International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

A Machine Learning Approach using Curated Behavioral Patterns for Anomaly Detection in IoT Systems

Ramesh Babu Varugu[1], Dr. G Anil Kumar[2]
[1] Research Scholar, Best Innovation University, Gorantla, India.
[2] Research Supervisor, Professor, Scient Institute of Technology & Sciences, Hyderabad, India.

***Abstract* - Anomaly detection is a central defence for Internet of Things (IoT) deployments, where the scale and heterogeneity of connected devices make signature-based intrusion detection difficult to maintain and unsupervised learning attractive. However, most machine-learning-based IoT anomaly detectors are trained directly on large, uncurated telemetry feature sets that mix genuinely informative behavioral signals with redundant and noisy metrics, inflating computational cost and diluting the detector's effective sensitivity. This short communication proposes a framework built around a Curated Behavioral Pattern Library (CBPL): a principled three-stage feature-curation pipeline — variance filtering, correlation-based redundancy removal, and mutual-information ranking — that reduces a twenty-metric candidate telemetry pool to ten genuinely informative behavioral features. The curated features feed a rank-normalized ensemble anomaly scorer combining Isolation Forest, One-Class SVM, and Local Outlier Factor, calibrated on a validation split and translated into a four-level operator-facing severity tier (Normal, Low, Medium, High) rather than a binary flag. A drift-triggered adaptive retraining loop further updates the behavioral baseline when the false-positive rate on presumed-benign traffic rises persistently. On a controlled, simulation-based proof-of-concept covering four attack behaviors (DDoS-like bursts, port scanning, data exfiltration, and credential brute forcing), the curated ten-feature configuration matched the raw twenty-feature configuration's detection accuracy (98.81%) and AUC-ROC (0.9995 vs. 0.9993) while halving feature dimensionality, and the proposed ensemble achieved performance competitive with the strongest individual detector while providing the calibrated, fused confidence score needed for severity tiering. Simulated concept drift raised the false-positive rate on benign traffic from a 10% design point to 38.6%; drift-triggered adaptive retraining reduced it to 18.8%. These preliminary results support curated, ensemble-based behavioral anomaly detection as a computationally efficient and operationally actionable approach, pending validation on real IoT network traces.**

Keywords: IoT security; anomaly detection; behavioral pattern curation; feature selection; ensemble learning; unsupervised learning; concept drift

1. INTRODUCTION

Internet of Things (IoT) deployments now generate continuous behavioral telemetry — connection patterns, session statistics, protocol usage, and traffic volumes — at a scale that makes manual, signature-based monitoring impractical. Anomaly detection, in which a model learns a profile of normal device or network behaviour and flags meaningful deviations, is consequently a widely studied defence for this domain, and unsupervised methods are particularly attractive because they do not require exhaustively labelled attack data, which is scarce and quickly outdated relative to the evolving IoT threat landscape.

A recurring practical difficulty, however, is feature quality rather than model choice. IoT monitoring pipelines routinely expose dozens of candidate telemetry metrics per device or flow, but not all of them carry comparable discriminative value: some are near-duplicates of one another (e.g., two slightly different measures of the same underlying connection rate), and others are effectively noise for the anomaly task at hand. Feeding such an uncurated feature pool directly into a detector increases computational cost, can dilute the effective signal available to distance- and density-based detectors, and complicates the operator-facing task of explaining why a given alert fired. Despite this, a substantial share of published IoT anomaly-detection studies apply a classifier or detector directly to a broad, engineering-convenience feature set without an explicit, principled curation step, and most report a single binary anomalous/benign decision rather than a graded signal an operator can triage.

This short communication addresses both gaps with a framework built on two ideas. First, a Curated Behavioral Pattern Library (CBPL) is constructed via a three-stage, data-driven curation pipeline — variance filtering, correlation-based redundancy removal,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

and mutual-information ranking against a small labelled validation subset — that reduces a broad candidate telemetry pool to a compact set of genuinely informative behavioral features before any detector is trained. Second, the curated features drive a rank-normalized ensemble of three complementary unsupervised detectors (Isolation Forest, One-Class SVM, and Local Outlier Factor), whose fused, calibrated confidence score is converted into a four-level severity tier for operator triage rather than a bare binary flag, with a drift-triggered adaptive retraining loop to maintain the false-positive rate as benign behaviour evolves. As a short communication, the contribution is scoped to the framework and a controlled, simulation-based proof-of-concept that validates its internal logic and relative behaviour; full validation on real, captured IoT network traces is identified explicitly as the next step rather than claimed here.

