

BEYOND MODEL COMPLEXITY: A DATA-CENTRIC APPROACH TO RELIABLE AI-DRIVEN CYBERSECURITY PREDICTION

Tehleel Basit^{*1}, Dr Eram Abbasi², Nazia Azim³, Hameed Hussain⁴

^{*1}University of Engineering and Technology Mardan, Pakistan

²Dept. CSIT, NED University, Karachi, Pakistan

³Lecturer department of computer science Abdul Wali Khan University Mardan

⁴Department of Computer Science, University of Buner, 19290, Khyber Pakhtunkhwa, Pakistan

^{*1}tehleelbasit78@gmail.com, ²eramabbasi@gmail.com, ³n.azim@awkum.edu.pk, ⁴dr.hameed@ubuner.edu.pk

DOI: <https://doi.org/10.5281/zenodo.22706152>

Keywords

data-centric AI; cybersecurity; intrusion detection; UNSW-NB15; CIC-IDS2017; class imbalance; data leakage; temporal validation; XGBoost; explainability; reliability

Article History

Received: 16 May 2026

Accepted: 12 June 2026

Published: 15 August 2026

Copyright @Author

Corresponding Author: *
Tehleel Basit

Abstract

Reliable AI-driven cybersecurity prediction depends not only on model architecture but also on how data are constructed, cleaned, balanced, split, and validated. This study develops a data-centric evaluation framework and tests it on two widely used intrusion-detection benchmarks, UNSW-NB15 and CIC-IDS2017. The primary UNSW-NB15 analysis retains the official 175,341-record training and 82,332-record testing partitions; additional uncertainty, intervention, and attack-level experiments use a fixed stratified analysis sample of 80,000 training and 40,000 testing records for computational reproducibility. For CIC-IDS2017, all eight machine-learning CSV files were audited to obtain exact full-file class counts, and a deterministic 103,245-record sample was used for controlled modeling. On UNSW-NB15, class-aware XGBoost increased F1 from 0.898 to 0.926 and reduced the false-positive rate from 25.6% to 14.3% in the controlled sample. Bootstrap analysis estimated an F1 improvement of 0.0278 (95% CI 0.0258-0.0298), while McNemar testing showed a highly significant change in paired classification errors ($p < 0.001$). Undersampling produced a comparable F1 of 0.926, and training-size analysis showed monotonic gains as usable data increased. Attack-level evaluation revealed that aggregate performance masked weaker recall for Fuzzers (0.743). On CIC-IDS2017, a random split yielded near-saturated F1 (0.996-0.997), whereas a Friday-afternoon temporal holdout reduced class-aware XGBoost F1 to 0.433 and attack recall to 0.277 despite ROCAUC of 0.918. This contrast demonstrates that apparent model excellence can be dominated by data partitioning and distribution shift. The results support a practical conclusion: trustworthy cybersecurity AI requires data provenance, leakage control, class-aware learning, uncertainty analysis, attack-level auditing, and temporal or cross-dataset validation before additional model complexity is interpreted as progress.

1. INTRODUCTION

Machine-learning intrusion detection is commonly framed as a model-selection problem: investigators compare classifiers or increasingly complex neural architectures and report the method with the highest benchmark score. That framing is incomplete because the observed score is jointly produced by the dataset, feature provenance, labels, preprocessing, class prior, train-test split, optimization objective, threshold, and model. A sophisticated model trained on a weak evaluation design can therefore appear more reliable than a simpler model evaluated under deployment-relevant conditions [1-5].

Data-centric AI shifts attention from repeated architecture changes toward systematic improvement and governance of the data used to train and evaluate models [6-9]. This perspective is especially relevant to cybersecurity, where attack families are uneven, traffic distributions evolve, benchmark features can accidentally encode collection artifacts, and random splitting can mix highly similar traffic patterns across training and test sets. Dataset surveys have repeatedly warned that benchmark characteristics strongly shape conclusions about intrusion-detection systems [1-5].

This paper asks whether carefully controlled data interventions can produce larger and more trustworthy gains than incremental model complexity. It evaluates that proposition first on UNSW-NB15 [10,11] and then on CIC-IDS2017 [12,13]. The design deliberately separates model-centric and data-centric changes. Feature-provenance control, class weighting, random undersampling, label-noise stress testing, training-size analysis, uncertainty estimation, attack-category auditing, feature-importance analysis, and temporal holdout validation are examined while keeping the core classifier family fixed wherever possible.

The principal contribution is not a new intrusion-detection algorithm. Instead, the study provides empirical evidence that a favorable random-split benchmark can substantially overstate reliability. On CIC-IDS2017, the same class-aware XGBoost configuration that approaches perfect performance under a random split performs far worse when Friday-afternoon traffic is held out as a later distribution. This result strengthens the central claim that reliable cybersecurity prediction is a property of the data-model-evaluation system rather than of the model alone.

Data-centric reliability framework used in this study

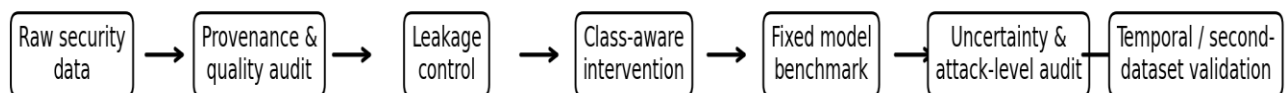


Figure 1. Data-centric reliability framework evaluated in the study.

