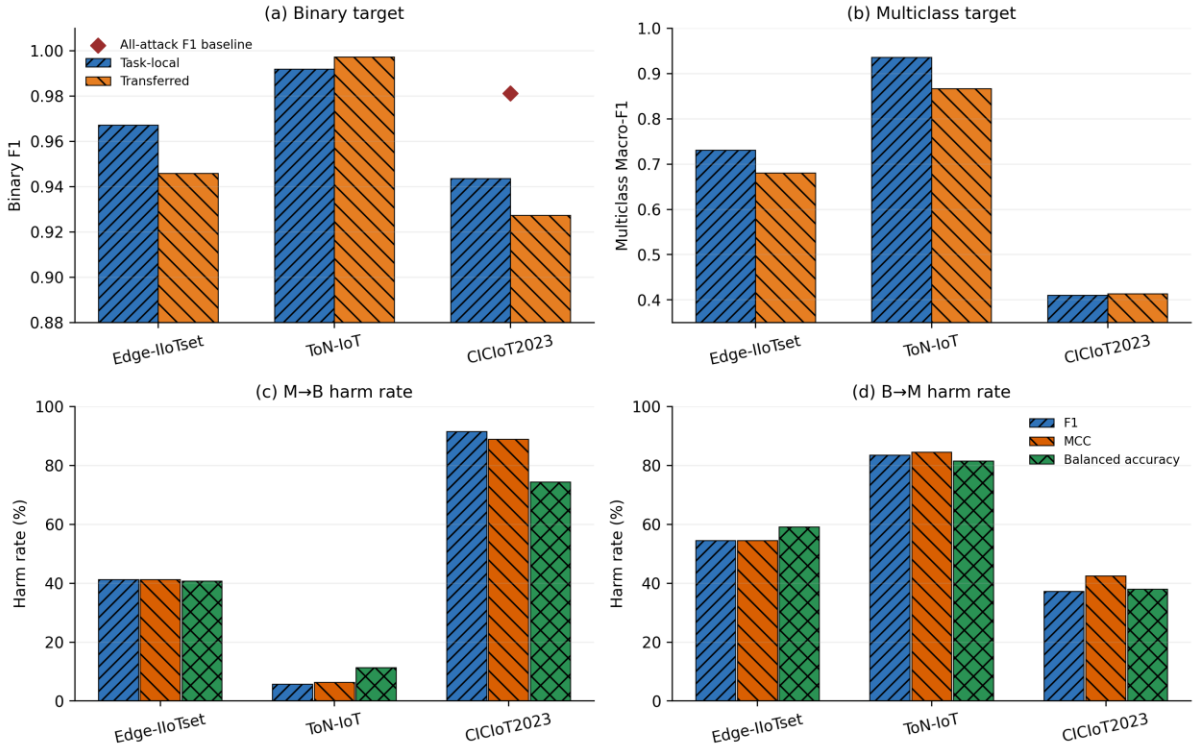


Figure 5 Absolute primary scores and metric robustness on the common k = 5–20 Grid.

Across all 30 splits in the sensitivity analysis, the aggregate HNTR differences had the same sign. With 30 nonzero pairs, Wilcoxon $p = 1.86 \times 10^{-9}$ is the lowest possible value. Because the splits overlap, the corrected tests carry greater inferential weight.

Harmful configurations that remained significant after correction showed rank-biserial effects of at least 0.996, with a median of 1.000. For RF at k = 5, the corrected 95% intervals were 0.327–0.331 for Edge-IIoTset ANOVA F under multiclass-to-binary transfer, 0.288–0.604 for ToN-IoT ANOVA F under binary-to-multiclass transfer, and 0.227–0.231 for CICIoT2023 chi-square, also under Binary-to-multiclass transfer.

Li et al. [4] reported both F1 and MCC. In the present experiment, MCC and balanced accuracy preserved the main dataset-level directional pattern and reduced the amplification caused by offset binary F1.

The direction that suffers more likely depends on two factors: what the labels reward and how much headroom the target task retains. Binary selection favors predictors that separate pooled attacks from benign traffic. Multiclass selection favors the same separation, along with distinctions among attacks. On Edge-IIoTset, the binary ANOVA F subset retained MQTT coverage absent from the multiclass subset. On ToN-IoT, both subsets appear to keep the binary boundary intact, yet binary selected predictors lose attribution information and ceiling effects constrain directional inference. While, in CICIoT2023, the capped, attack-dominated distribution inflate normalized binary loss, opposing multiclass changes offset one another. The interpretations are still subject to change. The number of shortcuts was never changed without changing the schema and taxonomy, and the main failures remained after removing the shortcuts.