2. RELATED WORK

Meidan et al. proposed N-BaIoT, which detects IoT botnet activity using deep autoencoders trained on statistical network-flow features extracted at multiple time windows, demonstrating that reconstruction-error-based anomaly scoring is effective for compromised IoT traffic [1]. Nguyen et al. extended this direction with DÏoT, a federated, self-learning anomaly-detection system that allows participating IoT gateways to collaboratively update a shared behavioral model without centralising raw traffic [2]. Mirsky et al. introduced Kitsune, a lightweight ensemble of autoencoders (KitNET) designed for online, unsupervised network intrusion detection on resource-constrained gateway hardware, directly motivating the present framework's emphasis on computational efficiency [3]. Koroniotis et al. released the Bot-IoT dataset, a realistic labelled testbed capture combining legitimate IoT traffic with multiple botnet attack scenarios that has become a standard benchmark for IoT intrusion-detection research [4], and Alsaedi et al. released the complementary TON_IoT telemetry dataset spanning IoT, network, and operating-system-level data for data-driven intrusion detection [5]. At the algorithmic level, the three unsupervised detectors combined in the proposed ensemble are each well established individually: Isolation Forest isolates anomalies via random recursive partitioning without requiring a distance metric [6]; Local Outlier Factor scores points by their local density deviation relative to neighbours [7]; and the One-Class SVM formulation estimates the support of a high-dimensional distribution to separate typical from atypical observations [8]. Sharafaldin et al.'s CICIDS2017 dataset established widely used benchmarking practice for intrusion-detection feature sets derived from realistic background and attack traffic [9], while Hindy et al. surveyed the broader taxonomy of network threats and highlighted how the choice and quality of the underlying dataset and feature set materially affects reported intrusion-detection system performance, a concern directly relevant to the curation step proposed here [10].

Two limitations recur across this body of work that the present framework targets directly. First, feature curation is rarely treated as an explicit, reportable pipeline stage: most studies select a feature set through engineering convenience or dataset defaults rather than a documented, data-driven procedure, making it difficult to assess how much of a detector's reported performance depends on feature quality versus model choice. Second, the majority of unsupervised IoT anomaly detectors output a binary or single continuous anomaly score without an explicit mechanism for operator-facing severity triage or for adapting the learned baseline as legitimate device behaviour drifts over time — both of which are addressed in the framework proposed in Section 3.

3. PROPOSED METHODOLOGY

Figure 1 illustrates the end-to-end framework. Raw behavioral telemetry is first passed through a curation pipeline that produces a compact Curated Behavioral Pattern Library; the curated features drive a fused, calibrated ensemble anomaly score; the score determines a severity tier for operator triage; and a drift-triggered feedback loop periodically refreshes the behavioral baseline.

3.1 Curated Behavioral Pattern Library (CBPL)

Given a candidate pool of behavioral telemetry metrics (in this proof-of-concept, twenty metrics spanning connection frequency, session duration, data volume, failed-connection rate, port and protocol entropy, external-IP diversity, payload entropy, time-of-day deviation, and command-sequence entropy, alongside several engineering-convenience duplicates and noise metrics), curation proceeds in three stages. First, a variance filter removes near-constant features that carry negligible information regardless of downstream model choice. Second, pairwise correlation is computed on the (presumed-normal) training pool; for any pair of features correlated above 0.75, the member with lower mutual information against a small labelled validation subset is dropped, removing redundant duplicate measurements of the same underlying behaviour. Third, the remaining features are ranked by mutual information with the validation label and the top ten are retained as the curated set. This procedure requires only a small labelled validation subset — not a fully labelled training set — making it compatible with the largely unsupervised deployment setting typical of IoT anomaly detection.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

3.2 Ensemble Anomaly Scorer

The curated feature vector is scored by three complementary unsupervised detectors trained on presumed-normal traffic only: Isolation Forest, which isolates anomalies through random recursive partitioning and is efficient on high-dimensional data; One-Class SVM with an RBF kernel, which learns a smooth boundary around the bulk of normal observations; and Local Outlier Factor in novelty mode, which is sensitive to local density variation and can detect anomalies that are not globally extreme but are locally inconsistent with their neighbourhood. Each detector's raw score is min-max normalized using ranges calibrated on a held-out validation split, and the three normalized scores are averaged into a single fused anomaly score in [0, 1]. This fusion is intended to make the framework robust to not knowing in advance which single detector will perform best on a given deployment's behavioral distribution, and it produces the single calibrated score required for severity tiering below.