2. Related Work

2.1 Data-centric AI and dataset governance

Data-centric AI treats data collection, labeling, validation, documentation, maintenance, and

iterative improvement as first-class engineering activities [6-9]. Datasheets for datasets and related documentation practices emphasize provenance, intended use, composition, and known limitations [8]. Production ML literature similarly identifies hidden data dependencies and changing external conditions as major sources of technical debt [9]. These ideas are directly applicable to intrusion detection because traffic datasets are not neutral containers: their capture environment, attack schedule, flow generator, feature definitions, and labeling process constrain what a fitted model can learn.

2.2 Intrusion-detection datasets and benchmark limitations

UNSW-NB15 was created using IXIA PerfectStorm traffic generation and includes normal traffic plus nine attack families [10,11]. CIC-IDS2017 was designed to provide benign traffic together with common attacks recorded across multiple weekdays, with labeled flows generated for machine-learning use [12,13]. Reviews of network intrusion datasets emphasize representativeness, attack diversity, recording environment, labeling, duplication, temporal coverage, and evaluation protocol as critical dataset properties [1-5,14-17].

CIC-IDS2017 has also been the subject of dataset-specific troubleshooting work identifying issues in flow generation and ground-truth handling, reinforcing the need to treat benchmark preprocessing as part of the experimental method rather than as invisible background [14]. Recent systematic reviews continue to identify class imbalance, unrealistic evaluation conditions, weak cross-dataset validation, and inconsistent benchmarking as barriers to deployment-relevant IDS evidence [2-5].

2.3 Imbalance, calibration, and explainability

Security classifiers operate under asymmetric costs. False positives consume analyst attention, while false negatives allow attacks to pass undetected. Accuracy alone is therefore insufficient, particularly under class imbalance [18-21]. Precision-recall analysis, calibrated probabilities, and explicit false-positive reporting provide complementary information [22-25]. Explainability methods such as SHAP and model-native feature importance can also reveal whether a detector relies on plausible traffic characteristics or on dataset artifacts [26-30]. Recent XAI reviews for IDS emphasize that transparent explanations are important for analyst trust but should be evaluated alongside computational cost and generalization [3,29,30].

3. Data and Methodology

3.1 Datasets

UNSW-NB15 was used as the primary benchmark. The official training and testing partitions contain 175,341 and 82,332 records respectively. As established in the original analysis, the supplied partitions contain no missing cells or exact duplicate rows. The target-derived `'attack_cat'` field is excluded from binary prediction, and the `'id'` field is removed in leakage-controlled conditions because it is not a deployable behavioral predictor.

CIC-IDS2017 was used as a second benchmark. All eight machine-learning CSV files supplied in MachineLearningCSV were audited in full. Together they contain 2,830,743 rows across Monday through Friday traffic. Monday contains only benign traffic; later files contain Bot, DDoS, PortScan, brute-force, web, infiltration, DoS, Heartbleed, and other attack families. Exact full-file counts are reported in Table 1. For repeated modeling, a deterministic 103,245-record sample was constructed by sampling across chunks from every file. This sampling decision is explicitly reported so that the computational experiment is reproducible and is not confused with a full-corpus model fit.

Table 1. Exact row and label counts obtained by auditing all eight CIC-IDS2017 CSV files.

CIC-IDS2017 file	Rows	Label counts
Friday-WorkingHours-Afternoon-DDos	225,745	DDoS: 128,027; BENIGN: 97,718
Friday-WorkingHours-Afternoon-PortScan	286,467	BENIGN: 127,537; PortScan: 158,930
Friday-WorkingHours-Morning	191,033	BENIGN: 189,067; Bot: 1,966
Monday-WorkingHours	529,918	BENIGN: 529,918
Thursday-WorkingHours-Afternoon-Infiltration	288,602	BENIGN: 288,566; Infiltration: 36
Thursday-WorkingHours-Morning-WebAttacks	170,366	BENIGN: 168,186; Web Attack - Brute Force: 1,507; Web Attack - XSS: 652; Web Attack - Sql Injection: 21
Tuesday-WorkingHours	445,909	BENIGN: 432,074; FTP-Patator: 7,938; SSH-Patator: 5,897
Wednesday-workingHours	692,703	BENIGN: 440,031; DoS slowloris: 5,796; DoS Slowhttptest: 5,499; DoS Hulk: 231,073; DoS GoldenEye: 10,293; Heartbleed: 11

3.2 Data-centric interventions

The intervention sequence was designed to distinguish justified data improvements from indiscriminate preprocessing. The first stage audits completeness, exact duplication, invalid numeric values, class prevalence, and feature provenance. The second removes target-derived or non-deployable fields. The third changes the learning data or objective through class-sensitive weighting and random undersampling. A 5% label-noise experiment tests sensitivity to imperfect labels rather than presenting label corruption as an improvement. Training-size experiments at 25%, 50%, and 100% of the fixed UNSW analysis

sample evaluate whether additional usable data produces measurable reliability gains.