There is a possibility of incompatibility with reuse, but the results from H1 do not indicate that any reuses are harmful, as some reuses have been improved. Both of these trends were not generally favored in the H2 results and the support found on ToN-IoT applies only to the group below the ceiling. If the pipeline is shared, absolute loss should be evaluated as well as metrics specific to the particular target task. Failure in any one configuration can be concealed by the single mean score or directional average.

 ISSN:2302-8149
e-ISSN:2540-9719

4.3 Testbed Shortcuts and Sensitivity Results

Each dataset held a shortcut linked to its communication protocols (Table 6). Because the audits cover all sections under Section 3.9, the sensitivities are diagnostic analyses. They are not estimates drawn from an untouched test set.

On Edge-IIoTset, MQTT activity appeared only in benign records (Table 6). None of the 132,685 attack records, which span 14 families, contained an active MQTT field. In CICIoT2023, the protocol ratios reflected fixed scenario mixtures. A TCP fraction of 0.49 was absent from 20 classes; its predictive precision is reported in Table 6.

The four DNS flags (authoritative answer, recursion desired, recursion available and query rejected) shared a common applicability signal, with V = 0.569. All four were removed so that the signal could not persist through any remaining flag.

There was a protocol-dependent association strength. Applicability V was 0.682 when looking at the service, and 0.087 for SSL (and 0.015–0.060 for HTTP), all with the same Zeek marker. No service value met the shortcut rule so service is maintained. The sensitivity analysis is then not just a matter of whether or not a DNS connection is recognised, but rather about the DNS flags.

The primary shortcomings were not eliminated. When MQTT is removed, 25 out of 64 common Edge-IIoTset configurations showed significant reduction in HNTR. The ANOVA F subsets at small budgets were already excluded from TCP and UDP on CICIoT2023 when transferred from multiclass to binary. The biggest loss in ToN-IoT with removing the DNS flags is 0.270377 (rounded off to 0.270) and 0.270378 (rounded off to 0.270) with RF and LightGBM respectively. Aggregate regret fell from 0.0689 to 0.0441. These shortcuts reduced the magnitude of the losses but did not account for all of the loss of transfer.

Previous inspections of benchmarks identified errors in construction and processing as a cause of bias [11] and [12]. This study extends this work to the individual predictor values, and reveals three mechanisms. They include protocol coverage, field applicability and fingerprints of fixed protocol mixtures. They are all related to how the testbed changes protocols on, loads its fields or repairs its traffic mixtures.

Table 6 Shortcut Mechanisms, triggering values and sensitivity scope.

Dataset	Mechanism and trigger	Support	Precision	Coverage	Sensitivity scope
Edge-IIoTset	Protocol coverage: any active MQTT field; 5,063 Normal rows and 0 attack rows	5,063	100.00%	20.86%	Remove ten MQTT predictors
ToN-IoT	Field applicability: dns_RD = F; 4,791 of 4,958 occurrences were Normal; same signal in dns_AA, dns_RA and dns_rejected	4,958	96.63%	15.97%	Remove the four DNS flags
CICIoT2023	Generation fingerprint: TCP = 0.49	6,800	98.9%	22.4%	Remove TCP and UDP
CICIoT2023	Generation fingerprint: TCP = 0.51	3,097	98.4%	10.2%	Remove TCP and UDP
CICIoT2023	Generation fingerprint: UDP = 0.48	3,439	98.5%	11.3%	Remove TCP and UDP
CICIoT2023	Generation fingerprint: UDP = 0.49	7,604	99.3%	25.2%	Remove TCP and UDP

These findings do not represent the true level of this practice among IoT data. They do specify that removing identifiers and payloads is not enough. The conditional value distributions and protocol coverage for classes should also be investigated. It does not however imply that all MQTT, DNS or protocol indications should be removed from all operational networks as the significance of these fields relies on the fact that the observed associations are valid in the targeted deployment. Weaker or unmeasured artefacts might still be present.

4.4 Attack Family Effects

H3 found no support. Edge-IIoTset showed a weak negative association. ToN-IoT showed a weak positive one, though it had only nine attack classes. In CICIoT2023, 23 of the 33 attack classes

had the same capped test support of 4,500. Switching to pre-cap support moved ρ from 0.137 to 0.378 in the primary analysis and from 0.063 to 0.349 without TCP/UDP. The raw p values, 0.030 and 0.047, both became 0.060 after Holm adjustment (Figure 6). A positive direction runs counter to the predicted rarity effect.