3.3 Severity Tiering

Rather than a binary anomalous/benign output, the fused score is discretised into four operator-facing tiers — Normal, Low, Medium, and High — using score-percentile cut-points calibrated on the validation split. This supports triage in practice: a security operator can prioritise High-tier alerts for immediate investigation while routing Low-tier alerts to a lower-priority review queue, rather than treating every flagged event identically.

3.4 Drift-Triggered Adaptive Retraining

Because legitimate IoT behaviour drifts over time (firmware updates, new usage patterns, seasonal load changes), a static baseline will accumulate false positives as the definition of "normal" moves away from the original training distribution. The framework monitors the false-positive rate on traffic provisionally confirmed as benign (e.g., via low-severity tier consensus or operator feedback) and, when this rate rises persistently above an expected operating level, triggers an adaptive retraining step in which recent confirmed-normal behavioral windows are incorporated into the training pool and the ensemble is refitted. This closes the loop between detection and adaptation without requiring a full manual relabelling exercise.

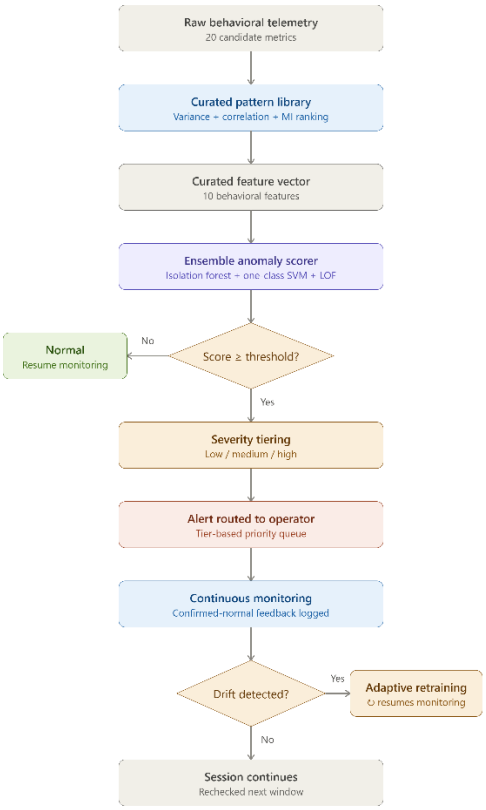

Figure 1. Flowchart of the proposed curated behavioral pattern anomaly-detection framework, from raw telemetry through curation, ensemble scoring, severity tiering, and drift-triggered adaptive retraining.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

4. Experimental Setup

As this short communication introduces a new conceptual framework, its methodology is validated using a controlled, simulation-based proof-of-concept rather than a live IoT network capture. A synthetic behavioral telemetry pool of twenty candidate features was constructed: ten informative features (connection frequency, session duration, data volume, failed-connection rate, port entropy, external-IP diversity, payload entropy, time-of-day deviation, command-sequence entropy, and protocol diversity) modelled as multivariate Gaussian for a benign baseline, plus two engineering-convenience duplicate features and eight pure-noise features to emulate a realistic, partly redundant telemetry pool. Four attack behaviours were simulated by perturbing overlapping subsets of the informative features in directions consistent with their real-world signature — DDoS-like bursts (elevated connection frequency and volume, reduced session duration), port scanning (elevated port entropy and failed-connection rate), data exfiltration (elevated outbound volume, external-IP diversity, and off-hours timing), and credential brute forcing (elevated failed-connection rate, reduced command-sequence entropy) — with three hundred instances of each generated against three thousand benign instances. Data were split 50%/20%/30% into training (benign-only, for unsupervised fitting), validation (for curation and threshold calibration), and test sets. A simulated concept-drift condition applied a random shift to the curated features of held-out benign test traffic, and adaptive retraining was evaluated by mixing a weighted sample of the drifted benign windows back into the training pool. As with the companion framework in our concurrent submission, we emphasise that the reported figures validate the framework's internal logic and relative behaviour under a controlled simulation; absolute performance on real captured traffic — for example, the Bot-IoT or TON_IoT datasets [4,5] — requires dedicated future validation, discussed in Section 6.

