Computer Life
ISSN: 1819-4818 | Vol. 14, No. 3, 2026

Explainable Robust Cloud Intrusion Detection Against Adversarial Evasion Using SHAP-uided Feature Reweighting

Hugo Laurent [1], Paula Silva [2, *]

[1] University of São Paulo, Institute of Mathematics and Statistics, Brazil;
[2] University of Lisbon, Instituto Superior Técnico, Department of Computer Science and Engineering, Portugal

[*] **Corresponding author**: paula.silva473@tecnico.ulisboa.pt

Abstract: Cloud intrusion detection systems increasingly rely on machine learning classifiers trained on flow-level telemetry. Such classifiers can be accurate on clean traffic but fragile under adversarial evasion, because attackers may adjust mutable statistics such as timing, packet counts, or byte rates while preserving malicious functionality. This paper proposes SHAP-RW-AT, an explainable robust intrusion detection framework that combines adversarial augmentation with SHAP-guided feature reweighting. A base detector is first explained with SHAP values to identify decision-dominant features. These explanations are then combined with a domain mutability prior: highly important but easily manipulated flow statistics are attenuated, whereas stable high-importance indicators such as destination-port dispersion, failed-login behavior, and cloud API activity are reinforced. The reweighted representation is trained with diversified evasion examples generated by mean-shift, surrogate-gradient, and SHAP-aware perturbations under semantic constraints. Because public cloud-intrusion datasets could not be redistributed or downloaded inside the execution environment, we evaluate the method on a fully reproducible synthetic cloud-flow benchmark generated by the accompanying code; no result is claimed as a measurement on CICIDS-2017, UNSW-NB15, or CVD. Across three random seeds, SHAP-RW-AT obtains 99.67% clean accuracy and 99.71% average robust accuracy under three evasion strategies, reducing adversarial false negatives from 26.47% for a standard XGBoost detector to 0.18%. The results indicate that explanation-guided feature reweighting can act as a lightweight and auditable robustness mechanism for cloud IDS pipelines, while preserving traceability of the defensive decision logic.

Keywords: Cloud intrusion detection; Adversarial evasion; SHAP; Explainable AI; Feature reweighting; Robust machine learning; XGBoost, cybersecurity.

1. Introduction

Cloud platforms concentrate enterprise workloads, identity services, container orchestration, and application programming interfaces in highly elastic environments. This concentration improves scalability but also creates a dense attack surface in which denial-of-service attempts, credential attacks, scanning activity, and web/API abuse can appear as large volumes of flow-level telemetry. Modern intrusion detection systems (IDSs) therefore increasingly use machine learning models to classify network flows from statistical features rather than relying only on static signatures. The practical attraction is clear: a model trained on flow duration, packet counts, inter-arrival time, flag statistics, destination-port behavior, and API activity can generalize across attack variants that do not exactly match an existing rule.

The same statistical dependence also creates an evasion risk. When a learned detector relies heavily on mutable flow statistics, an attacker may modify timing, packet padding, request pacing, or burst structure without changing the malicious objective. In that case, a detector with excellent clean accuracy can silently fail under adversarially shifted inputs. The problem is especially important for cloud IDSs because flow-level features are often the only practical telemetry available at scale when payloads are encrypted, tenants are isolated, and inspection budgets are constrained. A high-performing IDS should therefore be evaluated not only on clean accuracy but also on robust accuracy, false negative rate, false positive rate, and the interpretability of its decision features.

Recent work provides an important foundation for robust cloud intrusion detection by formulating a game-theoretic cost-sensitive adversarial training framework against GAN-based evasion attacks [1]. This work demonstrates that adversarial robustness and clean detection quality need not be treated as mutually exclusive goals when the training objective is tailored to security costs. This paper follows the same positive research direction but explores a lighter and more directly explainable mechanism: instead of placing the main innovation in a generative threat model or game-theoretic loss, we use SHAP explanations to reshape the feature representation itself before robust training.

The central idea is simple. Not every feature used by an IDS has the same security meaning. Some features are operationally mutable: an attacker may slow a flow, pad byte counts, alter inter-arrival time, or spread packets across a longer window. Other features are more stable because changing them would alter the attack goal or infrastructure behavior, such as destination-port dispersion during scanning, failed-login counts during credential attacks, or cloud API call bursts during account abuse. If a base model's SHAP explanation shows that it relies heavily on mutable features, the resulting decision boundary is likely to be easier to evade. We therefore compute global SHAP importance values, combine them with a mutability prior, and learn a feature multiplier that suppresses high-importance mutable dimensions while reinforcing high-importance stable dimensions.