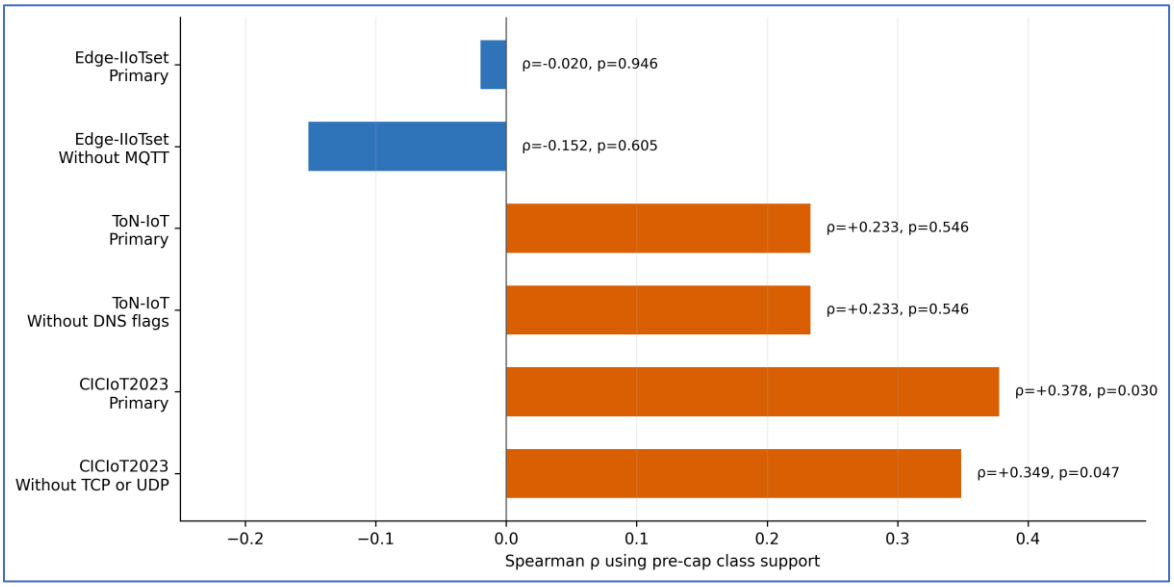


Figure 6 Pre-cap Attack-class support versus mean B→M Recall loss: spearman association.

On Edge-IIoTset, Fingerprinting recall fell from 0.429 to 0.256, giving a regret of 0.174 (Holm $p = 2.8 \times 10^{-8}$). MITM lost 0.033 across all 30 seeds ($p = 1.2 \times 10^{-5}$). Since this class had approximately 60 test records, each record moved recall by about 1.7 points, so the loss corresponds to approximately two records per split. Fingerprinting continued to lose recall after MQTT removal, by 0.118 (0.371 to 0.253; $p = 1.0 \times 10^{-5}$). The change for MITM was nonsignificant (−0.0025; $p = 0.104$). On ToN-IoT, MITM had about 156 test records. Its regret was −0.001 in the primary variant and −0.004 after removing of the DNS flags.

Among the CICIoT2023 families shown in Figure 7, TCP/UDP removal reduced recall loss for three and increased it slightly for two. Their capped supports are identical, which masks differences in original class size.

Post hoc comparisons of low-support families, defined as those with fewer than 1,500 eligible rows, did not agree across datasets. On Edge-IIoTset, Fingerprinting and MITM lost more recall than the remaining families in all 30 seeds of both variants (each $p = 1.86 \times 10^{-9}$). On CICIoT2023, Uploading_Attack lost more in 26 of 30 seeds ($p = 1.6 \times 10^{-7}$) and in 23 of 30 without TCP/UDP ($p = 2.1 \times 10^{-4}$). The ToN-IoT pattern ran the other way: MITM showed lower regret in every seed of both variants. Such contrasts are descriptive only. Both are based on one or two families per data set.

In the current experiment, the global rankings are transferred while in Mchina et al. [6] Minority-class predictors are protected during selection process. Rarity alone was not responsible for the losses as there was no support for H3. It doesn't mean that rare attacks were safe. This overall association is shown in Figure 6. The damage is unevenly distributed among classes of similar support as seen in Figure 7 and the family-level contrasts. Therefore, recall and support should be assessed for every pertinent family in practice. In ToN-IoT, due to the few classes of attacks and the few families with low support, no general conclusion about rarity can be drawn.