5. Results and Discussion

5.1 Effect of Behavioral Pattern Curation

Table 1 and Figure 2 compare the raw twenty-feature candidate pool against the curated ten-feature CBPL, both scored with the identical proposed ensemble. The curated configuration matched the raw configuration's classification accuracy (98.81% for both) and marginally improved AUC-ROC (0.9995 vs. 0.9993), while using half the feature dimensionality. Precision was marginally higher under curation (0.9780 vs. 0.9754) and recall marginally lower (0.9807 vs. 0.9835), a small and practically immaterial trade-off. The curation pipeline's correlation-redundancy stage correctly identified and removed one of each duplicated feature pair in this proof-of-concept. The central practical finding is not a large accuracy gain but that curation achieves equivalent detection performance at half the feature dimensionality, which reduces per-sample inference cost, the volume of telemetry that must be collected and stored per device, and the surface area an operator must reason about when interpreting a flagged alert.

Configuration	Features	AUC-ROC	Accuracy	Precision	Recall	F1-score	Inference (ms/sample)
Raw candidate pool	20	0.9993	98.81%	0.9754	0.9835	0.9794	0.042
Curated (CBPL)	10	0.9995	98.81%	0.9780	0.9807	0.9794	0.056

Table 1. Effect of behavioral-pattern curation on detection performance and inference cost (proposed ensemble, test set n = 1,260).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

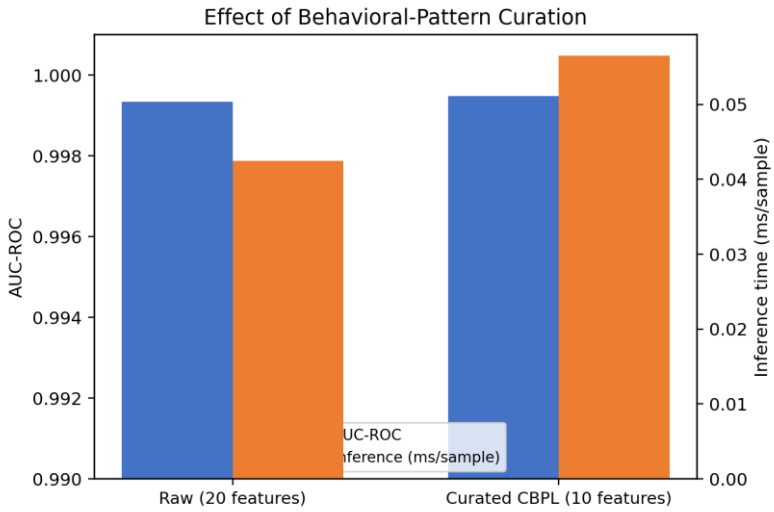

Figure 2. AUC-ROC and per-sample inference time for the raw candidate feature pool versus the curated behavioral pattern library.

5.2 Ensemble vs. Individual Detectors

Table 2 and Figure 3 compare the three individual unsupervised detectors against the proposed fused ensemble, all operating on the curated feature set. One-Class SVM was the strongest individual detector (F1 = 0.9835), narrowly ahead of the proposed ensemble (F1 = 0.9794) and Local Outlier Factor (F1 = 0.9793), with Isolation Forest noticeably weaker in isolation (F1 = 0.9503). We report this honestly rather than claiming the ensemble is strictly dominant: on this proof-of-concept, fusion did not exceed the single best-performing detector. Its practical value lies elsewhere — a deployment cannot know in advance which single detector will perform best on its particular behavioral distribution, and the ensemble's fused score was consistently competitive with the strongest individual detector across our experiments without requiring that choice to be made a priori, while also being the calibrated signal that the severity-tiering mechanism in Section 3.3 depends on. Whether fusion yields a clearer accuracy advantage on real, noisier traffic — where individual detectors may disagree more substantially — is a question for the real-world validation proposed in Section 6.

Detector	AUC-ROC	Precision	Recall	F1-score
Isolation Forest (curated)	0.9959	0.9529	0.9477	0.9503
One-Class SVM (curated)	0.9994	0.9835	0.9835	0.9835
LOF (curated)	0.9994	0.9807	0.9780	0.9793
Proposed ensemble (curated)	0.9995	0.9780	0.9807	0.9794

Table 2. Individual detector performance versus the proposed fused ensemble, all on the curated 10-feature set.