For CIC-IDS2017, infinities were converted to missing values, features containing no valid numeric information were removed, remaining missing numeric values were median-imputed, and constant features were excluded. Seventy usable numeric features remained in the deterministic modeling sample. The primary comparison uses both stratified random splitting and a later-distribution temporal holdout in which the two Friday-afternoon files are reserved for testing and the other sampled weekday files are used for fitting.

3.3 Models and reproducibility

Table 2. Reproducibility specification for the principal experiments.

Model	Key hyperparameters	Preprocessing	Class handling	Seed	Software
Logistic regression	max_iter=800	One-hot categorical encoding; leakage-controlled fields removed	None unless specified	42	scikit-learn 1.8.0

Decision tree	max_depth=18; min_samples_leaf=2	Same leakage controls	None unless specified	42	scikit-learn 1.8.0
Random forest	180 trees for UNSW primary; class_weight=balanced in weighted condition	Same leakage controls	Balanced weights in intervention	42	scikit-learn 1.8.0
XGBoost (UNSW intervention sample)	140 trees; max_depth=5; learning_rate=0.10; subsample=0.9; colsample_bytree=0.9	One-hot categorical encoding; id and attack_cat removed	scale_pos_weight for class-aware condition	42	xgboost 3.1.3
XGBoost (CIC sample)	130 trees; max_depth=5; learning_rate=0.10; subsample=0.9; colsample_bytree=0.9	Numeric cleaning, median imputation, constant removal	scale_pos_weight for class-aware condition	42	xgboost 3.1.3

3.4 Evaluation and statistical analysis

Performance is reported using accuracy, precision, recall, F1, ROC-AUC, PR-AUC, false-positive rate (FPR), and Brier score. Lower FPR and Brier score are preferable. For the class-aware versus unweighted UNSW XGBoost comparison, 150 paired nonparametric bootstrap resamples of the fixed test sample were used to estimate 95% confidence intervals for differences in F1 and PR-AUC. McNemar's exact test was applied to discordant paired classification outcomes. Because the test sample is large, statistical significance is interpreted together with effect magnitude rather than in isolation [31-33].

4. Results

4.1 UNSW-NB15 data-centric effects

The expanded analysis reproduced the direction of the original full-partition findings. On the fixed 80,000/40,000 stratified analysis sample, unweighted leakage-controlled XGBoost achieved F1 = 0.898, precision = 0.825, recall = 0.985, and FPR = 0.256. Class-aware weighting increased F1 to 0.926 and precision to 0.892 while reducing FPR to 0.143; recall remained high at 0.963. Random undersampling produced a nearly identical F1 of 0.926 and an FPR of 0.140. The label-noise stress test retained high aggregate F1 (0.927) but reduced PR-AUC relative to the clean weighted condition, illustrating that aggregate thresholded metrics can conceal degradation in ranking quality.

Table 3. XGBoost data-centric intervention results on the fixed UNSW-NB15 analysis sample.

UNSW condition	Accuracy	Precision	Recall	F1	ROC-AUC	PR-AUC	FPR	Brier
Baseline	0.877	0.825	0.985	0.898	0.984	0.988	0.256	0.079
Class weighted	0.915	0.892	0.963	0.926	0.985	0.989	0.143	0.058
Random undersampling	0.916	0.894	0.962	0.926	0.985	0.989	0.140	0.058
5% label-noise stress test	0.917	0.894	0.964	0.927	0.983	0.986	0.141	0.063