 ISSN:2302-8149
e-ISSN:2540-9719

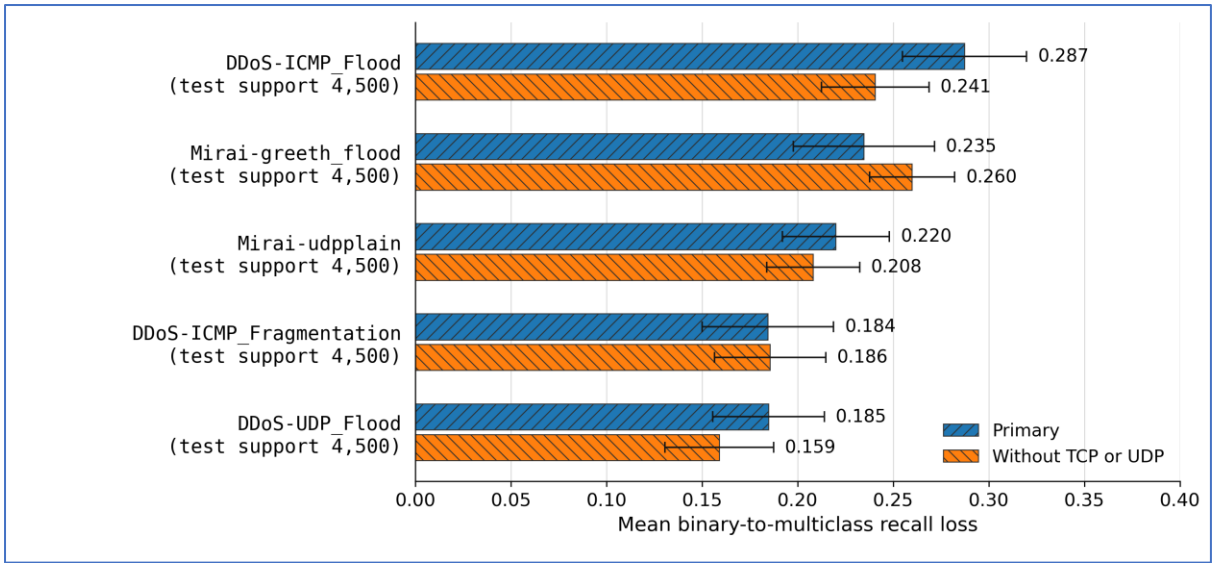


Figure 7 Largest CICIoT2023 Mean B→M Recall Losses at k = 5 and 10. Descriptive 95% t intervals are uncorrected for split overlap; each class has capped test support 4,500. Literal result-file labels and hatching distinguish classes and variants.

4.5 Joint Feature Subset Performance

The Borda fusion that was tested did not reliably lower worst-task loss, so the evaluated component of H4 is unsupported. Figure 8 separates subsets that match a parent from those that differ from both.

