InfoTech Spectrum: Iraqi Journal of Data Science (IJDS)
ISSN: 3007-5467, Vol 3(No.2), 67-78 (2026)
DOI: https://doi.org/10.51173/ijds.v3i2.84
https://ijds.mtu.edu.iq

AN EXPLAINABLE AI-BASED ALERT TRIAGE FRAMEWORK FOR SECURITY OPERATIONS CENTRES: ARCHITECTURE, EVALUATION, AND ANALYST-CENTRED DESIGN

Shubham Vilas Poinkar [1*], **Majharoddin Kazi** [2],

[1] *Department of Computer Science and Application, JSPM University, Pune, India*
[2] *Faculty of Science and Technology, JSPM University, Pune, India*

** Corresponding author E-mail: shubhampoinkar@gmail.com* ***(Shubham Vilas Poinkar)***

RESEARCH ARTICLE

ARTICLE INFORMATION

SUBMISSION HISTORY:
Received: 24 May 2026
Revised: 25 June 2026
Accepted: 27 June 2026
Published: 30 June 2026

KEYWORDS:
Explainable Artificial Intelligence (XAI);
Security Operations Center (SOC);
Alert Triage;
SHAP;
Intrusion Detection

ABSTRACT

Security operations centers (SOCs) today are overwhelmed by a continually growing volume of security alerts from a variety of detection technologies. Analysts typically receive thousands of alerts per shift. Many of these alerts are false positives or low-priority events that cause serious analyst fatigue, delayed incident response, and increased organizational risk. The false-positive rate is typically greater than 50% in business environments. This paper proposes and evaluates ExTriage, a novel Explainable Artificial Intelligence (XAI)-based alert triage framework integrating: (i) a high-performance ensemble classifier combining XGBoost and a multi-layer perceptron; (ii) a SHAP-based global and local explanation engine; (iii) a counterfactual explanation module; and (iv) a RankNet-inspired alert prioritization module. Evaluated on CICIDS-2017, CICIDS-2018, and UNSW-NB15, ExTriage achieves a macro F1-score of 97.89% and AUC-ROC of 0.9963 while reducing false-positive rates by 70.8%. In a controlled study with 24 professional SOC analysts, average triage time was reduced from 47 to 8.2 seconds per alert, with statistically significant improvements in trust calibration, decision confidence, and alert disposition accuracy.

1. INTRODUCTION

Digital infrastructures, enterprise applications, cloud services, and distributed communication systems are becoming the backbone of modern organizations. Technological developments have increased productivity within organizations and enabled worldwide connectivity; however, they have also introduced organizations to the potential for Cyber Crime. Cyber-attacks include unauthorized access attempts, insider threats, ransomware, malware, and advanced persistent threats, all of which continually target enterprise systems to compromise the integrity of sensitive data and disrupt operational continuity. All digital systems produce operational logs. The Records created by these logs document virtually all aspects of how end users interact with systems; for example, they document user activity, authentication events, application behavior, communication sessions, system process execution, and network connectivity. Monitoring operational logs provides organizations with essential visibility into operational activities within the enterprise environment; thus, effective log monitoring is critical to managing cybersecurity, since many indicators of suspicious operational behavior appear in logs before cybersecurity incidents are fully visible.

Traditional monitoring solutions depended heavily on manual log analysis and reviewing isolated event occurrences. However, due to the modern enterprise infrastructure generating tremendous amounts of logs across multiple distributed systems, manual log analysis is becoming increasingly difficult and inefficient. In line with this, organizations now need a centralized log. Monitoring solutions that can continuously gather, analyze, and correlate. Real-time operational

DOI: https://doi.org/10.51173/ijds.v3i2.84

events. Log monitoring tools and SIEM platforms improve Enterprise Security Operations by providing centralized visibility into events, automatic alert generation, behavioral analysis, and threat Intelligence correlation, which allows companies to detect suspicious operational behavior and respond to cybersecurity incidents more quickly. Kent and Souppaya (2006) stressed that implementing a centralized log management system greatly enhances the organization's security visibility and forensic investigation capabilities [1]. As Freitas et al. [2] describe, SIEM technologies enhance threat monitoring for enterprises by providing intelligence through event correlation and real-time analytics.

New developments in cybersecurity analytics have yielded new technologies for monitoring environments against cyber threats. There now exist AI-based monitoring frameworks, systems for analyzing event behavior, and cloud-based monitoring solutions that enable monitoring a much larger volume of data more efficiently than previously possible. Some examples of open-source platforms that have enjoyed significant adoption due to their inherent flexibility and scalability include ELK Stack, Graylog, and Wazuh. At the same time, enterprise solutions such as SIEM systems offer more comprehensive automation capabilities and integrated threat intelligence. Despite these advancements in the cybersecurity industry, organizations continue to struggle to find a suitable monitoring platform due to significant differences among available products, including deployment complexity, scale, user-friendliness, analytical intelligence capabilities, and operational performance. For example, some monitoring tools are very effective at providing graphical displays or visualizations of events; however, they have limited capabilities for analyzing behavioral or activity-related data. Alternatively, other tools can provide robust capabilities for analyzing events using advanced threat intelligence, but at the expense of requiring a high volume of infrastructure to perform optimally.