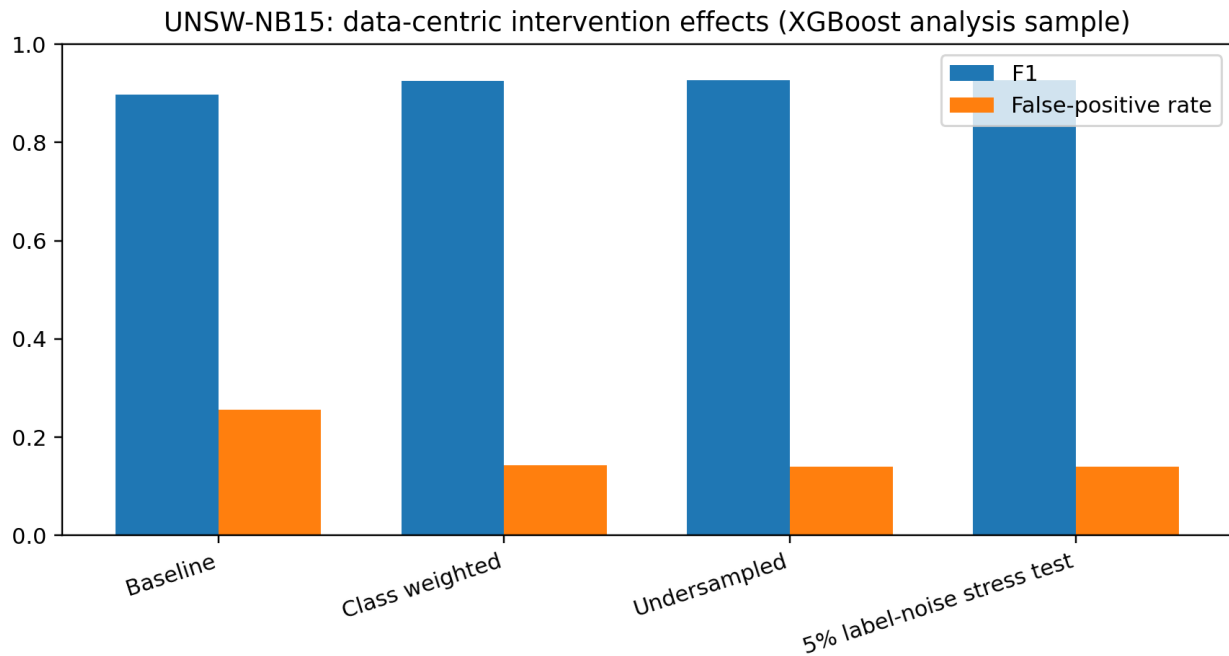


Figure 2. Effect of controlled data interventions on UNSW-NB15 XGBoost performance.

4.2 Statistical uncertainty

The paired bootstrap estimated an F1 improvement of 0.0278 for class-aware weighting versus the unweighted XGBoost baseline, with a 95% confidence interval of 0.0258 to 0.0298. The PR-AUC difference was smaller but consistently positive (0.000635; 95% CI 0.000534 to 0.000734). McNemar's test identified 501

observations correctly classified only by the baseline and 2034 correctly classified only by the weighted model; the exact p-value was < 0.001 . Thus, the improvement is not merely a point-estimate fluctuation, although its operational importance is better represented by the large reduction in false positives than by the very small PR-AUC shift.

4.3 Training-size and attack-category analysis

Table 4. Training-size sensitivity for class-aware XGBoost on UNSW-NB15.

Training fraction	F1	PR-AUC	FPR	Brier
25%	0.915	0.986	0.168	0.065
50%	0.922	0.988	0.155	0.062
100%	0.926	0.989	0.143	0.058

Performance improved as the amount of usable training data increased: F1 rose from 0.915 at 25% of the analysis training sample to 0.922 at 50% and 0.926 at 100%, while FPR declined from 0.168 to 0.155 and then 0.143. This result supports the data-centric premise that additional representative data can yield practical gains without increasing model depth or architectural complexity.