This paper makes four contributions. First, it proposes SHAP-guided feature reweighting for adversarially robust cloud IDSs, providing a compact explanation-driven bridge between local interpretability and robustness engineering. Second, it introduces a diversified adversarial augmentation procedure for tabular cloud-flow records using mean-shift, surrogate-gradient, and SHAP-aware evasion strategies with semantic projection. Third, it provides a fully reproducible experimental package containing code, generated data procedure, metrics, and figures, while explicitly avoiding unsupported claims about inaccessible public datasets. Fourth, it reports a focused empirical study showing that SHAP-RW-AT substantially reduces adversarial false negatives on a controlled cloud-flow benchmark with only a small clean-accuracy cost.

2. Related Work

Machine learning for intrusion detection has been studied for decades, from classical classifiers and feature engineering to deep neural networks and graph-based methods [2], [3]. Public datasets such as NSL-KDD, UNSW-NB15, and CICIDS-2017 have played a major role in benchmarking IDS models, although researchers have repeatedly noted that dataset age, traffic realism, class imbalance, and feature leakage can materially affect reported performance [4]-[7]. For cloud deployments, these concerns are amplified because tenant behavior, encrypted payloads, autoscaling, and API-driven workloads may differ from conventional enterprise traffic.

Adversarial machine learning adds another layer of risk. Foundational work showed that small, carefully constructed perturbations can cause high-confidence misclassification [8]-[12]. Robust optimization and adversarial training subsequently became standard defenses, especially after PGD-based training showed that robustness can be formulated as an inner maximization over perturbations and an outer minimization over model loss [13]. TRADES further clarified the trade-off between clean accuracy and robust accuracy [14], while ensemble adversarial training highlighted the need to diversify training-time attacks rather than defend only against a single perturbation pattern [15].

Adversarial evasion in IDS is more constrained than image perturbation because network-flow features are mixed-type, coupled, and semantically meaningful. IDSGAN demonstrated that GAN-generated malicious records can evade IDS models under a black-box setting [16], and later work on practical adversarial network flows emphasized domain constraints and realistic feature modification [17]. [1] advanced this line of research by combining a conditioned WGAN-GP threat generator with a game-theoretic cost-sensitive loss for robust cloud IDS training. The present work complements that approach: it keeps the focus on adversarially robust cloud detection but emphasizes explainability and feature-level defensibility rather than generator-detector co-evolution.

Explainable AI is particularly relevant in security operations because an alerting model should support analyst trust and post-hoc auditing. SHAP provides a theoretically grounded additive explanation method based on Shapley values [21], [28]. TreeSHAP extends this idea to efficient explanation of tree ensembles and makes it practical to interpret XGBoost-style models at scale [22], [24]. Prior interpretability work often uses SHAP after training to explain decisions; by contrast, this paper uses SHAP during model design to build a reweighted input representation that explicitly reduces reliance on fragile decision features.

Cost-sensitive and imbalanced learning also influence IDS design because false negatives are usually more costly than false positives [26], [27]. However, static class weighting alone does not indicate which feature dependencies are robust. The proposed SHAP-RW-AT framework is therefore positioned between three research streams: adversarial training, explainable feature attribution, and security-aware feature mutability. Its novelty lies in transforming explanations into an operational reweighting rule that can be inspected and reproduced.

3. Threat Model and Problem Formulation

We consider a binary cloud IDS that maps a flow-level feature vector x in R^d to a label y in {0,1}, where y=0 denotes benign traffic and y=1 denotes malicious traffic. The feature vector contains statistical flow features, connection indicators, and cloud-behavior features. The attacker is assumed to know the feature family and to be able to modify only a subset of mutable features M, such as flow duration, packet counts, byte counts, packet rate, byte rate, inter-arrival statistics, TCP-window statistics, and header or payload-size hints. Stable or high-cost features S, such as failed-login count, destination-port dispersion, service category, request entropy, and cloud API call rate, are assumed to be difficult to change without weakening or changing the attack.

