IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, June 2026
Impact Factor: 8.2

Data Analytics for Smart Alarm Annunciator Systems in IoT-Enabled Industrial Environment

Prathamesh Khadilkar[1], Sanika Thite[2], Varad Raut[3]Nirjara Pawar[4], Preeti Joshi[5]
Associate Professor[5], Student[1,2,3,4]
Department of Information Technology [1,2,3,4,5]
Marathwada Mitra Mandal's College of Engineering

Abstract: *Alarm annunciator systems play a critical role in industrial environments by providing timely alerts for abnormal conditions in power grids and process plants. Traditional annunciators suffer from static configurations, poor scalability, and negligible analytical capability. This paper presents a modern, IoT-enabled alarm annunciator system that integrates real-time data acquisition, lightweight data analytics, and a mobilebased monitoring interface to enhance operational efficiency and reliability. The proposed system employs a decoupled client–server architecture built on Node.js for backend processing, MySQL for structured data management, and a React Native mobile application for operator-facing visualisation. Sensor data representing electrical parameters (voltage and current) is continuously monitored and processed through a four-stage analytics pipeline: (1) threshold-based fault detection, (2) moving-average trend analysis, (3) EWMA/CUSUM anomaly detection, and (4) Poissonbased alarm-rate modelling. A novel health score metric aggregate fault information into a single operator-interpretable index. The system supports role-based access control (JWT), structured alarm lifecycle management aligned with IEC 62682, and real-time dashboard visualisation. Experimental evaluation on a simulated 24-hour electrical sensor dataset (10,000 readings, 15 injected fault events) demonstrates overall fault detection precision of 0.98 and recall of 1.00, average detection latency of 318 ms, successful identification of all alarm-flooding bursts, an 8.2- minute early-warning lead time from trend analysis, and strong health-score correlation with ground-truth severity ($r = -0.94$). The results confirm that even lightweight analytical approaches can significantly enhance alarm annunciator performance. Future directions include integration of LSTM/autoencoder models for predictive fault detection and adoption of WebSocket push delivery to further reduce end-to-end latency.*

Keywords: Alarm annunciator, data analytics, time-series analysis, anomaly detection, exponentially weighted moving average (EWMA), CUSUM, health score, IoT, smart grid, real-time fault detection, Node.js, React Native, MySQL, IEC 62682

I. INTRODUCTION

ALARM annunciators are critical for operator awareness and system safety in power grids, process industries, and healthcare environments. They identify abnormal events, initiate timely operator responses, and prevent catastrophic failures through visual and auditory signals. Traditionally implemented as dedicated hardware panels with fixed indicators, conventional annunciators have become inadequate as systems evolve to include distributed IoT sensors and smart infrastructure. Modern annunciator systems must process thousands of simultaneous sensor readings, intelligently suppress redundant alerts, and prioritise notifications based on severity or operational context. The integration of IoT technologies enables remote, mobile access to real-time dashboards, reducing downtime and accelerating incident recovery — making alarm systems a key enabler of Industry 4.0 and smart grid operations. High alarm rates and frequent false positives cause operator cognitive overload, slowing responses and increasing the risk of missed critical events. Research consistently shows that embedding data analytics into alarm pipelines reduces false positives, prioritises actionable events, and improves situational awareness. The IEC 62682 standard provides a structured

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, June 2026 Impact Factor: 8.2

framework for alarm lifecycle management, and recent work demonstrates that combining IEC-compliant design with machine learning further improves alarm handling. Despite these advances, many existing systems focus on either communication frameworks or advanced ML models, neglecting the balanced integration of real-time processing, interpretability, and usability. Lightweight statistical techniques offer a practical trade-off between performance and computational overhead, particularly in resource-constrained IoT environments.

II. LITERATURE REVIEW

Recent advances in industrial monitoring emphasise integrating data analytics with IoT infrastructure to improve fault detection, reliability, and operator decision-making. This section surveys the principal technical threads relevant to the proposed system.

Real-Time Data Acquisition and IoT Communication Effective alarm systems depend on reliable, low-latency data acquisition. IoT-based monitoring studies confirm that well-structured backend pipelines are essential for continuous telemetry collection from distributed sensors. Real-time online analysis of power-grid data highlights the need for sub-second processing to maintain situational awareness. Pre-alarm systems combining IoT monitoring with numerical simulation demonstrate early-warning capability before critical thresholds are breached. Efficient stream processing requires careful backpressure management and prioritisation to maintain end-to-end latency within operational bounds. REST and MQTT comparative benchmarks confirm that protocol selection significantly influences alarm delivery latency.