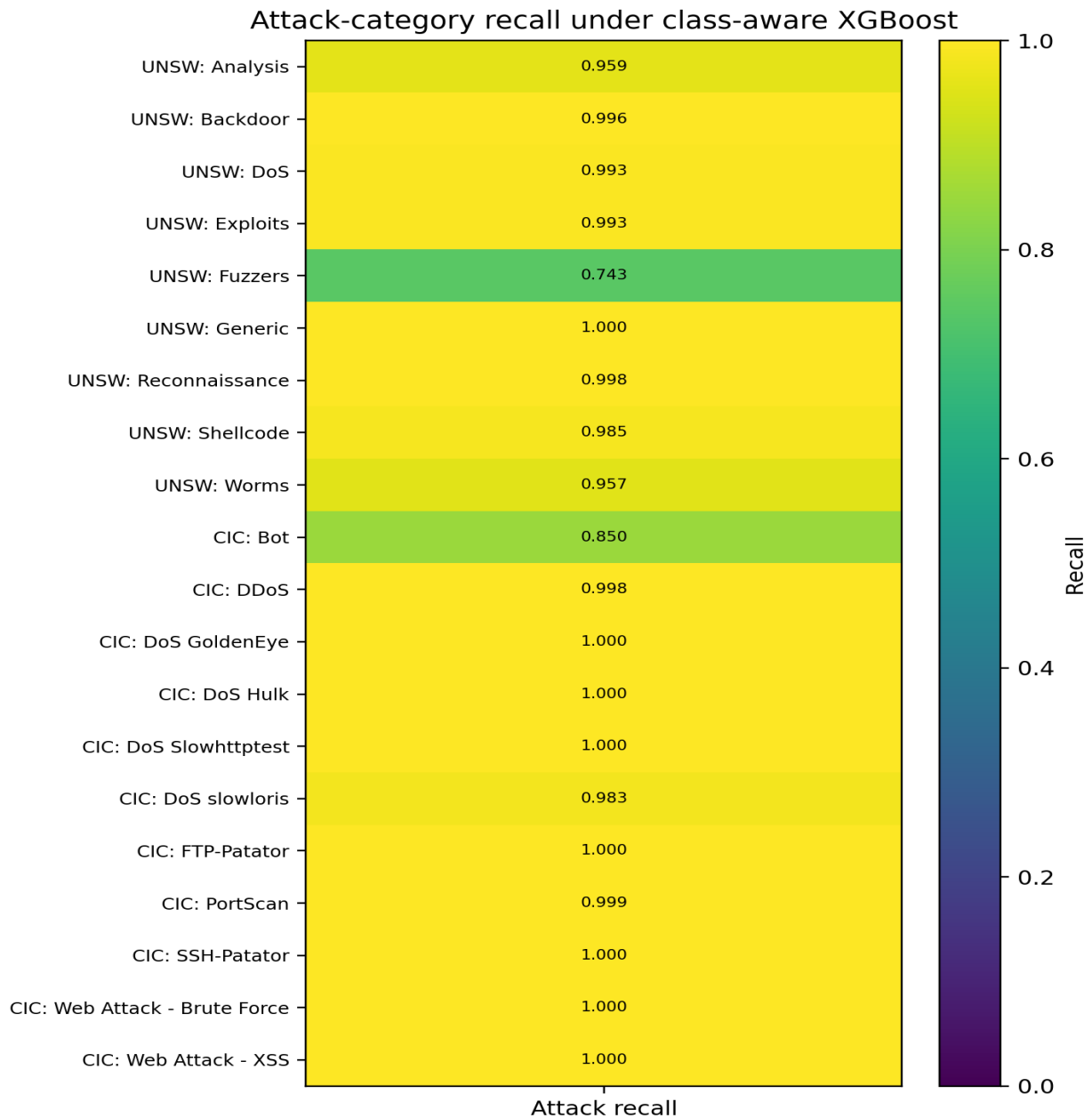


Figure 3. Attack-category recall for the class-aware XGBoost evaluations. Rare categories should be interpreted with their sample sizes.

Attack-level analysis showed why aggregate F1 is insufficient. On UNSW-NB15, weighted XGBoost recall exceeded 0.95 for most attack families but fell to 0.743 for Fuzzers. Worms recall

was 0.957 but was based on only 23 records in the fixed test sample. This heterogeneity would be hidden by the overall binary F1 of 0.926.

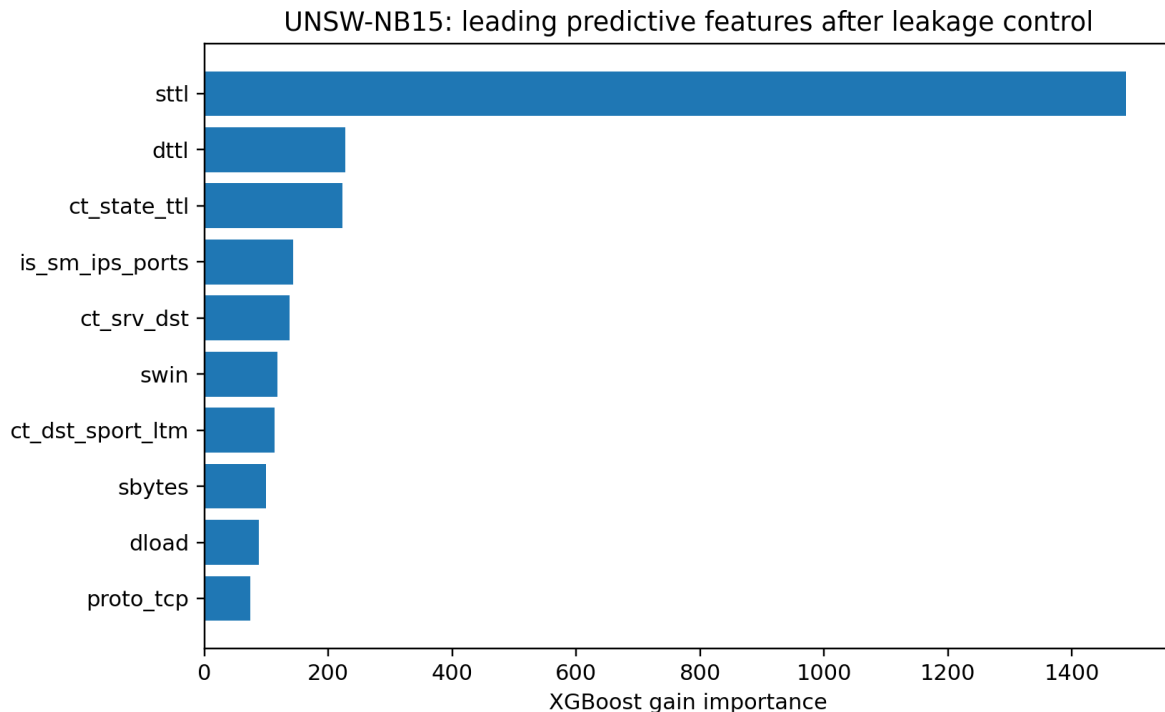


Figure 4. Top XGBoost gain-based features on leakage-controlled UNSW-NB15.

After leakage control, the highest-gain UNSW features included source and destination TTL (sttl and dttl), ct_state_ttl, is_sm_ips_ports, ct_srv_dst, swin, ct_dst_sport_ltm, sbytes, dload, and protocol/service indicators. These variables describe traffic or connection behavior rather than row identity or target metadata. On CIC-IDS2017, the leading features included Subflow Fwd Packets, Bwd Packet Length Min, Total Fwd Packets, Average Packet Size, Bwd Packet Length Std, Subflow Bwd Bytes, PSH Flag Count, and Bwd Header Length. This cross-benchmark difference is itself informative: feature importance is dataset-dependent and should not be interpreted as a universal causal explanation. SHAP remains a useful extension for local explanations, but model-native gain was used here for a low-overhead reproducible global audit [26-30].

4.5 Independent Replication and Temporal Validation on CIC-IDS2017

CIC-IDS2017 provides the strongest independent replication of the paper's central argument. Under a conventional stratified random split of the deterministic modeling sample, unweighted XGBoost achieved $F1 = 0.9970$, $ROC-AUC = 0.99995$, and $FPR = 0.00071$. Class-aware weighting did not improve this already saturated random-split result: $F1$ was 0.9962 and FPR increased slightly to 0.00134 . Interpreted without a distribution-aware holdout, these values could create an overly optimistic impression of deployment reliability.