The current research study will therefore present a comparison of major tools for monitoring logs of cybersecurity events to provide insight into how effective each of these tools performs from an operational perspective, the level of analytical intelligence provided by each tool, and the degree to which each tool is practically suitable for use in an enterprise environment [3], [4]. The digital threat landscape has grown structurally complex. By 2024, enterprise-grade Security Information and Event Management (SIEM) platforms were reported to generate between 1,000 and 10,000 alerts per day for mid- to large-sized organizations [1]. The signal-to-noise ratio within that flood is frequently poor: studies consistently report that between 45% and 99% of security alerts are false positives, depending on the maturity of the detection infrastructure [2], [3]. This illness, alert fatigue, has real operational consequences. Security operations centers (SOCs) that are either understaffed or overwhelmed incur breach costs that are, on average, $1.44 million above the worldwide mean [4] in the 2023 IBM Cost of a Data Breach Report. Several times, machine learning (ML) has been proposed as a solution to automation. Neural networks, ensemble classifiers, and anomaly detection algorithms have achieved accuracy levels close to, and in some cases exceeding, human performance when tested on benchmark intrusion detection datasets. The use of these systems in operational SOCs remains quite limited. The most crucial hurdle is the lack of transparency: current machine learning classifiers work like black boxes, providing binary or probabilistic outcomes but unable to elucidate the reasoning behind their decisions [5]. Besides a label (benign or malevolent), SOC analysts need an explanation that includes the characteristics that led to the classification, the model's confidence level, and the factors that would need to change for the classification to be different [6].

So the discipline's response to this lack of transparency has been explainable artificial intelligence, or XAI. In particular, SHAP [7] and LIME [8] are examples of post-hoc explanation strategies that provide model-agnostic feature attribution for individual predictions. The counterfactual explanation approaches go further by identifying the minimum perturbations to the input that would change the expected outcome [9]. However, adapting XAI techniques to operational SOC constraints, such as real-time latency, high alert volume, analyst cognitive load, and SIEM/SOAR integration, remains a focus of ongoing research [10], [11]. This work closes this gap through three independent contributions. ExTriage is a full-fledged, multi-component XAI triage framework specifically built for the SOC alert workflow. This is the first thing that we recommend. Second, we do an extensive empirical examination of the CICIDS-2017, CICIDS-2018, and UNSW-

NB15 standards. The evaluation consists of ablation experiments, cross-dataset generalization tests, and adversarial robustness probes. Thirdly, we provide the results of a human factors study conducted with twenty-four expert SOC analysts. What the research is aimed at is:

- RQ1: How well can an XAI-augmented triage pipeline achieve classification performance comparable to state-of-the-art black-box models, with the added advantage of a per-alert explanation lag of less than 5 seconds?
- RQ2: Does the combination of SHAP-based feature attribution and counterfactual explanations produce a measurable decrease in the number of false-positive alert dispositions performed by SOC analysts over unaided or label-only help?
- RQ3: What are the required architectural and human-computer interaction design principles for XAI-triage tools to be accepted by experienced security analysts and to be used appropriately?

As to the other parts of this book, their organization is as follows. Section 2 describes the datasets used. The report includes a literature review (Section 3). Section 4 explains the extriage methodology. Results are reported in Section 5. The results are presented in Section 6. Section 7 concludes.

2. DATASET DESCRIPTION AND SELECTION JUSTIFICATION

2.1 Overview of Available Datasets

It was decided to conduct an extensive analysis of publicly available datasets on network intrusion and host-based security. Table I summarises the main options identified in the literature.

Table 1. Comparative summary of security benchmark datasets

Dataset	Records	Attack Types	Features	Imbalance	Public
CICIDS-2017	2.8M flows	14 categories	79	High	Yes
CICIDS-2018	6.3M flows	7 categories	80	Very High	Yes
NSL-KDD	125,973	4 categories	41	Moderate	Yes
UNSW-NB15	2.5M packets	9 categories	49	High	Yes
BETH Dataset	~1M logs	Malicious process	47	Extreme	Yes
DAPT 2020	35,000 events	APT lateral	38	Extreme	Yes
CTU-13	~2.8M flows	13 botnet	14	High	Yes

2.2 Selected Datasets and Rationale

The main corpora selected for cross-dataset generalization testing are CICIDS-2017 and CICIDS-2018. UNSW-NB15 was also used. The selection process considered both criteria:

a. Sufficient volume for imbalanced learning evaluation.
b. Multiple MITRE ATT&CK-aligned attack categories.
c. Fine-grained per-record ground truth labels.
d. Demonstrated use in 2022–2025 XAI-security literature.
e. Unrestricted public access for reproducibility.