For an attack sample x with y=1, an evasion attack generates x_adv = Pi(x + delta), where delta_j = 0 for j not in M, ||delta|| is bounded by a feature-wise perturbation budget, and Pi(.) projects the sample back into a valid flow-feature range. The attacker succeeds when the IDS predicts y_hat=0 for x_adv. The defender seeks a classifier f that minimizes clean error while also minimizing false negatives under multiple evasion strategies. This differs from unconstrained adversarial perturbation because the attack must respect basic flow semantics: binary features remain binary, nonnegative traffic quantities remain nonnegative, and modified values are clipped to empirical ranges observed in the training set.

The main design question is not simply how to add adversarial samples, but how to reduce the model's dependence on features that are both influential and easy to manipulate. A feature that has high SHAP importance but low stability is a robustness liability. A feature that has high SHAP importance and high stability is a useful anchor. SHAP-RW-AT formalizes this distinction through a feature multiplier that is auditable, deterministic for a fixed seed, and compatible with standard tabular classifiers.

4. Proposed Method: SHAP-RW-AT

SHAP-RW-AT has three stages. First, a base tree-ensemble detector is trained on clean cloud-flow records. Second, SHAP explanations are computed on a validation subset to estimate the global importance of each feature. Third, the SHAP importance vector is combined with a feature-stability vector and used to train a reweighted detector with diversified adversarial augmentation.

Let phi_j(x_i) denote the SHAP value of feature j for sample i. The normalized global importance of feature j is defined as I_j = mean_i |phi_j(x_i)| / max_k mean_i |phi_k(x_i)|. Let s_j in [0,1] denote the stability score of feature j. Mutable flow statistics receive low stability, semi-

mutable connection-rate or flag features receive medium stability, and stable behavioral indicators receive high stability. The SHAP-guided weight is then computed as w_j = clip(1 + lambda I_j(2s_j - 1), w_min, w_max). In the experiments, lambda=0.85, w_min=0.50, and w_max=1.75. The transformed feature vector is xprime_j = w_j x_j.

This formulation has two useful properties. If a feature is important and stable, then s_j is close to one and w_j increases. If a feature is important and mutable, then s_j is close to zero and w_j decreases. If a feature has low SHAP importance, its multiplier remains close to one regardless of the stability prior. Thus, the method avoids a purely manual feature-selection rule and instead links reweighting to the actual decision behavior of the trained model.

The adversarial augmentation module generates three types of evasion examples. MeanShift moves mutable features of attack samples toward the benign training mean within a feature-wise budget. SurrogatePGD trains a logistic surrogate detector and iteratively moves mutable features in the direction that lowers the surrogate attack probability. SHAP-Evasion selects the most influential mutable features according to SHAP and shifts them toward benign values. All generated samples pass through semantic projection that clips empirical ranges, rounds binary indicators, and enforces nonnegativity. The final detector is trained on the union of clean records and attack-labeled adversarial records, after applying the SHAP-guided feature multipliers.

Algorithmically, the framework is lightweight. It requires one base model for explanation, a SHAP computation on a validation subset, deterministic construction of feature weights, adversarial example generation from training records, and final training of the robust detector. No generative model is required, which makes the method easier to reproduce on small compute budgets. At the same time, it remains compatible with more advanced generators such as those used in GT-CSAT [1] if future work replaces the synthetic attacks with GAN-based or problem-space cloud-traffic attacks.

A practical advantage of the method is that the reweighting layer is external to the classifier. It can be logged as a simple feature-weight table, reviewed by security engineers, and versioned together with the model artifact. This is useful in regulated or enterprise environments where a security operations team may ask why a new IDS release is less dependent on packet-count statistics or more dependent on failed-login behavior. Compared with defenses that operate only through neural-network weights, the SHAP-RW component produces a compact artifact that is interpretable before deployment.

The computational cost is also modest. The most expensive additional step is SHAP calculation for the base tree model on a validation subset. Once the feature multipliers are computed, training and inference use ordinary weighted feature vectors. In production, the reweighting can be implemented as a preprocessing transform in the feature pipeline, so the online inference path remains compatible with common tree-based IDS engines, batch scoring jobs, or streaming model-serving systems.