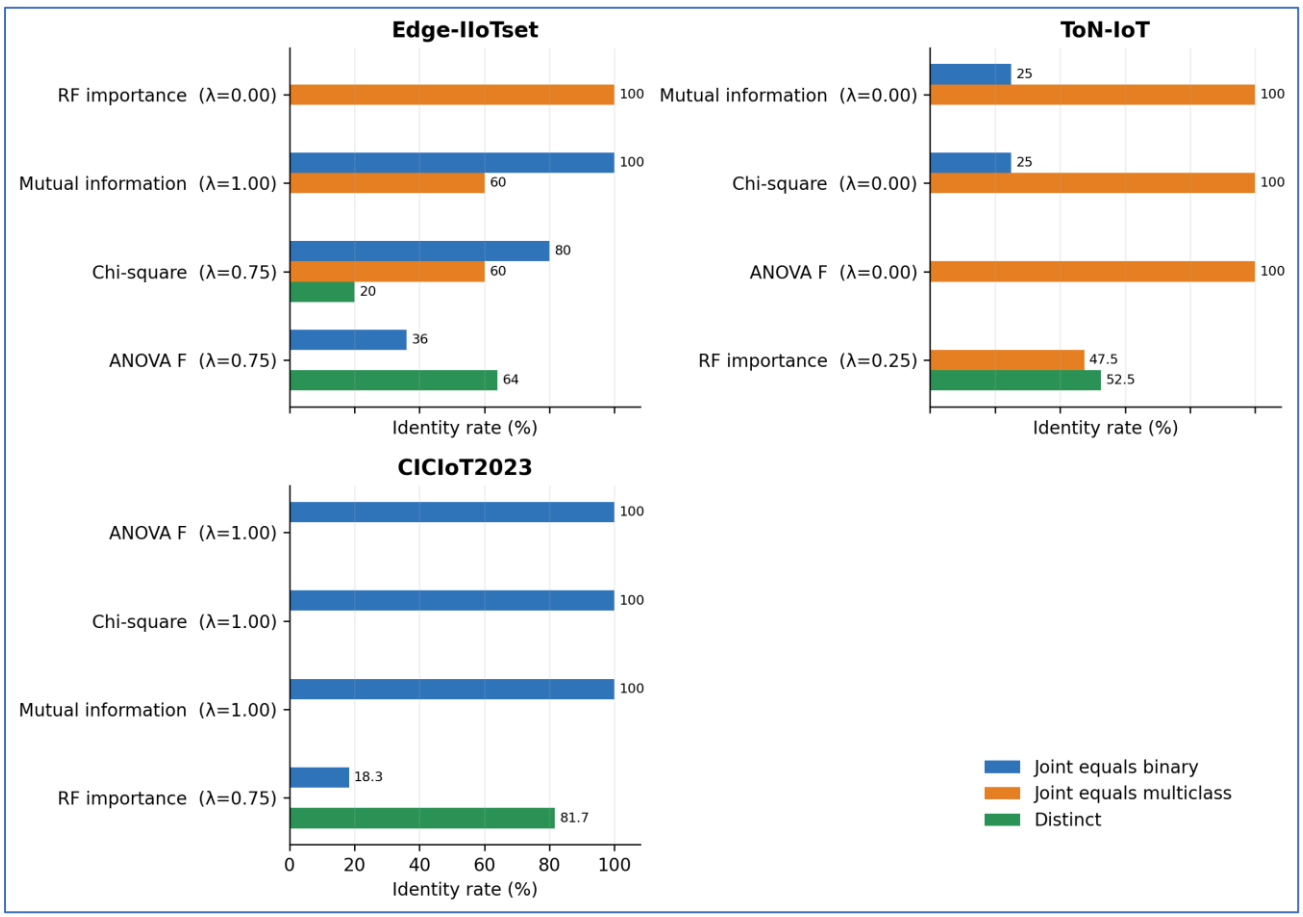


Figure 8 Joint-Subset identity rates relative to binary and multiclass parents.

 ISSN:2302-8149
e-ISSN:2540-9719

The joint subset beat the better endpoint in 97 of 1,200 Edge-IIoTset cells, in 36 of 960 ToN-IoT cells and in 100 of 960 CICIoT2023 cells. Endpoint reproduction and parent identity dominate these rates.
Restricting attention to distinct subsets, the rates of improvement, ties and deterioration on Edge-IIoTset were 38.5%, 0.0% and 61.5% (252 cells). On ToN-IoT they were 28.6%, 1.6% and 69.8% (126 cells). On CICIoT2023 they were 51.0%, 0.0% and 49.0% (196 cells).
In the sensitivity variants, the matching rates were 34.6%, 0.0% and 65.4% without MQTT, 29.2%, 0.0% and 70.8% without DNS flags, and 33.0%, 0.0% and 67.0% without TCP/UDP. Removing the shortcuts did not lift the caution about fusion.
None of the eight endpoint choices yielded a distinct subset, while all four interior choices did. Some of the advantages on validation data were negligible. For Edge-IIoTset mutual information, for instance, $\lambda = 1.00$ improved the minimax objective over $\lambda = 0.75$ by only about 3.7×10^{-5}. Choosing an endpoint, or winning by a near tie on validation data, is not evidence that fusion helps.
Class-aware techniques [6, 7] are used to preserve information relevant to individual categories while label-aware selection and sampling [26] are used to improve representation. The baseline included here is unique. It combines the ranks acquired separately using the two label views. The results from the distinct-subset test for the Borda construction tested do not reveal any reliable advantage. These results have not been tested with multi-objective or Pareto methods that explicitly include constraints on the worst task, nor do they exclude all types of shared selection. If the deployability is evaluated [31] reports should be given the six-cell matrix, the class recall and support, the budget, the overlap, the absolute regret and the harm rate. Any recommendation is always conditional on the dataset, target task, classifier and budget.

4.6 Limitations

All evidence comes from three controlled public testbeds, which cannot be regarded as a representative sample of operational IoT networks. Row-level splitting may distribute records or windows from a single capture across several partitions, particularly in CICIoT2023. Local and transferred scores could both be optimistic as a result. Pairing leaves this shared capture structure in place. Magnitudes and directional ordering might therefore shift under capture-grouped, time-separated or external-site evaluation.
The prevalence of class caps varies in deployment and taxonomies vary by data set. The corrected paired tests, therefore, do not convert samples to operational false alarms or incident risk, and the samples as constructed determine whether the results are eligible for inclusion in an F1 or are excluded because they do not meet the eligibility criteria.Ceiling eligibility and HNTR, thus, depend on the samples as constructed, and the corrected paired tests are not directly applicable to converting samples to operational false alarms or incident risk. The reusability of a resource is determined across three test beds; but there is no universal direction of transfer. Other tree ensembles and other selection objectives are not covered by other model families and/or not examined.
The common λ was picked using the first 10 overlapping splits and then used for both classifiers. This selection was also done on capped samples for CICIoT2023. Because rows that are used for validation in early splits can be put into test sets in later splits, fusion performance is estimated on repeated splits and not on an external test set. The H4 outcome may vary for other validation sets or for different weights trained for each classifier. The rates are partially mechanical over all of the cells since endpoints are reproduced. Conclusions are therefore limited to subsets as well as the Borda grid of five weights analyzed here.
The sensitivity analysis in the CICIoT2023 is limited to $k = 5, 10$. MQTT inspection was performed on Edge-IIoTset prior to threshold formaling, and all shortcut audits analyzed test partitions. The sensitivity analyzes are therefore a diagnosis of the current result, and not a diagnosis of any causal mechanism. For Zeek, the numeric conversion of its values can lead to the confusion of inapplicability with zero in the case of ToN-IoT. Loosely coupled or multivariate shortcuts can also be left behind. The reported scores should not be interpreted as a deployment score without artefacts.
Several configurations were left untested: full-feature and random-k baselines, attack-only attribution, binary decisions derived from multiclass outputs, and capture-grouped splits. Their prevents loss from being attributed with absence of full confidence to the evaluated design selection and leaves alternative pipeline constructions unexplored. The random-Jaccard comparison and the pre-cap H3 analysis are both post hoc.