2.3 Dataset Characteristics and Pre-processing

The CICIDS-2017 dataset was generated by the Canadian Institute for Cybersecurity using CICFlowMeter to extract 79 features of bidirectional flows from five days of recorded traffic. The features included DoS, DDoS, Heartbleed, Web Attack, Infiltration, Botnet, and Port Scan scenarios. CICIDS-2018 extends this paradigm with seven attack categories and approximately 6.3 million records. Both datasets exhibit severe class imbalance: benign traffic constitutes approximately 83% of CICIDS-2017 and 77% of CICIDS-2018. Pre-processing steps included: removal of infinite and NaN values (replaced with column medians), standard z-score normalization, Synthetic Minority Over-sampling Technique (SMOTE) applied selectively to minority classes with fewer than 5,000 instances, and label consolidation into a 10-class taxonomy aligned with MITRE ATT&CK initial access and execution tactics.

2.4 Dataset Partitioning and Validation Strategy

Each benchmark dataset (CICIDS-2017, CICIDS-2018, and UNSW-NB15) was independently split using stratified sampling to maintain the original dataset's class distribution across all subsets, thereby avoiding biased performance evaluation and enabling robust model training. The datasets were split into 70% training, 15% validation, and 15% test sets. Model learning was performed using the training set, and the validation set was used for hyperparameter optimization, model selection, and early stopping to avoid overfitting. To obtain an unbiased estimate of the generalization capabilities, we evaluated the final model's performance only on the unseen test set. Stratified partitioning was particularly crucial, as the benchmark datasets are highly imbalanced, potentially leading to under-representation of minority attack types in each subset. Five-fold stratified cross-validation was performed on the training dataset during model construction. In each fold, 80% of the training data was used for model fitting and 20% for validation. The best model configuration was chosen based on the average performance across the five folds. After hyperparameter optimization, the final model was retrained on the entire training dataset and tested on the independent test set.

3. LITERATURE REVIEW

Sommer and Paxson [12] set the ground rules: anomaly detectors produce outputs in registers of information that differ from those used by procedural reasoning representations, leading to a semantic mismatch. [14] measured it as alert fatigue: the ratio of actionable to non-actionable alerts negatively affects analysts' performance regardless of the number of alerts. [15] found that analysts who received 500+ alerts per day experienced a measurable loss of accuracy after 90 minutes, with an additional 19% committing more inaccurate dispositions than those receiving fewer than 100 prioritized alerts per day.

On CICIDS benchmarks, ensemble methods, in particular Random Forest, Gradient Boosting, and XGBoost, have consistently outperformed other methods in terms of accuracy, with several studies reporting F1 scores>97% [16-18]. All these approaches, however, have explainability deficits. However, technical accuracy is required but not sufficient for SOC deployment, as shown by [19], which found that 78% of SOC analysts rejected a LightGBM model with an accuracy of 98.2% on CICIDS-2018 due to its lack of explainability. In [20], the authors compared SMOTE variants, cost-sensitive learning, and threshold calibration strategies across seven benchmark datasets. They showed that the best minority recall is achieved with cost-sensitive XGBoost with calibrated thresholds, which aligns with the ExTriage classifier's design.

Since 2021, adoption of XAI in cybersecurity has increased significantly. Table II provides a comparative synthesis of key studies and the proposed framework. The SHAP framework [7], [21] is the most popular post-hoc explanation technique in the cybersecurity literature, due to its guarantees of game-theoretic consistency and low computational overhead for tree-based models. Warnecke et al. [23] compared SHAP, LIME, and Anchors across four security classification tasks and found that rule explanations generated by Anchors were preferred for low-complexity alerts, and SHAP was preferred for high-dimensional alerts.

Prioritization differs from classification: classification assigns a label to the alert's category; prioritization provides an urgency score that directs the analyst's attention. There are well-known limitations of rule-based prioritization in dynamic threat environments [24-27]. The present framework introduces a novel contribution by applying RankNet [28] to alert prioritization, accounting for classification confidence and contextual threat intelligence. Dodge et al. [29] demonstrated that analysts' mental models influence the explanation utility function. Bansal et al. [30] pointed out automation bias in XAI-driven decisions. The ExTriage human factors study is addressing these questions by using the primary outcome measure, trust calibration, based on the theoretical model of appropriate reliance proposed by Parasuraman and Riley [31].

The synthesis identifies three current gaps:

a. No prior study has combined classification, SHAP explanation, counterfactual reasoning, and prioritization under a single, operationally evaluated framework.

b. There has been a lack of human factors evaluations, which are generally conducted on

a small scale with simulated participants.

c. Adversarial robustness of XAI explanations has not been systematically addressed in SOC triage. ExTriage directly addresses the first two gaps, and this is preliminary evidence for the third.

4. METHODOLOGY

4.1 Research Design and Framework Architecture

The quantitative experimental research design will be complemented with a controlled human factors study in ExTriage. The framework architecture comprises four tightly integrated modules: the Classification Engine, the SHAP Explanation Engine, the Counterfactual Guidance Module, and the Priority Ranking Module.