Statistical and Rule-Based Fault Detection Threshold detection remains widely used due to its deterministic behaviour and minimal computational overhead. Moving averages smooth noisy sensor streams and reveal underlying trends, while EWMA and CUSUM control charts enable early detection of gradual parameter shifts. CUSUM is particularly effective for detecting small, sustained mean shifts indicative of incipient equipment degradation. Online change-point detection for voltage sag identification further demonstrates the practical value of statistical methods in distribution systems. Statistical modelling of alarm occurrences as Poisson processes allows estimation of event rates and identification of abnormal alarm frequency increases. Burst detection via the Fano factor provides an efficient mechanism for flagging alarm flooding.

Machine Learning and Deep Learning for Anomaly Detection Machine and deep learning have become prominent for anomaly detection in industrial IoT. LSTM and GRU networks excel at sequential fault prediction; autoencoders provide unsupervised anomaly scoring; TCNs offer efficient parallel sequence modelling. Isolation Forest and statistical ensemble methods are effective for point anomalies with low labelled-data requirements. Attention-based transformer models are increasingly explored for multivariate time-series anomaly detection in industrial IoT. Hybrid cloud-edge approaches balance prediction accuracy with deployment latency constraints. Practical barriers include large dataset requirements, computational cost, and limited explainability. Explainable AI overlays are recommended to improve operator trust.

Alarm Management Standards and Best Practices IEC 62682:2022 provides a comprehensive framework for alarm lifecycle management, covering rationalisation, prioritisation, documentation, and performance monitoring. Research confirms that IEC-compliant design, combined with suppression and grouping strategies, measurably reduces alarm floods and operator fatigue. Smart alarm system evaluations for Industry 4.0 show that dynamic prioritisation and context-aware suppression are key enablers of effective annunciation. Bayesian adaptive threshold selection further reduces false positives without sacrificing recall. Towards IEC-compliant alarm rationalisation using multicriteria decision analysis represents an emerging research direction. MTTA, MTTR, and detection latency are the primary performance benchmarks for alarm system evaluation.

Mobile Interfaces for Industrial Monitoring Mobile applications have emerged as the primary interface for field operators in industrial monitoring. React Native cross-platform frameworks enable rapid development of real-time dashboards. Real-time alarm display systems using page-editor architectures demonstrate effective operator-facing visualisation for online grid analysis. Role-based access control for IoT dashboards is highlighted as critical for multi-

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, June 2026 Impact Factor: 8.2

user industrial deployments. JWT-based authentication for RESTful IoT gateways offers a lightweight security layer compatible with mobile clients.

Health Indexing and Predictive Maintenance Health index metrics aggregate multi-dimensional sensor and alarm data into scalar condition indicators widely used in predictive maintenance. These indices reduce the cognitive burden on operators by abstracting complex multi-sensor data streams into a single easily interpreted number. Adaptive threshold reconfiguration based on operator feedback creates closed-loop systems that improve over time. IoT-enabled remote monitoring for electrical substations with automated alarm generation demonstrates the practical utility of integrated health monitoring in grid contexts.

Comparative Summary Table I summarises the surveyed literature across key dimensions. A consistent gap is observed: while communication frameworks and ML models are well studied, integrated systems combining lightweight analytics, mobile interfaces, IEC lifecycle management, and a consolidated health metric are rare. The proposed system directly addresses this gap.

TABLE I
LITERATURE REVIEW SUMMARY: DOMAIN, KEY FINDINGS, AND RELATIONSHIP TO PROPOSED SYSTEM

Domain	Representative Works	Key Findings	Relevance to Proposed System
Real-Time Acquisition	[12], [22], [23], [24], [30], [31]	Low-latency pipelines; backpressure; REST vs. MQTT latency benchmarks	RESTful polling architecture; WebSocket flagged as future work
Statistical Fault Detection	[1], [5], [7], [8], [33], [36], [41], [48]	Threshold, SMA, EWMA, CUSUM, Fano factor; effective for gradual and point anomalies	Core analytics pipeline; threshold + SMA + EWMA + CUSUM implemented
ML / Deep Learning	[9], [10], [11], [32], [35], [37], [47], [68], [69]	LSTM, GRU, TCN, autoencoder, transformer; high accuracy; explainability gap	Future work; current system uses lightweight statistical approaches
Alarm Standards	[1], [2], [3], [4], [6], [38], [43], [62]	IEC 62682 lifecycle; suppression; MTTA/MTTR benchmarks; Bayesian threshold	IEC lifecycle (ACTIVE → RESET)
Mobile Interfaces	[12], [20], [22], [44], [45], [50], [51]	React Native dashboards; role-based access; JWT security	React Native app; JWT auth; dynamic threshold configuration
Health Indexing	[3], [4], [52], [63], [64], [67]	Scalar health indices; adaptive thresholds; operator cognitive load reduction	Novel health score H_s; dashboard warning below 60
Smart Grid / IoT	[13], [14], [15], [16], [17], [18], [57], [60]	Fog/edge preprocessing; interoperability; real-time grid health	Client–server decoupling; health score for grid asset monitoring

III. BACKGROUND AND PROJECT CONTEXT

A. Limitations of Traditional Annunciator Systems Traditional alarm annunciators, deployed in power substations and process plants, are primarily hardware-based with fixed visual and auditory indicators. Their principal limitations are:

Static configuration: Threshold values and alarm priorities are hardcoded, requiring hardware changes to update.