 ISSN:2302-8149
e-ISSN:2540-9719

5 Conclusion

This paper proposes a paired comparison study for feature-subset reuse between binary detection and multiclass attribution using target local references and ceiling-aware regret and dependence corrected inference. The usefulness of a subset was shown to be influenced by two factors labeling in the ranking of the subset and the task the subset is assigned to. Both the up and down transfer direction were not found to be generally safer in the test beds evaluated.

Protocol value audits also show that the losses that are measured are influenced by protocol shortcuts. But they leave the major failures just partially explained. There was no consistent correlation between the rarity of attacks in the family and damage to the recall. Neither was there any consistency in the Borda fusion's effectiveness in reducing worst-task losses when parent reproductions were separated from subsets. All of these results combine to suggest that some sort of checking needs to be done to ensure that one feature pipeline is not used in place of task specific alternatives.

If there are features shared by a class, they should be validated for both label views against an explicit loss tolerance, in addition to their class-level recall and support being assessed, and compared to the aggregate scores. However, if this is not the case, it is still reasonable to keep task specific subsets. The extent to which these conclusions hold for operational IoT communication traffic is still open. To set up it will need capture grouped and external site experiments with full feature and random-k baselines and alternative shared selection objectives.

Data Availability

The public datasets can be obtained from the original data providers: Edge-IIoTset [32] from IEEE DataPort; ToN-IoT [33] from UNSW; and CICIoT2023 [34] from the Canadian Institute for Cybersecurity, at the University of New Brunswick. This study relies on the data sets: ML-EdgeIIoT-dataset.csv, Train_Test_Network.csv and the CICIoT2023 CSV release. The settings for cleaning rules and retained schemas, sampling caps, split seeds, ranking representations, and frozen classifiers are defined in Section 3.

References

[1] M. M. Rahman, S. A. Shakil, and M. R. Mustakim, "*A Survey on Intrusion Detection System in IoT Networks*," *Cyber Security and Applications*, Vol. 3, Art. No. 100082, 2025, DOI: 10.1016/j.csa.2024.100082.

[2] M. Fatima, O. Rehman, I. M. H. Rahman, A. Ajmal, and S. J. Park, "*Towards Ensemble Feature Selection for Lightweight Intrusion Detection in Resource-Constrained IoT Devices*," *Future Internet*, Vol. 16, No. 10, Art. No. 368, 2024, DOI: 10.3390/fi16100368.

[3] U. Otokwala, A. Petrovski, and H. Kalutarage, "*Optimized Common Features Selection and Deep-Autoencoder (OCFSDA) for Lightweight Intrusion Detection in Internet of Things*," *International Journal of Information Security*, Vol. 23, No. 4, pp. 2559–2581, 2024, DOI: 10.1007/s10207-024-00855-7.

[4] J. Li, M. S. Othman, H. Chen, and L. M. Yusuf, "*Optimizing IoT Intrusion Detection System: Feature Selection versus Feature Extraction in Machine Learning*," *Journal of Big Data*, Vol. 11, No. 1, Art. No. 36, 2024, DOI: 10.1186/s40537-024-00892-y.

[5] A. Sezgin, M. Ulaş, and A. Boyacı, "*Multi-Objective Feature Selection for Intrusion Detection Systems: A Comparative Analysis of Bio-Inspired Optimization Algorithms*," *Sensors*, Vol. 25, No. 19, Art. No. 6099, 2025, DOI: 10.3390/s25196099.

[6] J. P. Mchina, N. Mduma, and R. S. Sinde, "*Adaptive Class-Aware Feature Selection for High-Dimensional and Imbalanced Multi-Class Network Intrusion Detection*," *Frontiers in Big Data*, Vol. 9, Art. No. 1871346, 2026, DOI: 10.3389/fdata.2026.1871346.

[7] Y. Lu, W. Tian, D. Zhang, and K. Zhao, "*Attack Category Mapping Modeling Method based on System-Level Features for Intrusion Detection Systems*," *Information Sciences*, Vol. 755, Art. No. 123791, 2026, DOI: 10.1016/j.ins.2026.123791.

[8] G. Sakthivel and A. K. Kumarasamy, "*A Drift-Aware RS2FS Pipeline with Confidence Gating for IDS*," *Scientific Reports*, Vol. 16, No. 1, Art. No. 4029, 2026, DOI: 10.1038/s41598-025-34124-x.