4.2 Classification Engine

The classification engine has a stacked ensemble architecture. A gradient-boosted tree (500 samples, learning rate 0.05, max depth 8, cost-sensitive weight) based on the base learner XGBoost (v2.0). We trained the meta-learner as a three-layer Multi-layer Perceptron (256-128-64 neurons with ReLU activation and dropout 0.3) on out-of-fold probability outputs using five-fold cross-validation. Hyperparameters were optimized using Bayesian optimization (Optuna, 100 trials). The classification target was a 10-class taxonomy of MITRE ATT&CK: Benign, DoS-Hulk, DoSGoldenEye, DDoS, PortScan, Brute Force, XSS, SQL Injection, Infiltration, and Botnet. For each class, a calibrated probability output (isotonic regression) was developed.

4.3 SHAP Explanation Engine

The SHAP engine used for XGBoost as a base learner is based on TreeSHAP [21], a polynomial-time algorithm for computing exact Shapley values. DeepSHAP is used for deep-layer attribution for the MLP meta-learner. The Global SHAP summary plots are pre-computed nightly, and local waterfall and force plots are computed on demand within the explanation latency budget. An important design feature was the use of a Least Recently Used (LRU) cache of 10,000 results, with the key being a feature vector hash, which significantly reduced the median time for the SHAP to respond to repeat alerts for the same collection of features to 18ms, down from 210ms. The results of the Explanation fidelity validation (based on the infidelity and sensitivity metrics from Yeh et al. [32]) are presented, with the infidelity scores for all attack categories below 0.02.

4.4 Counterfactual Guidance Module

The module identifies x' with f(x') not equal to c, such that the L1 distance d(x, x') is minimized subject to feasibility constraints and with a sparse perturbation (less than five features changed). The generation algorithm is modified from [33] by incorporating domain-specific constraints specified by the network protocol. A counterfactual is represented as a natural-language template that looks like: "If [feature A] was [value] instead of [value], this alert was classified as [lower severity class]."

4.5 Priority Ranking Module

A continuous urgency score u(x) is computed for each alert combining:

a. Maximum probability of each ensemble classifier.

b. Top-3 feature criticality score, using SHAP analysis.

c. A multiplier from the MITRE ATT&CK framework for the alert stage.

d. A temporal recency factor that penalizes stale alerts (default threshold: 15 minutes). The component weights were obtained by minimizing a pairwise ranking loss function inspired by RankNet, trained on 12,400 pairs of historical analyst-labeled alerts from a cooperating financial institution.

4.6 Human Factors Study

A within-subjects study was conducted with 24 financial services, healthcare IT, and government-sector professional SOC analysts. Participants were presented with three types of alert triage:

a. Label only.
b. Label + SHAP explanation.
c. Label + SHAP + Counterfactual. 30 alerts were used for each condition to ensure equal numbers of true positives, false positives, and borderline alerts. Primary outcomes were triage accuracy, trust calibration (Brier score), and average triage time. Repeated-measures ANOVA with Bonferroni correction (α=0.05) was used to perform the statistical analysis.

4.7 Evaluation Metrics

Macro-averaged Precision, Recall, F1-Score, and AUC-ROC were employed for model evaluation. The end-to-end alert triage latency was evaluated for a containerized deployment (Docker, 8 vCPUs, 32 GB RAM, NVIDIA T4 GPU). On 500 test alerts, Adversarial robustness was evaluated using JSMA feature perturbations.

4.8 Validation Strategy and Model Selection

4.8.1 Stratified 5-Fold Cross-Validation:

During model development, a stratified five-fold cross-validation was applied to obtain an accurate evaluation of the model and decrease the risk of overfitting. The training dataset was split into 5 mutually exclusive folds, and the original distributions of the attack and benign classes were preserved within each fold. The model was trained in each iteration using four folds (80% of the total), with the last fold (20% of the total) used as the validation set. This was repeated five times, so each sample was used once for validation and four times for training. This was done to ensure the results were accurate. The final cross-validation performance was computed as the average of the evaluation metrics across all five folds to obtain a reliable estimate of the model's generalization capabilities.

4.8.2 Validation Dataset Usage:

The dataset was split into training (70%), validation (15%), and testing (15%) data. The validation data was only used for model development (hyperparameter optimization, model selection, probability calibration, and early stopping). The validation set was not used to update model parameters, ensuring an unbiased evaluation during optimization. The model was trained on the whole training dataset using the best model configuration, and then evaluated on the independent test dataset.

4.8.3 Prevention of Data Leakage:

To protect the integrity of the experimental evaluation and avoid data leakage, many safeguards were put in place. The dataset was split before any preparation operations were undertaken. The feature normalization parameters, such as the mean and standard deviation for z-score normalization, were computed only on the training data and then applied to the validation and test datasets. Likewise, after the division, the Synthetic Minority Over-sampling Technique (SMOTE) was applied only to the training subset to avoid contaminating the validation and test sets with synthetic data. Hyperparameter optimization, feature selection, and model calibration were performed only on the training and validation data, and the independent test set was withheld until the final performance evaluation.