The interpretation changes under a temporally structured holdout. When both Friday-afternoon files were reserved for testing and the remaining sampled weekday files were used for fitting, class-aware XGBoost accuracy fell to 0.613 , recall to 0.277 , and $F1$ to 0.433 . $ROC-AUC$ remained 0.918 and $PR-AUC$ 0.920 , indicating that the model retained substantial ranking information even though the default operating threshold

transferred poorly to the shifted distribution. The FPR remained low at 0.0041. This random-versus-temporal contrast is much larger than the differences among strong model families on

UNSW-NB15 and shows that evaluation design can dominate apparent gains from model complexity.

Table 5. CIC-IDS2017 second-benchmark replication and temporal validation.

CIC condition	Accuracy	Precision	Recall	F1	ROC-AUC	PR-AUC	FPR	Brier
Random split, baseline	0.999	0.997	0.997	0.997	1.000	1.000	0.001	0.001
Random split, class-aware	0.999	0.994	0.998	0.996	1.000	1.000	0.001	0.001
Friday-afternoon temporal holdout, class-aware	0.613	0.987	0.277	0.433	0.918	0.920	0.004	0.372

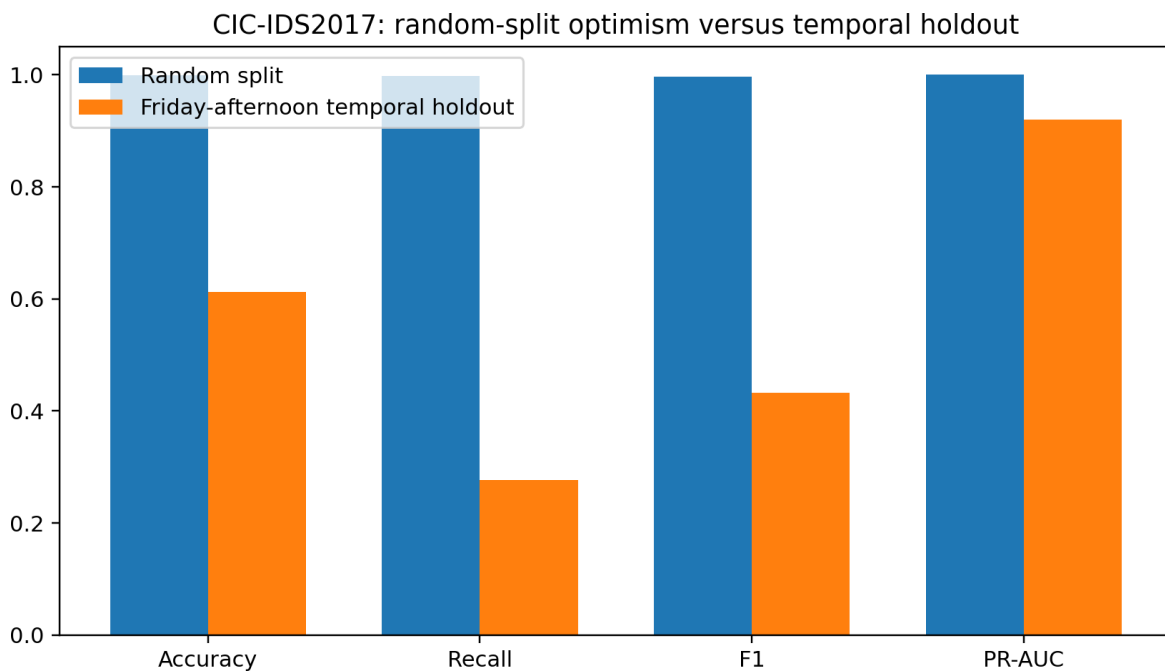


Figure 5. CIC-IDS2017 random-split versus Friday-afternoon temporal-holdout performance.

5. Discussion

The two-benchmark evidence changes the interpretation of model performance in three

important ways. First, justified data interventions can produce larger operational improvements

than switching between already strong ensemble models. On UNSW-NB15, class-aware weighting and undersampling substantially reduce false positives while preserving high recall. The paired bootstrap and McNemar analyses indicate that the change in classification behavior is systematic rather than a chance fluctuation in the evaluated sample.

Second, aggregate scores can hide category-specific weakness. Fuzzers remain substantially harder for the weighted UNSW classifier than Generic, Reconnaissance, or Exploits. Rare classes such as Worms and CIC web attacks also carry wide implicit uncertainty because their sample counts are small. A publication or deployment decision based only on binary F1 would fail to expose these vulnerabilities.

Third, and most importantly, the CIC-IDS2017 random/temporal contrast shows that near-perfect random-split performance is not equivalent to temporal robustness. The model ranks Friday-afternoon observations reasonably well but its thresholded attack recall collapses. This behavior is consistent with dataset-shift concerns discussed in modern IDS reviews and with long-standing warnings that benchmark design can produce optimistic conclusions [1-5,14-17]. The practical implication is that data partitioning should be treated as an experimental intervention with the same importance as choosing a classifier.