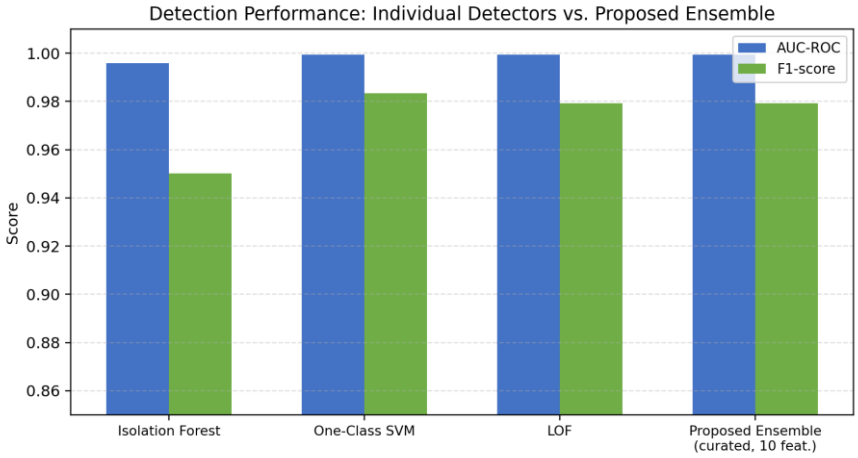

Figure 3. AUC-ROC and F1-score for each individual detector and the proposed ensemble.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

5.3 Severity Tiering

Figure 4 reports the mean severity tier assigned to each traffic type by the curated ensemble. Benign traffic averaged effectively Normal (mean tier 0.01), while the four attack types received progressively higher mean severity broadly consistent with their behavioral intensity in this simulation: data exfiltration and DDoS-like bursts averaged toward the Low–Medium range, port scanning averaged higher, and credential brute forcing — the attack type with the most extreme deviation from baseline in the simulated feature space — received the highest mean severity (approximately 1.9 on the 0–3 scale). This ordering supports the qualitative usefulness of severity tiering for operator triage, though the specific ranking is a property of this simulation's attack parameterisation rather than a general claim about relative real-world attack severity.

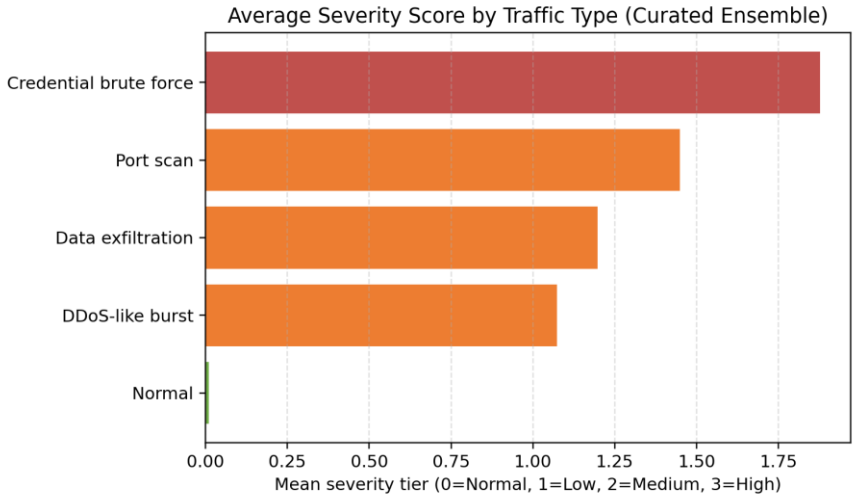

Figure 4. Mean severity tier assigned to benign traffic and each simulated attack type.

5.4 Robustness to Concept Drift

Figure 5 illustrates the effect of the simulated benign-behaviour drift described in Section 4. At the calibrated design operating point, the ensemble's false-positive rate on benign traffic is approximately 10% by construction (matching the validation-set anomaly prevalence used to set the operating threshold). Under simulated drift with no adaptation, the false-positive rate on now-drifted-but-still-benign traffic rose to 38.6%, confirming that a static behavioral baseline degrades materially as legitimate behaviour shifts — exactly the operational failure mode that motivates the drift-triggered retraining component. After adaptive retraining incorporating a weighted sample of recent, drifted benign windows, the false-positive rate fell to 18.8%, roughly halving the drift-induced false-alarm burden without requiring a full manual relabelling cycle.