4.8.3 Model Selection Criteria:

The model selection was performed based on the average performance from stratified five-fold cross-validation. The hyperparameters were tuned via Bayesian optimization with the Optuna framework, using 100 optimization trials. The optimization target was to maximize the macro-averaged F1-score, which is a balanced performance measure across the majority and minority classes in highly imbalanced intrusion detection datasets. Other evaluation parameters, such as Precision, Recall, AUC-ROC, and False Positive Rate (FPR), were considered to ensure a balanced trade-off between detection accuracy and false alarm reduction. We chose the final ExTriage classifier as the model with the highest validation macro F1-score and the highest stability across all folds.

5. RESULTS AND ANALYSIS

5.1 Classification Performance

Table 2 shows the classification performance of the proposed system. The CICIDS2018 held-out test set consists of 126,000 records split into a stratified test set. The macro F1 and AUC-ROC of the complete ExTriage framework are 97.89% and 0.9963, respectively. The incremental contribution of each module is confirmed by ablation analysis. SHAP provides a slight increase in the marginal classification gain (+0.06%), but, most importantly, in human factors outcomes. The counterfactual module is responsible for 1.27% of the score. The RankNet module doesn't change any classification metrics, but it can reduce analyst queue errors by pushing high-confidence False Positives to the bottom of the queue.

Table 2. Classification and triage performance comparison

Model Variant	Accuracy	Precision	Recall	F1	AUC-ROC	Triage Time
Baseline XGBoost (no XAI)	97.12%	96.84%	97.40%	97.12%	0.9921	N/A
XAI-Triage (SHAP only)	97.18%	97.03%	97.34%	97.18%	0.9928	2.3 s
XAI-Triage (SHAP + CF)	97.45%	97.29%	97.61%	97.45%	0.9941	3.1 s
XAI-Triage (Full)	97.89%	97.72%	98.06%	97.89%	0.9963	3.8 s
Analyst-Only Baseline	N/A	88.10%	82.30%	85.10%	N/A	~47 s
Human + XAI-Triage	N/A	96.20%	95.80%	96.00%	N/A	8.2 s

5.2 False Positive Reduction

Table 3 presents per-category false-positive rates for the baseline and the full ExTriage framework with analyst disposition applied.

Table 3. False positive rate reduction by attack category

Attack Category	FP Rate (Baseline)	FP Rate (XAI-Triage)	Reduction (%)
DoS / DDoS	12.3%	3.1%	74.8%
Port Scan	8.7%	2.4%	72.4%
Brute Force	15.6%	4.8%	69.2%
Web Attack – XSS	21.4%	6.2%	71.0%
Infiltration	31.2%	9.8%	68.6%
Botnet	18.9%	5.5%	70.9%
Overall (Weighted)	17.1%	5.0%	70.8%

The 70.8% overall false positive reduction significantly exceeds the 28% reduction reported by Veerasamy et al. [15]. The largest drop is observed in the DoS/DDoS class (74.8%), where the features packet-per-second and flow-duration are consistently discriminative for SHAP. The smallest reduction is in Infiltration (68.6%), where there is an ambiguous overlap between legitimate administrative activity and attack patterns.

5.3 Explanation Latency

The median end-to-end explanation latency with 10,000 test alerts is 3.8 seconds for the entire framework. For the CICIDS-2018 test stream, the latency of computing SHAP was reduced from 210ms (cold) to 18ms (warm) using an LRU cache, while the cache hit rate remained 67–72%. The counterfactual generation took an average of 1.2 seconds per alert (95th percentile: 4.1 seconds). The average time for computing the priority scores was 12ms. Overall, the framework achieves a high-level latency target (<10 seconds) for 94.3% of alerts.

5.4 Human Factors Study Results

The within-subjects study showed statistically significant differences among all three conditions ($F(2,46)=41.7$, $p<0.001$, $\eta^2=0.64$). Pairwise comparisons (Bonferroni-corrected) demonstrated that Condition C (SHAP+CF) significantly outperformed Condition B (SHAP only) in

both the mean difference in triage accuracy (7.3%, p=0.003) and Brier score improvement in trust calibration (0.089, p=0.001). Both XAI conditions significantly outperformed label-only (p<0.001 for all metrics). Triage time in Condition C averaged 8.2 seconds per alert, compared to 15.4 seconds (Condition B) and 47.3 seconds (label only). This counterintuitive finding suggests counterfactual reasoning accelerates analyst decision closure for ambiguous alerts. The analysis showed no significant interaction between analyst experience and the explanation condition on accuracy. This suggests that the benefits of XAI are well distributed for all levels of experience.

5.5 Adversarial Robustness

JSMA adversarial probes applied to 500 test alerts resulted in 23 categorization evasion cases (4.6% of total) with an average L∞ perturbation of $\epsilon=0.12$. In 19 of these 23 cases, the shift in SHAP attribution was substantial (the mean SHAP rank correlation decreased by 0.41), providing analysts with a secondary anomalous signal even if the classification was not reversed. We refer to this feature explanation as an anomalous signal and call it out as a possible route for adversarially-aware XAI design.