The two benchmarks also show that class imbalance interventions are context-dependent. Weighting improves the UNSW operating trade-off but slightly worsens CIC random-split F1 because the latter task is already nearly separable under random mixing. Data-centric AI is therefore not a recipe to apply balancing, cleaning, or feature selection mechanically. It is a disciplined process of auditing, intervening only where justified, and measuring whether the intervention improves reliability under realistic validation.

6. Practical Framework for Reliable AI-Driven Cybersecurity Prediction

A deployment-oriented workflow should follow seven stages: (1) document data provenance and intended prediction time; (2) audit completeness, duplication, invalid values, label construction, and class prevalence; (3) remove features unavailable or illegitimate at inference time; (4) test justified class-aware or sampling interventions; (5) benchmark a small set of models under fixed preprocessing; (6) report uncertainty, calibration, false-positive burden, and attack-family performance; and (7) perform temporal and preferably cross-environment validation. Explainability should be used to identify implausible predictors and support analyst interpretation, not to convert correlation into causal claims.

7. Reproducibility Materials

To support independent replication, the submission package should preserve the exact source-file names, preprocessing rules, deterministic sampling indices, train-test assignments, random seed (42), model hyperparameters, software versions, bootstrap procedure, and code used to generate every reported table and figure. The manuscript already reports the principal model settings and software versions in Table 2; the companion package should additionally include a machine-readable environment specification (for example, requirements.txt or environment.yml), a single entry-point script or notebook that rebuilds the analysis from the raw benchmark files, and a README that maps each output file to the corresponding manuscript table or figure. No unpublished code repository or DOI is claimed in this manuscript; these materials should be archived at submission or acceptance in accordance with the target journal's data/code policy.

Table 6. Reproducibility checklist for the journal submission package.

Item	Reported in manuscript	Submission-package action
Raw data provenance	UNSW-NB15 and all eight CIC-IDS2017 CSV files identified	Preserve original filenames and checksums
Randomness	Seed = 42	Set seed in every executable script

Preprocessing	Leakage control, encoding, CIC cleaning and imputation described	Provide executable preprocessing pipeline
Model specification	Principal hyperparameters and software versions in Table 2	Export complete estimator configuration
Statistical analysis	150 paired bootstrap resamples and exact McNemar test described	Archive bootstrap indices/code
Figures and tables	Metrics and analysis conditions reported	Provide script-to-output mapping in README
Environment	Core package versions reported	Add requirements.txt/environment.yml and Python version

8. Limitations

Several limitations remain. The additional UNSW uncertainty and intervention experiments use a fixed stratified subsample rather than refitting every experiment on all 175,341/82,332 official records; the original full-partition baseline results remain reported in the earlier analysis, while the subsample is used for controlled comparisons. CIC-IDS2017 model fitting likewise uses a deterministic 103,245-record sample, although the eight source files were fully audited for exact row and label counts. This choice should be addressed in a future compute-intensive replication on the full corpus. The CIC temporal holdout changes both time and attack mixture, so the observed degradation cannot be attributed to pure temporal drift alone. Some rare attack families have too few observations for stable family-specific estimates. Gain-based feature importance provides global model insight but does not replace full SHAP-based local explanation or causal analysis. Finally, neither benchmark fully represents contemporary production traffic; newer, independently collected datasets are needed before deployment claims are justified.

9. Conclusion

The results support the argument that cybersecurity prediction reliability cannot be inferred from model complexity or random-split accuracy alone. On UNSW-NB15, class-aware data handling produces a statistically supported F1 gain and a large reduction in false positives without

increasing XGBoost complexity. Training-size analysis shows additional gains from more usable data, while attack-level evaluation exposes weaknesses hidden by aggregate metrics. On CIC-IDS2017, near-perfect random-split performance collapses under a Friday-afternoon temporal holdout, providing direct evidence that evaluation design can dominate apparent algorithmic progress. The practical priority should therefore be to make data provenance, leakage control, class structure, uncertainty, attack-level behavior, and temporal generalization explicit before claiming that a more complex model is a more reliable cybersecurity system.

REFERENCES

- Ring M, Wunderlich S, Scheuring D, Landes D, Hotho A. A survey of network-based intrusion detection data sets. *Computers & Security*. 2019;86:147-167. doi:10.1016/j.cose.2019.06.005.
- Rehman HMRU, Liaquat S, Gul MJ, Jhandir MZ, Gavilanes D, Vergara MM, et al. A systematic literature study of machine learning techniques based intrusion detection: datasets, models, challenges, and future directions. *Journal of Big Data*. 2025;12:264. doi:10.1186/s40537-025-01323-2.