[9] R. Sommer and V. Paxson, *"Outside the Closed World: on using Machine Learning for Network Intrusion Detection*," in Proc. *IEEE Symp. Security and Privacy*, 2010, pp. 305–316, DOI: 10.1109/SP.2010.25.

[10] G. Apruzzese, P. Laskov, and J. Schneider, "*SoK: Pragmatic Assessment of Machine Learning for Network Intrusion Detection*," in Proc. *IEEE 8th European Symp. Security and Privacy (EuroS&P)*, 2023, pp. 592–614, DOI: 10.1109/EuroSP57164.2023.00042.

[11] R. Flood, G. Engelen, D. Aspinall, and L. Desmet, "*Bad Design Smells in Benchmark NIDS Datasets*," in Proc. *IEEE 9th European Symp. Security and Privacy (EuroS&P)*, 2024, pp. 658–675, DOI: 10.1109/EuroSP60621.2024.00042.

[12] L. Liu, G. Engelen, T. Lynar, D. Essam, and W. Joosen, "*Error Prevalence in NIDS Datasets: A Case Study on CIC-IDS-2017 and CSE-CIC-IDS-2018*," in Proc. *IEEE Conf. Communications and Network Security (CNS)*, 2022, pp. 254–262, DOI: 10.1109/CNS56114.2022.9947235.

[13] D. Arp et al., "*Dos and Don'ts of Machine Learning in Computer Security*," in *Proc. 31st USENIX Security Symp. (USENIX Security 22)*, 2022, pp. 3971–3988. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/arp (accessed Oct. 31, 2026).

[14] S. Kaushik et al., "*Robust Machine Learning based Intrusion Detection System using Simple Statistical Techniques in Feature Selection*," *Scientific Reports*, Vol. 15, No. 1, Art. No. 3970, 2025, DOI: 10.1038/s41598-025-88286-9.

[15] M. A. Umar, Z. Chen, K. Shuaib, and Y. Liu, "*Effects of Feature Selection and Normalization on Network Intrusion Detection*," *Data Science and Management*, Vol. 8, No. 1, pp. 23–39, 2025, DOI: 10.1016/j.dsm.2024.08.001.

[16] G. Nassreddine, M. Nassereddine, and O. Al-Khatib, "*Ensemble Learning for Network Intrusion Detection based on Correlation and Embedded Feature Selection Techniques*," *Computers*, Vol. 14, No. 3, Art. No. 82, 2025, DOI: 10.3390/computers14030082.

[17] U. Ahmed et al., "*Hybrid Bagging and Boosting with SHAP based Feature Selection for Enhanced Predictive Modeling in Intrusion Detection Syst*ems," *Scientific Reports*, Vol. 14, No. 1, Art. No. 30532, 2024, DOI: 10.1038/s41598-024-81151-1.

[18] M. Awad and S. Fraihat, "*Recursive Feature Elimination with Cross-Validation with Decision Tree: Feature Selection Method for Machine Learning-based Intrusion Detection Systems*," *Journal of Sensor and Actuator Networks*, Vol. 12, No. 5, Art. No. 67, 2023, DOI: 10.3390/jsan12050067.

[19] J. Ding, J. Du, H. Wang, and S. Xiao, "*A Novel Two-Stage Feature Selection Method based on Random Forest and Improved Genetic Algorithm for Enhancing Classification in Machine Learning*," *Scientific Reports*, Vol. 15, No. 1, Art. No. 16828, 2025, DOI: 10.1038/s41598-025-01761-1.

[20] Y. K. Saheed and S. Misra, "*A Voting Gray Wolf Optimizer-based Ensemble Learning Models for Intrusion Detection in the Internet of Things*," *International Journal of Information Security*, Vol. 23, No. 3, pp. 1557–1581, 2024, DOI: 10.1007/s10207-023-00803-x.

[21] S. S. Alqahtany, A. Shaikh, and A. Alqazzaz, "*Enhanced Grey Wolf Optimization (EGWO) and Random Forest based Mechanism for Intrusion Detection in IoT Networks*," *Scientific Reports*, Vol. 15, No. 1, Art. No. 1916, 2025, DOI: 10.1038/s41598-024-81147-x.

[22] M. Benmalek and A. Seddiki, "*Particle Swarm Optimization-Enhanced Machine Learning and Deep Learning Techniques for Internet of Things Intrusion Detection*," *Data Science and Management*, Vol. 8, No. 4, pp. 423–435, 2025, DOI: 10.1016/j.dsm.2025.02.005.

[23] A. K. Dey, G. P. Gupta, and S. P. Sahu, "*A Metaheuristic-based Ensemble Feature Selection Framework for Cyber Threat Detection in IoT-Enabled Networks*," *Decision Analytics Journal*, Vol. 7, Art. No. 100206, 2023, DOI: 10.1016/j.dajour.2023.100206.