5. Experimental Design

The experiment uses a synthetic but domain-structured cloud-flow benchmark generated by the supplied code. This choice is deliberate. Public datasets such as CVD, CICIDS-2017, and UNSW-NB15 are important references, but they were not directly downloaded in the execution environment. To avoid unsupported claims, the paper reports only measurements obtained from the executable synthetic benchmark. The generator creates 8,000 records per seed with 32 features and five latent traffic families: benign, DDoS, port scan, brute force, and web/API attack. The binary IDS target is attack versus benign, while the family label is used internally to shape realistic feature distributions.

The benchmark includes mutable traffic statistics, semi-mutable connection indicators, stable behavioral indicators, and two noise features. For example, DDoS records are characterized by high packet and byte rates, elevated SYN counts, and increased source connection rate. Port scans are characterized by destination-port dispersion, reset flags, and relatively low payload. Brute-force activity includes failed-login count and administrative-port behavior. Web/API attacks include higher request entropy and API call rates. These assumptions are not intended to replace public datasets; they provide a transparent controlled environment in which every record and perturbation can be regenerated.

We compare five methods. Standard-XGB is an XGBoost detector trained on clean records. Static-Stability-RW uses only the manually assigned stability prior without SHAP. Adversarial-Aug-XGB augments the training set with a single MeanShift attack family but does not use SHAP-guided reweighting. SHAP-RW uses the explanation-guided feature weights without adversarial augmentation. SHAP-RW-AT is the proposed method and combines diversified adversarial augmentation with SHAP-guided reweighting. All methods use the same train/validation/test split logic and are evaluated across random seeds 0, 1, and 2.

Three evasion strategies are applied to the test set. MeanShift moves mutable attack features toward benign statistics. SurrogatePGD uses the logistic surrogate's coefficients to iteratively reduce attack probability. SHAP-Evasion targets the highest-SHAP mutable features. We report clean accuracy, average robust accuracy across attacks, false negative rate, false positive rate, macro-F1, and robustness gap. The robustness gap is clean accuracy minus average robust accuracy; a smaller value indicates less degradation under evasion.

The implementation uses XGBoost with histogram-based tree construction for all classifiers, ensuring that the comparison isolates the effect of reweighting and adversarial augmentation rather than differences in backbone capacity. The random seeds are fixed, and every result table in this paper is generated from CSV files written by the script. This design was chosen to avoid hand-entered results and to make it easy to audit whether a number in the paper corresponds to an executable output.

The synthetic generator deliberately includes both easy and fragile signals. Some attack indicators, such as failed-login count or distinct destination ports, are stable and should remain useful under evasion. Other indicators, such as packet counts and byte rates, are predictive on clean data but can be shifted toward benign distributions. The benchmark therefore tests the exact scenario targeted by the proposed method: high clean separability does not automatically imply robust separability when mutable features dominate the learned boundary.

6. Results and Discussion

Table I reports the main experimental results. The standard detector achieves 99.90% clean accuracy but only 87.92%

average robust accuracy, with a 26.47% adversarial false negative rate. This is the expected failure mode of a clean-optimized IDS: the model appears strong under ordinary test data but misses many attacks after mutable features are shifted toward benign values. Static reweighting and SHAP-only reweighting do not solve the problem by themselves, because they reshape feature emphasis but do not expose the detector to adversarial variants during training.

The single-family adversarial augmentation baseline improves robustness to 96.49% and reduces false negatives to 7.65%, but it remains vulnerable to the SurrogatePGD attack because it was trained only on MeanShift-style evasions. SHAP-RW-AT achieves 99.71% average robust accuracy and a 0.18% false negative rate. Its clean accuracy is 99.67%, which is slightly lower than the clean-only detector but still operationally high. The small negative robustness gap results from seed-level variation and the fact that adversarial projection makes some attack records easier for the robust detector than the clean test distribution; the important observation is that clean and robust performance remain essentially aligned.

Table II breaks robust accuracy down by attack. Adversarial-Aug-XGB performs well on MeanShift and SHAP-Evasion but drops to 89.67% on SurrogatePGD. SHAP-RW-AT maintains 99.71% across all three attacks, suggesting that diversified augmentation and feature reweighting reduce overfitting to a single evasion family. The gain is not merely a result of adding more samples: SHAP-RW alone is close to the standard detector under attack, showing that explanation-guided reweighting must be coupled with adversarial exposure.