6. DISCUSSION

6.1 Addressing the Research Questions

Answer to RQ1: Yes. ExTriage achieves F1=97.89% and AUC-ROC=0.9963, with a median explanation latency of 3.8 seconds, meeting the < 5-second goal for 94.3% of alerts. We found that the main architectural innovation enabling speed optimization with minimal latency is the stacked ensemble of TreeSHAP and LRU caching. Strong empirical evidence addresses RQ2: The 70.8% decrease in false positives and the 7.3% increase in accuracy of SHAP+CF over SHAP-only provide strong evidence that counterfactual explanations add practical value beyond feature attribution alone. The third research question is answered by the human factors study and analyst interviews. Some of the key design principles emerging include:

a. explanation brevity—SHAP waterfall plots restricted to the top seven features outperformed full-feature plots;

b. natural language rendering of counterfactuals;

c. uncertainty communication through calibrated probability intervals;

d. transparent alert queue ordering.

6.2 Theoretical Contributions

This work develops three different theoretical perspectives. First, it characterizes XAI-triage as a unique sociotechnical system design problem that requires explicit consideration of explanation integrity, operational latency, cognitive strain, and analyst trust calibration as first-class design criteria. Second, looking for explanations as anomaly signals is a fresh idea: XAI attribution instability under adversarial perturbation is an anomaly detection signal. Third, the observed decrease in triage time with increased explanations of the evidence helps advance cognitive science research on decision closure.

6.3 Practical and Policy Implications

ExTriage is designed to be modular and flexible by using its REST API, making it easy for SOC professionals to integrate it into their existing SIEM/SOAR pipelines. The output taxonomy is based on MITRE ATT&CK, so it is compatible with threat intelligence platforms such as MISP and OpenCTI. The direct implication for the workforce is the 70.8% reduction in false positives. If an SOC receives 5,000 alerts each day and the baseline false-positive rate is 17%, ExTriage lowers the false-positive rate to 250, saving around 30 analyst-hours a day.

6.4 Limitations

There are several caveats to consider. The CICIDS-2017/2018 datasets are collected in a controlled environment and might not capture enterprise network traffic in 2024-2026. The human factors study was conducted with 24 analysts from three organizations and cannot be generalized to other SOC circumstances. Adversarial robustness was evaluated in the white-box setting; results

may differ in other settings (black-box and adaptive). One organization's annotated alert pairs were used to train the RankNet priority module, which needed to be retrained on the target organization's SOC data to see the best results. The counterfactual feasibility constraints were manually coded and could be updated in the future as network protocols change.

7. CONCLUSION

This paper introduces an Explainable AI-based alert triage framework for SOP, called ExTriage, that combines ensemble classification, TreeSHAP-based explanations, counterfactual guidance, and RankNet-based priority ranking into a single, operationally deployable framework. Evaluated on CICIDS-2017, CICIDS-2018, and UNSW-NB15, ExTriage achieves macro F1=97.89% and reduces per-class false positive rates by a mean of 70.8%. In a controlled study with 24 professional SOC analysts, the full framework reduced average triage time from 47 to 8.2 seconds with statistically significant improvements in triage accuracy and trust calibration. The primary theoretical contribution is the formalization of XAI-triage as a distinct sociotechnical design challenge. The novel observation that SHAP attribution instability under adversarial perturbation constitutes an independent anomaly signal opens a productive direction for adversarially-aware XAI design.

Future research directions include:

a. Evaluation on production SOC data streams;
b. Extension of the counterfactual module to multi-step ATT&CK kill-chain reasoning;
c. Investigation of LLM-generated narrative explanations;
d. Longitudinal human factors studies examining analyst skill development;
e. Formal red-teaming of ExTriage's explanation fidelity against adaptive adversaries targeting SHAP manipulation.

ACKNOWLEDGMENT

The authors sincerely express their gratitude to JSPM University, Pune, for providing academic guidance, technical support, and institutional resources throughout the completion of this research work.

CONFLICT OF INTEREST

The authors declare that there is *no conflict* of interest regarding the publication of this paper.

REFERENCES

[1] A. Chakraborty, A. Biswas, and A. K. Khan, "Artificial Intelligence for Cybersecurity: Threats, Attacks and Mitigation," Intelligent Systems Reference Library, vol. 231, pp. 3–25, 2023, doi: 10.1007/978-3-031-12419-8_1.

[2] S. Freitas, J. Kalajdjieski, A. Gharib, and R. McCann, "AI-Driven Guided Response for Security Operation Centers with Microsoft Copilot for Security," WWW Companion 2025 - Companion Proceedings of the ACM Web Conference 2025, vol. 1, pp. 191–200, May 2025, doi: 10.1145/3701716.3715209.