[24] S.-M. Tseng, Y.-Q. Wang, and Y.-C. Wang, "*Multi-Class Intrusion Detection based on Transformer for IoT Networks using CIC-IoT-2023 Data*set," *Future Internet*, Vol. 16, No. 8, Art. No. 284, 2024, DOI: 10.3390/fi16080284.

[25] C. Christy, A. Nirmala, A. M. O. Teena, and A. I. Amali, "*Machine Learning based Multi-Stage Intrusion Detection System and Feature Selection Ensemble Security in Cloud Assisted Vehicular Ad Hoc Networks*," *Scientific Reports*, Vol. 15, No. 1, Art. No. 27058, 2025, DOI: 10.1038/s41598-025-96303-0.

[26] M. N. Amr, A. Elliethy, T. Mekkawy, and A. Mahran, "*Sequential Submodular Feature-Sample Selection for Lightweight and Update-Efficient IoT Intrusion Detec*tion," *Scientific Reports*, Vol. 16, No. 1, Art. No. 26315, 2026, DOI: 10.1038/s41598-026-66614-x.

[27] M. B. Musthafa et al., "*Optimizing IoT Intrusion Detection using Balanced Class Distribution, Feature Selection, and Ensemble Machine Learning Tec*hniques," *Sensors*, Vol. 24, No. 13, Art. No. 4293, 2024, DOI: 10.3390/s24134293.

[28] C. Kılıç and G. Şengül, "*SHAP-Guided Feature Selection for Cross-Dataset Generalization in Network Intrusion Detection Systems*," *IEEE Access*, Vol. 14, pp. 90926–90936, 2026, DOI: 10.1109/ACCESS.2026.3703481.

[29] M. Sarhan, S. Layeghy, and M. Portmann, "*Evaluating Standard Feature Sets Towards Increased Generalisability and Explainability of ML-based Network Intrusion Detection*," *Big Data Research*, Vol. 30, Art. No. 100359, 2022, DOI: 10.1016/j.bdr.2022.100359.

[30] M. Wang, N. Yang, and N. Weng, "*K-GetNID: Knowledge-Guided Graphs for Early and Transferable Network Intrusion Detection*," *IEEE Transactions on Information Forensics and Security*, Vol. 19, pp. 7147–7160, 2024, DOI: 10.1109/TIFS.2024.3431932.

[31] M. M. Islam, U. Salsabil, M. Nurmamatov, and S. Hossain, "*Lightweight Intrusion Detection Systems for IoT–Edge Environments: A PRISMA-ScR Systematic Review of Deployability Evidence and a Unified Assessment Framework*," *Future Internet*, Vol. 18, No. 6, Art. No. 300, 2026, DOI: 10.3390/fi18060300.

[32] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "*Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning*," *IEEE Access*, Vol. 10, pp. 40281–40306, 2022, DOI: 10.1109/ACCESS.2022.3165809.

[33] N. Moustafa, "*A New Distributed Architecture for Evaluating AI-based Security Systems at the Edge: Network TON_IoT Datasets*," *Sustainable Cities and Society*, Vol. 72, Art. No. 102994, 2021, DOI: 10.1016/j.scs.2021.102994.

[34] 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, Art. No. 5941, 2023, DOI: 10.3390/s23135941.

[35] L. Breiman, "Random forests," *Machine Learning*, Vol. 45, No. 1, pp. 5–32, 2001, DOI: 10.1023/A:1010933404324.

[36] G. Ke et al., "*LightGBM: A Highly Efficient Gradient Boosting Decision Tree*," in *Advances in Neural Information Processing Systems*, Vol. 30, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html (accessed Oct. 31, 2026).

[37] F. Wilcoxon, "*Individual Comparisons by Ranking Methods*," *Biometrics Bulletin*, Vol. 1, No. 6, pp. 80–83, 1945, DOI: 10.2307/3001968.

[38] J. Demšar, "*Statistical Comparisons of Classifiers Over Multiple Data Sets*," *Journal of Machine Learning Research*, Vol. 7, pp. 1–30, 2006. [Online]. Available: https://www.jmlr.org/papers/v7/demsar06a.html (accessed Oct. 31, 2026).

[39] C. Nadeau and Y. Bengio, "*Inference for the Generalization Error*," *Machine Learning*, Vol. 52, No. 3, pp. 239–281, 2003, DOI: 10.1023/A:1024068626366.

[40] S. Holm, "*A Simple Sequentially Rejective Multiple Test Procedure*," *Scandinavian Journal of Statistics*, Vol. 6, No. 2, pp. 65–70, 1979, DOI: 10.2307/4615733. [Online]. Available: https://www.jstor.org/stable/4615733 (accessed Oct. 31, 2026).