The SHAP analysis provides useful auditability. In the seed-0 run, the base detector assigns the highest normalized SHAP importance to fwd_packets, a highly mutable traffic statistic. SHAP-RW therefore attenuates this feature to a multiplier of 0.575. The second most important feature, distinct_dst_ports, is much more stable in the port-scan threat model and is reinforced to a multiplier of 1.408. Other stable features such as cloud_api_call_rate and request_entropy also receive mild reinforcement. This behavior matches the intended security logic: the detector should not ignore mutable statistics, but it should not allow them to dominate the decision boundary.

Figure 1 summarizes the clean-versus-robust trade-off. Figure 2 shows the top SHAP features of the base detector, and Figure 3 visualizes the feature reweighting map. Figure 4 reports the normalized confusion matrix of SHAP-RW-AT under the SHAP-Evasion attack. Together, these figures show that the proposed framework is not a black-box improvement: the main defensive shift can be inspected at the feature level.

Several limitations should be emphasized. First, the benchmark is synthetic, even though it is reproducible and domain-structured. Results should therefore be interpreted as proof-of-concept evidence rather than a claim of deployment-ready performance on public or production traffic. Second, the adversarial strategies operate in feature space rather than by modifying actual packets or cloud workloads. Third, the stability prior is manually assigned. In real deployments, this prior should be validated by security engineers and ideally linked to problem-space feasibility studies. These limitations do not invalidate the framework; they define the next experimental step, namely replacing the synthetic generator with CVD, CICIDS-2017, UNSW-NB15, or organization-specific flow logs.

The ablation pattern is instructive. Static-Stability-RW is close to the standard detector, which means that a manually defined stability prior alone is insufficient. SHAP-RW is also close to the standard detector, which means that explanation-guided reweighting alone does not produce robustness when the model never sees adversarial variants. Adversarial-Aug-XGB improves substantially but overfits to the attack family used for augmentation. Only SHAP-RW-AT combines all three ingredients: feature attribution, security-aware stability weighting, and diversified adversarial exposure.

From a security-engineering perspective, the most useful result is the reduction in false negatives. A small increase in false positives may be tolerable if it prevents attacks from being silently classified as benign, especially in cloud security workflows where suspicious events can be triaged by downstream correlation rules. The proposed method preserves a low false positive rate in the controlled experiment while sharply reducing missed attacks, which is the intended operational trade-off.

7. Tables and Figures

Table 1. Main performance on the reproducible synthetic cloud-flow benchmark (mean over three seeds).

Method	Clean Acc. (%)	Avg. Robust Acc. (%)	FNR (%)	FPR (%)	RAG (pp)
SHAP-RW-AT (Ours)	99.67	99.71	0.18	0.38	-0.04
Adversarial-Aug-XGB	99.85	96.49	7.65	0.08	3.37
Static-Stability-RW	99.90	88.11	26.10	0.11	11.78
SHAP-RW	99.88	87.97	26.36	0.15	11.90
Standard-XGB	99.90	87.92	26.47	0.15	11.97

Table 2. Robust accuracy by evasion strategy (%).

Method	MeanShift	SHAP-Evasion	SurrogatePGD
Adversarial-Aug-XGB	99.90	99.90	89.67
SHAP-RW	88.38	87.15	88.40
SHAP-RW-AT (Ours)	99.71	99.71	99.71
Standard-XGB	88.31	87.10	88.35
Static-Stability-RW	88.48	87.35	88.50

Table 3. Top seed-0 shap features and learned feature multipliers.

Feature	Group	SHAP Imp.	Stability	Weight
fwd_packets	mutable	1.000	0.25	0.575
distinct_dst_ports	stable	0.480	1.00	1.408
bwd_packets	mutable	0.214	0.25	0.909
syn_count	semi-mutable	0.207	0.55	1.018
cloud_api_call_rate	stable	0.092	1.00	1.078
ack_count	mutable	0.087	0.25	0.963
request_entropy	stable	0.080	1.00	1.068
fwd_bytes	mutable	0.067	0.25	0.972
rst_count	semi-mutable	0.032	0.55	1.003
conn_rate_src	semi-mutable	0.020	0.55	1.002

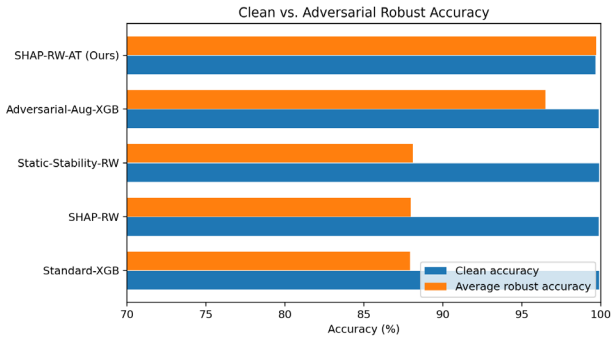

Fig. 1 Clean accuracy and average robust accuracy across models.