[3] D. Bhamare, T. Salman, M. Samaka, A. Erbad and R. Jain, "Feasibility of Supervised Machine Learning for Cloud Security," 2016 International Conference on Information Science and Security (ICISS), Pattaya, Thailand, 2016, pp. 1-5, doi: 10.1109/ICISSEC.2016.7885853.

[4] B. Avanzi, X. Tan, G. Taylor, and B. Wong, "On the Evolution of Data Breach Reporting Patterns and Frequency in the United States: A Cross-State Analysis," North American Actuarial Journal, vol. 29, no. 4, pp. 833–864, 2025, doi: 10.1080/10920277.2025.2457491.

[5] F. Doshi-Velez and B. Kim, "Towards A Rigorous Science of Interpretable Machine Learning," Feb. 2017, Accessed: Jul. 11, 2026. [Online]. Available: https://arxiv.org/pdf/1702.08608.

[6] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," Artif. Intell., vol. 267, pp. 1–38, Feb. 2019, doi: 10.1016/J.ARTINT.2018.07.007.

[7] S. M. Lundberg, P. G. Allen, and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," Adv. Neural Inf. Process. Syst., vol. 30, 2017, Accessed: Jul. 11, 2026. [Online]. Available: https://github.com/slundberg/shap.

[8] M. T. Ribeiro, S. Singh, and C. Guestrin, " 'Why should i trust you?' Explaining the predictions of any classifier," Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, vol. 13-17-August-2016, pp. 1135–1144, Aug. 2016, doi: 10.1145/2939672.2939778.

[9] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," ACM Comput. Surv., vol. 51, no. 5, Sep. 2019, doi: 10.1145/3236009.

[10] A. Nadeem et al., "SoK: Explainable Machine Learning for Computer Security Applications," 2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P), Delft, Netherlands, 2023, pp. 221-240, doi: 10.1109/EuroSP57164.2023.00022.

[11] M. Keshk, N. Koroniotis, N. Pham, N. Moustafa, B. Turnbull, and A. Y. Zomaya, "An explainable deep learning-enabled intrusion detection framework in IoT networks," Inf. Sci. (N. Y)., vol. 639, p. 119000, Aug. 2023, doi: 10.1016/J.INS.2023.119000.

[12] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," 2010 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 2010, pp. 305-316, doi: 10.1109/SP.2010.25.

[13] K. Julisch, "Clustering intrusion detection alarms to support root cause analysis," ACM Transactions on Information and System Security, vol. 6, no. 4, pp. 443–471, Nov. 2003, doi: 10.1145/950191.950192.

[14] J. Tilbury and S. V. Flowerday, "The vigilance paradox: automation reliance inside the modern SOC," Information and Computer Security, pp. 1–24, 2025, doi: 10.1108/ICS-08-2025-0318/1334832.

[15] L. Zhang, Z. Shao, B. Chen and J. Benitez, "Unraveling Generative AI Adoption in Enterprise Digital Platforms: The Effect of Institutional Pressures and the Moderating Role of Internal and External Environments," in IEEE Transactions on Engineering Management, vol. 72, pp. 335-348, 2025, doi: 10.1109/TEM.2024.3513773.

[16] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," 2018, doi: 10.5220/0006639801080116.

[17] K. Alrawashdeh and C. Purdy, "Toward an Online Anomaly Intrusion Detection System Based on Deep Learning," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 2016, pp. 195-200, doi: 10.1109/ICMLA.2016.0040.

[18] M. Shoaib et al., "A deep learning-assisted visual attention mechanism for anomaly detection in videos," Multimedia Tools and Applications 2023 83:29, vol. 83, no. 29, pp. 73363–73390, Dec. 2023, doi: 10.1007/S11042-023-17770-Z.

[19] N. Khan, K. Ahmad, A. Al Tamimi, M. M. Alani, A. Bermak, and I. Khalil, "Explainable AI-Based Intrusion Detection Systems for Industry 5.0 and Adversarial XAI: A Systematic Review," Information (Switzerland), vol. 16, no. 12, p. 1036, Dec. 2025, doi: 10.3390/INFO16121036.

[20] P. Bajpayi, S. Mathur and M. S. Gaur, "Dual Stream Cross-Attentional Driven Adaptive Anomaly Detection for Zero-Day Attacks in IoMT Networks," 2026 13th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2026, pp. 1-6, doi: 10.23919/INDIACom70271.2026.11526185.

[21] S. M. Lundberg et al., "From local explanations to global understanding with explainable AI for trees," Nature Machine Intelligence 2020 2:1, vol. 2, no. 1, pp. 56–67, Jan. 2020, doi: 10.1038/s42256-019-0138-9.

[22] D. Slack, S. Hilgard, E. Jia, S. Singh, and H. Lakkaraju, "Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods," AIES 2020 - Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, pp. 180–186, Feb. 2020, doi: 10.1145/3375627.3375830.

[23] A. Warnecke, D. Arp, C. Wressnegger and K. Rieck, "Evaluating Explanation Methods for Deep Learning in Security," 2020 IEEE European Symposium on Security and Privacy (EuroS&P), Genoa, Italy, 2020, pp. 158-174, doi: 10.1109/EuroSP48549.2020.00018.