- Mohale VZ, Obagbuwa IC. A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity. *Frontiers in Artificial Intelligence*. 2025;8:1526221. doi:10.3389/frai.2025.1526221.
- Akuthota UC, Bhargava L. The role of machine and deep learning in modern intrusion detection systems: A comprehensive review. *Computers & Electrical Engineering*. 2025;124:110318. doi:10.1016/j.compeleceng.2025.110318.
- Ashraf W, Masoodi FS. A survey of intrusion detection datasets for communication networks. *Discover Computing*. 2026;29:577.
- Jakubik J, Vössing M, Kühl N, Walk J, Satzger G, et al. Data-Centric Artificial Intelligence. *Business & Information Systems Engineering*. 2024;66:507-515. doi:10.1007/s12599-024-00857-8.
- Zha D, Bhat ZP, Lai K-H, Yang F, Jiang Z, Zhong S, Hu X. Data-centric Artificial Intelligence: A Survey. *arXiv:2303.10158*. 2023. doi:10.48550/arXiv.2303.10158.
- Geburu T, Morgenstern J, Vecchione B, Vaughan JW, Wallach H, Daumé III H, Crawford K. Datasheets for Datasets. *Communications of the ACM*. 2021;64(12):86-92.
- Sculley D, Holt G, Golovin D, Davydov E, Phillips T, Ebner D, et al. Hidden Technical Debt in Machine Learning Systems. *Advances in Neural Information Processing Systems*. 2015:2503-2511.
- Moustafa N, Slay J. UNSW-NB15: a comprehensive data set for network intrusion detection systems. 2015 Military Communications and Information Systems Conference. 2015:1-6. doi:10.1109/MilCIS.2015.7348942.
- Moustafa N, Slay J. The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective*. 2016;25(1-3):18-31.
- Sharafaldin I, Habibi Lashkari A, Ghorbani AA. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP)*. 2018:108-116. doi:10.5220/0006639801080116.
- Canadian Institute for Cybersecurity, University of New Brunswick. CIC-IDS2017: Intrusion Detection Evaluation Dataset. Dataset documentation.
- Engelen G, Rimmer V, Joosen W. Troubleshooting an Intrusion Detection Dataset: the CICIDS2017 Case Study. 2021 IEEE Security and Privacy Workshops (SPW). 2021:7-12. doi:10.1109/SPW53761.2021.00009.
- Hindy H, Brosset D, Bayne E, Seem AK, Tachtatzis C, Atkinson R, Bellekens X. A taxonomy of network threats and the effect of current datasets on intrusion detection systems. *IEEE Access*. 2020;8:104650-104675. doi:10.1109/ACCESS.2020.3000179.
- Ferrag MA, Maglaras LA, Moschoyiannis S, Janicke H. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*. 2020;50:102419. doi:10.1016/j.jisa.2019.102419.
- Sommer R, Paxson V. Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. 2010 IEEE Symposium on Security and Privacy. 2010:305-316.
- He H, Garcia EA. Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*. 2009;21(9):1263-1284. doi:10.1109/TKDE.2008.239.
- Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*. 2002;16:321-357.

- Ahsan M, Nygard KE, Gomes R, Chowdhury MM, Rifat N, Connolly JF. Network intrusion detection using machine learning approaches: Addressing data imbalance. *IET Cyber-Physical Systems: Theory & Applications*. 2022;7(1):30-39. doi:10.1049/cps2.12013.
- Shanmugam V, Razavi-Far R, Hallaji E. Addressing Class Imbalance in Intrusion Detection: A Comprehensive Evaluation of Machine Learning Approaches. *Electronics*. 2025;14(1):69. doi:10.3390/electronics14010069.
- Davis J, Goadrich M. The Relationship Between Precision-Recall and ROC Curves. *Proceedings of ICML*. 2006:233-240.
- Fawcett T. An introduction to ROC analysis. *Pattern Recognition Letters*. 2006;27(8):861-874.
- Brier GW. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*. 1950;78(1):1-3.
- Guo C, Pleiss G, Sun Y, Weinberger KQ. On Calibration of Modern Neural Networks. *Proceedings of ICML*. 2017:1321-1330.
- Lundberg SM, Lee S-I. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*. 2017;30.
- Lundberg SM, Erion G, Chen H, et al. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*. 2020;2:56-67.
- Molnar C. *Interpretable Machine Learning*. 2nd ed. 2022.
- Al S, Sagirolu S. Explainable artificial intelligence models in intrusion detection systems. *Engineering Applications of Artificial Intelligence*. 2025;144:110145. doi:10.1016/j.engappai.2025.110145.
- Khediri A, Slimi H, Djeddar A, Bendjenna H, Yahiaoui A, Derdour M. Toward Trustworthy Explainable Intrusion Detection Systems (X-IDS): A Comprehensive Survey of Methods, Interpretability Enhancements, and Future Opportunities. 2025 *International Conference on Recent Advances in Mathematics and Informatics (ICRAMI)*. 2025:1-4. doi:10.1109/ICRAMI64946.2025.11472758.
- Efron B, Tibshirani RJ. *An Introduction to the Bootstrap*. Chapman & Hall/CRC; 1993.
- McNemar Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*. 1947;12:153-157.
- Dietterich TG. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*. 1998;10(7):1895-1923.
- Breiman L. Random Forests. *Machine Learning*. 2001;45:5-32. doi:10.1023/A:1010933404324.
- Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016:785-794. doi:10.1145/2939672.2939785.
- Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 2011;12:2825-2830.
- Barredo Arrieta A, Díaz-Rodríguez N, Del Ser J, et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*. 2020;58:82-115.
- Kohavi R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Proceedings of IJCAI*. 1995:1137-1145.

Wilkinson MD, Dumontier M, Aalbersberg IJJ, et al. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*. 2016;3:160018.