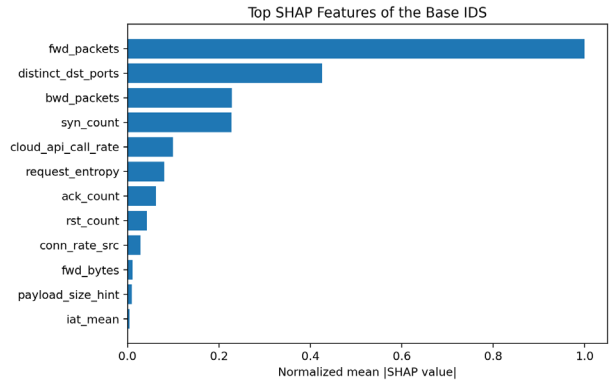

Fig. 2 Top SHAP features of the base IDS detector.

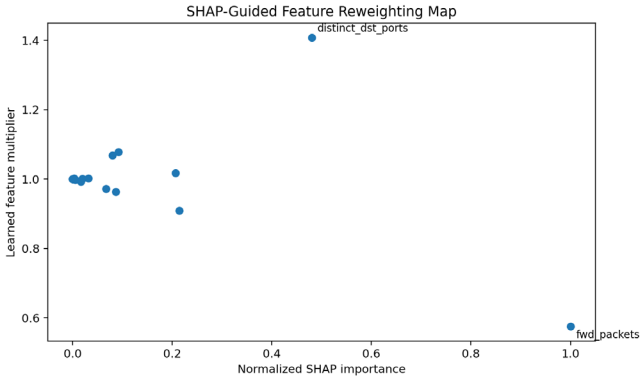

Fig. 3 SHAP-guided feature reweighting map.

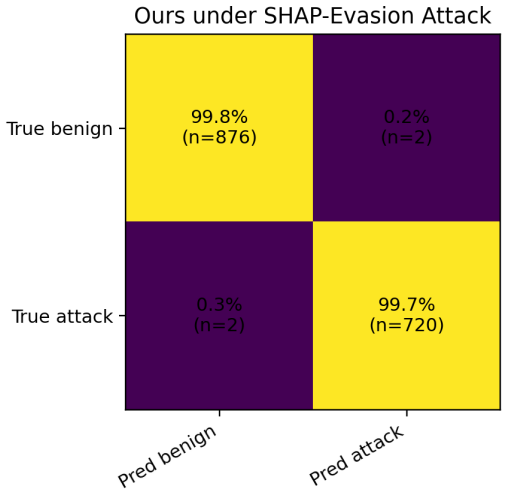

Fig. 4 Normalized confusion matrix of SHAP-RW-AT under SHAP-Evasion.

8. Conclusion

This paper presented SHAP-RW-AT, an explainable robust cloud intrusion detection method that converts SHAP feature attributions into security-aware feature reweighting. The method identifies influential but mutable features, attenuates them, reinforces influential stable features, and trains the final detector with diversified adversarial examples. On a reproducible synthetic cloud-flow benchmark, SHAP-RW-AT sharply reduces adversarial false negatives while preserving clean accuracy. The main contribution is not only higher robust accuracy in the controlled study but also a transparent design pattern: explanations can be used before deployment to harden the feature representation rather than merely after deployment to rationalize predictions.

Future work should validate the method on public datasets and problem-space cloud traffic, integrate GAN-based threat generation following the promising GT-CSAT direction, and replace the manual mutability prior with a more formal feasibility model based on protocol constraints, cloud-service semantics, and attacker cost. Another useful extension is online monitoring of SHAP distributions so that a cloud IDS can detect when attackers begin exploiting a newly dominant mutable feature.