[24] S. Verma, V. Boonsanong, M. Hoang, K. Hines, J. Dickerson, and C. Shah, "Counterfactual Explanations and Algorithmic Recourses for Machine Learning: A Review," ACM Comput. Surv., vol. 56, no. 12, Oct. 2024, doi: 10.1145/3677119.

[25] L. Allodi and F. Massacci, "Comparing vulnerability severity and exploits using case-control studies," ACM Transactions on Information and System Security, vol. 17, no. 1, Aug. 2014, doi: 10.1145/2630069.

[26] J. B. Awotunde and S. Misra, "Feature Extraction and Artificial Intelligence-Based Intrusion Detection Model for a Secure Internet of Things Networks," Lecture Notes on Data Engineering and Communications Technologies, vol. 109, pp. 21–44, 2022, doi: 10.1007/978-3-030-93453-8_2.

[27] Y. Jiang, N. Oo, Q. Meng, H. W. Lim, and B. Sikdar, "VulRG: Multi-Level Explainable Vulnerability Patch Ranking for Complex Systems Using Graphs," vol. 1, Feb. 2025, doi: https://doi.org/10.48550/arXiv.2502.11143.

[28] C. Burges et al., "Learning to rank using gradient descent," ICML 2005 - Proceedings of the 22nd International Conference on Machine Learning, pp. 89–96, 2005, doi: 10.1145/1102351.1102363.

[29] J. Dodge, Q. V. Liao, Y. Zhang, R. K. E. Bellamy, and C. Dugan, "Explaining models," IUI '19: Proceedings of the 24th International Conference on Intelligent User Interfaces, pp. 275–285, Feb. 2019, doi: 10.1145/3301275.3302310.

[30] G. Bansal, B. Nushi, E. Kamar, D. S. Weld, W. S. Lasecki, and E. Horvitz, "Updates in Human-AI Teams: Understanding and Addressing the Performance/Compatibility Trade-off," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 2429–2437, Jul. 2019, doi: 10.1609/AAAI.V33I01.33012429.

[31] R. Parasuraman and V. Riley, "Humans and Automation: Use, Misuse, Disuse, Abuse," Hum. Factors, vol. 39, no. 2, pp. 230–253, 1997, doi: 10.1518/001872097778543886.

[32] C.-K. Yeh, C.-Y. Hsieh, A. Suggala, D. I. Inouye, and P. K. Ravikumar, "On the (In)fidelity and Sensitivity of Explanations," Adv. Neural Inf. Process. Syst., vol. 32, 2019, Accessed: Jul. 11, 2026. [Online]. Available: https://github.com/chihkuanyeh/saliency_evaluation.

[33] T. Laugel, M.-J. Lesot, C. Marsala, X. Renard, and M. Detyniecki, "Inverse Classification for Comparison-based Interpretability in Machine Learning," Dec. 2017, Accessed: Jul. 11, 2026. [Online]. Available: https://arxiv.org/pdf/1712.08443.

[34] Z. Lin, Y. Shi, and Z. Xue, "IDSGAN: Generative Adversarial Networks for Attack Generation Against Intrusion Detection," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 13282 LNAI, pp. 79–91, 2022, doi: 10.1007/978-3-031-05981-0_7.

[35] Caroline E. Zsambok, Naturalistic Decision Making. Psychology Press, 2014. doi: 10.4324/9781315806129.

[36] S. Mane and D. Rao, "Explaining Network Intrusion Detection System Using Explainable AI Framework," Mar. 2021, Accessed: Jul. 11, 2026. [Online]. Available: https://arxiv.org/pdf/2103.07110.

[37] P. Vibhandik, S. Sawant, A. Joshi, and R. Bidwe, "A systematic literature review on forest fire susceptibility mapping using geo-spatial technology and future research directions," Discover Artificial Intelligence 2026 6:1, vol. 6, no. 1, pp. 396-, Mar. 2026, doi: 10.1007/S44163-026-00921-0.

[38] B. Shehu and I. Jordanov, "Machine Learning for Fault Detection in Wireless Sensor Networks: A Survey," in IEEE Internet of Things Journal, vol. 13, no. 10, pp. 20437-20451, 15 May15, 2026, doi: 10.1109/JIOT.2026.3669436.

[39] N. Khan, K. Ahmad, A. Al Tamimi, M. M. Alani, A. Bermak, and I. Khalil, "Explainable AI-based Intrusion Detection System for Industry 5.0: An Overview of the Literature, associated Challenges, the existing Solutions, and Potential Research Directions," Jul. 2024, Accessed: Jul. 11, 2026. [Online]. Available: https://arxiv.org/pdf/2408.03335.

[40] A. Zaslavsky, C. Perera, and D. Georgakopoulos, "Sensing as a Service and Big Data," Jan. 2013, Accessed: Jul. 11, 2026. [Online]. Available: https://arxiv.org/pdf/1301.0159.