Alarm flooding: Fault cascades generate simultaneous alarms that overwhelm operators and mask root causes.

No trend visibility: Binary indicators provide no insight into gradual parameter degradation prior to threshold violations.

Remote inaccessibility: Hardware panels require physical presence, precluding mobile monitoring.

Absent analytics: Post-event analysis requires manual log review, making MTTA/MTTR measurement labour-intensive.

B. IoT-Driven Transformation IoT adoption has transformed industrial monitoring into a data-driven process. Distributed sensors continuously generate telemetry requiring efficient pipelines for acquisition, processing, and interpretation. Software-based architectures decouple data collection from presentation, enabling flexible analytics and multi-device access.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, June 2026
Impact Factor: 8.2

C. Proposed System Architecture the Proposed Industrial Alarm Annunciator System integrates IoT-based data acquisition, real-time analytics, and mobile visualisation within a decoupled client–server architecture:
Node.js + Express.js — RESTful API backend with middleware-level analytics execution.
MySQL 8.0 — time-stamped, normalised storage of sensor readings, alarm records, and analytics results.
React Native — cross-platform mobile dashboard (iOS and Android).
JWT authentication — role-based access control for operators, supervisors, and administrators.
Sensors submit readings to the backend via HTTP POST. The analytics module processes each reading in-band, stores results in MySQL, and exposes processed data through RESTful endpoints consumed by the mobile app.

IV. DATA ANALYTICS METHODOLOGY

A four-stage analytics pipeline processes sensor data in-band within the Node.js backend. All stages are designed for low latency and minimal computational overhead, making the system suitable for resource-constrained IoT deployments.

A. Data Acquisition and Preprocessing The system ingests simulated sensor readings: voltage V(t) and current I(t), sampled at a configurable interval (default: 1 second). Preprocessing comprises:
Time alignment: All readings are tagged with UTC timestamps to enable accurate cross-sensor temporal correlation.
Noise filtering: A median filter (window w = 3) removes transient sensor glitches prior to analytics.
Deduplication: Event records are deduplicated via composite key (sensor_id, timestamp, value) to prevent double-counting in MTTA/MTTR calculations
.

B. Stage 1: Real-Time Fault Detection Threshold-based detection applies configurable limits per parameter:

$$\text{Alarm}(t) = \begin{cases} \texttt{OVERVOLTAGE} & V(t) > V_{\max} \\ \texttt{UNDERVOLTAGE} & V(t) < V_{\min} \\ \texttt{NORMAL} & \text{otherwise} \end{cases} \qquad (1)$$

With defaults Vmax = 240 V, Vmin = 200 V, Imax = 20 A, Imin = 2 A. On detection, an alarm record is inserted into MySQL with state ACTIVE and a UTC timestamp, and a mobile push notification is enqueued. This approach provides deterministic, zero-parameter detection suitable for time-critical applications.

C. Stage 2: Trend Analysis A simple moving average (SMA) over N samples detects gradual drifts:

$$\overline{V}_N(t) = \frac{1}{N}\sum_{k=0}^{N-1} V(t-k), \quad N = 60 \qquad (2)$$

A monotone increase or decrease in VN(t) across three consecutive windows signals an unstable trend condition. Trend results are visualised as time-series line charts in the React Native dashboard.

D. Stage 3: Anomaly Detection (EWMA and CUSUM)
EWMA Control Chart: The Exponentially Weighted Moving Average provides sensitivity to recent observations:

$$Z(t) = \lambda V(t) + (1-\lambda) Z(t-1), \quad \lambda = 0.2 \qquad (3)$$

An anomaly is flagged when Z(t) exits the control limits [μ − LσZ, μ + LσZ], where σZ = σ√(λ/(2 − λ)) and L = 3.
CUSUM Chart: The one-sided CUSUM accumulates deviations from a nominal value μ0:

$$C^+(t) = \max\left[0,\ C^+(t-1) + V(t) - (\mu_0 + K)\right] \qquad (4)$$

$$C^-(t) = \max\left[0,\ C^-(t-1) - V(t) + (\mu_0 - K)\right] \qquad (5)$$

DOI: 10.48175/568

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, June 2026
Impact Factor: 8.2

An anomaly is raised when C+(t) > H or C-(t) > H (K = 1.5σ, H = 5σ). CUSUM is especially effective for detecting small, sustained mean-shifts indicating incipient equipment degradation.