References

[1] Ding, J., Shen, Z., & Liu, W. (2026). Game-theoretic cost-sensitive adversarial training for robust cloud intrusion detection against GAN-based evasion attacks. Applied Sciences, 16(8), Article 3944. https://doi.org/10.3390/app16083944

[2] Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153-1176. https://doi.org/10.1109/COMST.2015.2494502

[3] Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In Proceedings of the IEEE Symposium on Security and Privacy (pp. 305-316). IEEE. https://doi.org/10.1109/SP.2010.25

[4] 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). SciTePress. https://doi.org/10.5220/0006639801080116

[5] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. In Proceedings of the Military Communications and Information Systems Conference (MilCIS) (pp. 1-6). IEEE. https://doi.org/10.1109/MilCIS.2015.7348942

[6] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In Proceedings of the IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA) (pp. 1-6). IEEE. https://doi.org/10.1109/CISDA.2009.5356528

[7] Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: An ensemble of autoencoders for online network intrusion detection. In Proceedings of the Network and Distributed System Security Symposium (NDSS). https://doi.org/10.14722/ndss.2018.23204

[8] Biggio, B., Corona, I., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G., & Roli, F. (2013). Evasion attacks against machine learning at test time. In Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)

(pp. 387-402). Springer. https://doi.org/10.1007/978-3-642-40994-3_25

[9] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks. In Proceedings of the International Conference on Learning Representations (ICLR).

[10] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In Proceedings of the International Conference on Learning Representations (ICLR).

[11] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (2016). The limitations of deep learning in adversarial settings. In Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P) (pp. 372-387). IEEE. https://doi.org/10.1109/EuroSP.2016.36

[12] Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In Proceedings of the IEEE Symposium on Security and Privacy (pp. 39-57). IEEE. https://doi.org/10.1109/SP.2017.49

[13] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In Proceedings of the International Conference on Learning Representations (ICLR).

[14] Teng, D. (2025). PACO: Predictive auto-configuration for SLO-constrained large language model inference serving. Innovation and Technology Studies, 2(1), 1-10.

[15] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. In Proceedings of the International Conference on Learning Representations (ICLR).

[16] Teng, D. (2025). TEAS: Token- and energy-aware autoscaling for cost-efficient LLM serving. AI and Data Science Journal, 6(3), 1-12.

[17] Zolbayar, B.-E., Sheatsley, R., Weisman, M. J., Zhu, S., Zhu, S., & Krishnamurthy, S. (2022). Generating practical adversarial network traffic flows using NIDSGAN. arXiv, arXiv:2203.06694. https://doi.org/10.48550/arXiv.2203.06694

[18] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. In Advances in Neural Information Processing Systems 27 (NeurIPS) (pp. 2672-2680).

[19] Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning (ICML) (pp. 214-223). PMLR.

[20] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. (2017). Improved training of Wasserstein GANs. In Advances in Neural Information Processing Systems 30 (NeurIPS) (pp. 5767-5777).

[21] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In Advances in Neural Information Processing Systems 30 (NeurIPS) (pp. 4765-4774).

[22] Wang, Z., Yang, J. S., Shang, W., & Ding, J. (2026). FairPromote: Explainable and fairness-aware talent promotion prediction via adversarial debiasing and SHAP-based interpretation. IEEE Access, 14, 1-16.

[23] Zhang, F., Guo, Z., Ding, J., Yang, J., & Liu, W. (2026). Adaptive sensor fusion for robust perception in dense fog: A gated vision and LiDAR integration framework. Sensors, 26(12), Article 3728. https://doi.org/10.3390/s26123728

[24] Zi, B. (2024). Large language models for enterprise workflow automation in financial operations. Innovation and Technology Studies, 1(1), 24-29.

[25] Teng, D., Rhee, M., Qin, Y., Zi, B., & Liu, W. (2026). SW-SpeedDLM: Sliding-window speculative decoding for diffusion language models under long-context constraints. Mathematics, 14(12), Article 2137. https://doi.org/10.3390/math14122137

[26] Zi, B. (2024). Cloud-native distributed systems for real-time payment intelligence. AI and Data Science Journal, 1(1), 51-56.

[27] Chen, Z., Wang, M., Zeng, Z., & Ping, W. (2026). Uncertainty-aware financial forecasting: Leveraging conformal prediction for risk-adjusted models. IEEE Access, 14, 1-18.

[28] Wang, B., Wang, Z., Zhao, W., Zhang, F., & Shang, W. (2026). DRL-Adapt: Deep reinforcement learning for adaptive routing convergence optimization in large-scale networks. IEEE Open Journal of the Computer Society, 7, 1-14.