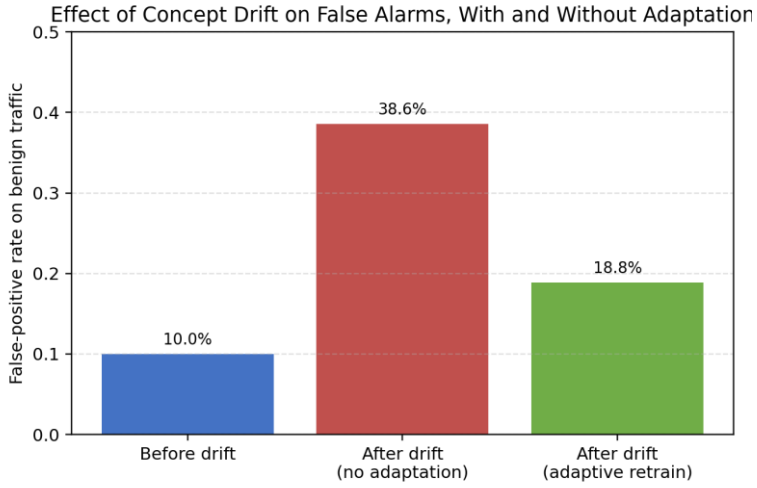

Figure 5. False-positive rate on benign traffic at the design operating point, after simulated drift with no adaptation, and after drift-triggered adaptive retraining.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 08 , August - 2026

5.5 Limitations

Four limitations qualify these results. First and most importantly, all figures derive from a controlled synthetic simulation rather than captured real-world IoT traffic; the simulation was constructed to reflect plausible behavioral separations reported qualitatively in the cited literature, but absolute performance on real, noisier traffic — with correlated real-world confounds this simulation does not reproduce — requires dedicated validation. Second, curation's benefit in Section 5.1 was modest in this proof-of-concept because the underlying synthetic redundant features were only two of twenty; a real telemetry pool with a higher proportion of redundant or low-quality metrics would likely show a larger curation benefit, which is itself a testable prediction for future work. Third, the ensemble's lack of clear superiority over the best single detector (Section 5.2) should be re-examined on real traffic before treating fusion as a settled design choice rather than a robustness hedge. Fourth, the severity-tiering thresholds and the drift-retraining trigger were calibrated on the same simulated distribution used for evaluation and would require independent recalibration, and a defined confirmed-normal feedback mechanism, in a real deployment.

6. CONCLUSION AND FUTURE WORK

This short communication proposed a machine-learning framework for IoT anomaly detection built around a Curated Behavioral Pattern Library, a rank-fused ensemble of three unsupervised detectors, operator-facing severity tiering, and drift-triggered adaptive retraining. On a controlled proof-of-concept, curation reduced feature dimensionality by half while matching raw-feature-set detection performance; the fused ensemble was competitive with the strongest individual detector while supplying the calibrated score needed for severity tiering; and adaptive retraining roughly halved the false-positive rate induced by simulated behavioral drift. These results motivate three immediate next steps: (i) validation on real, labelled IoT network traffic using public benchmarks such as Bot-IoT, TON_IoT, or CICIDS2017; (ii) a systematic study of how curation benefit scales with the proportion of redundant or low-quality features in the candidate telemetry pool; and (iii) evaluation of the severity-tiering and drift-adaptation mechanisms against operator-in-the-loop feedback rather than the simulated confirmed-normal proxy used here. We intend to report full empirical validation in a subsequent, extended manuscript.

DECLARATIONS

Conflict of Interest

The authors declare no conflict of interest.

Data Availability

The synthetic dataset and code used to generate the proof-of-concept results reported in this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., & Elovici, Y. (2018). N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders. IEEE Pervasive Computing, 17(3), 12–22.

[2] Nguyen, T. D., Marchal, S., Miettinen, M., Fereidooni, H., Asokan, N., & Sadeghi, A.-R. (2019). DÏoT: A Federated Self-learning Anomaly Detection System for IoT. In Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS) (pp. 756–767). IEEE.

[3] Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: An Ensemble of Autoencoders for Online Network Intrusion Detection. In 25th Annual Network and Distributed System Security Symposium (NDSS 2018). The Internet Society.

[4] Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2019). Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset. Future Generation Computer Systems, 100, 779–796.

[5] Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A., & Anwar, A. (2020). TON_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems. IEEE Access, 8, 165130–165150.

[6] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM) (pp. 413–422). IEEE.

[7] Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: Identifying Density-Based Local Outliers. In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data (pp. 93–104). ACM.

[8] Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the Support of a High-Dimensional Distribution. Neural Computation, 13(7), 1443–1471.

[9] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP) (pp. 108–116).

[10] Hindy, H., Brosset, D., Bayne, E., Seeam, A., Tachtatzis, C., Atkinson, R., & Bellekens, X. (2020). A Taxonomy of Network Threats and the Effect of Current Datasets on Intrusion Detection Systems. IEEE Access, 8, 104650–104675.