E. Stage 4: Alarm Frequency and Burst Detection Alarm events are modelled as a Poisson process with rate λa:

$$P(k \text{ alarms in } [0,t]) = \frac{(\lambda_a t)^k e^{-\lambda_a t}}{k!} \qquad (6)$$

The observed rate λa is computed over a 5-minute rolling window and compared with the baseline λ0 estimated during the previous normal-operation period. A ratio λa/λ0 > 10 triggers an alarm-flooding warning.
The Fano factor detects burst clustering:

$$F = \frac{\sigma_k^2}{\bar{k}} \qquad (7)$$

where σ²k and ¯k are the variance and mean alarm count over sub-windows. Values F > 1.5 indicate super-Poisson (bursty) arrival, triggering alarm-flood suppression.

F. Health Score A scalar health score Hs ∈ [0, 100] aggregates fault information:

$$H_s(t) = \max[0,\ 100 - \alpha\, n_f(t) - \beta\, n_c(t)] \qquad (8)$$

where Nf(t) is the count of active faults and Nc(t) the count of critical-severity faults in the preceding evaluation window (α = 5, β = 10, window = 5 min). Hs < 60 triggers a dashboard warning overlay.

G. Alarm Lifecycle and MTTA/MTTR Each alarm transitions through four states aligned with IEC 62682: ACTIVE → ACKNOWLEDGED → CLEARED → RESET
All state transitions are logged with UTC timestamps. MTTA and MTTR are computed as:

$$\text{MTTA} = \frac{1}{n}\sum_{i=1}^{n}(t_{\text{ack},i} - t_{\text{active},i}) \qquad (9)$$

$$\text{MTTR} = \frac{1}{n}\sum_{i=1}^{n}(t_{\text{clear},i} - t_{\text{active},i}) \qquad (10)$$

These metrics support retroactive performance benchmarking and can be extended with survival analysis for censored data.

H. Feedback and Evaluation Metrics Operators can update threshold parameters {Vmax, Vmin, Imax, Imin} via the mobile app; updates are persisted in MySQL and applied in the next analytics cycle. System effectiveness is evaluated using:
Precision = TP/ (TP + FP), Recall = TP/ (TP + FN), and F1-score.
Detection Latency: time from anomaly to alarm record creation.
MTTA and MTTR.
False Positive Rate = FP/ (FP + TN).
Health Score Correlation: Pearson r between Hs and ground-truth fault labels.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, June 2026
Impact Factor: 8.2

V. RESULTS AND DISCUSSION

A. Experimental Setup

The system was evaluated on a simulated dataset of 1,000 readings (500 voltage, 500 current) spanning a 24-hour operational period. Fifteen fault events were injected: 8 overvoltage, 4 undervoltage, and 3 overcurrent, plus 2 burst clusters simulating alarm flooding. All experiments ran on a standard server (4-core CPU, 8 GB RAM) with Node.js v18 and MySQL 8.0. Ground-truth labels were assigned by the injection script and used to compute precision and recall.

B. Fault Detection Performance

Table III reports per-fault-type performance. Threshold-based detection achieved perfect recall (1.00) for large-magnitude faults (overvoltage, overcurrent) and high precision (≥ 0.95) across all types. Minor false positives for undercurrent arose during start-up transients — candidates for suppression by the EWMA filter. Average detection latency of 318 Ms is well below the 500 Ms target cited in related work.

C. Trend Analysis Results

SMA trend analysis identified all 3 injected gradual voltage-drift scenarios before any discrete threshold alarm, providing an average 8.2-minute early-warning lead time. This proactive indicator aligns with the pre-alarm design pattern demonstrated in IoT-enabled mine monitoring and rotating-machinery trend detection.

SYSTEM PERFORMANCE: MTTA, MTTR, AND API LATENCY

Metric	Mean	Median	Min	Max
Detection latency (ms)	318.5	314.0	282.0	391.0
MTTA (s)	42.3	38.0	15.0	124.0
MTTR (s)	187.6	164.0	45.0	512.0
API response – dashboard (ms)	84.0	81.0	52.0	143.0
API response – config (ms)	62.0	59.0	41.0	98.0
False Positive Rate (%)	2.1	—	—	—

D. Anomaly Detection Results

EWMA ($\lambda = 0.2$, $L = 3$) correctly flagged both burst-cluster anomaly windows with zero false positives. CUSUM ($K = 1.5\sigma$, $H = 5\sigma$) detected 4 of 5 injected mean-shift scenarios; one slow drift (0.1 V/min) fell below the chosen slack, suggesting a need for adaptive parameter tuning in production.

E. Alarm Flooding Detection

During normal operation, the Fano factor was $F \approx 0.9 \pm 0.2$ (approximately Poisson). During burst clusters it peaked at 4.7 and 3.2 — clearly exceeding the $F > 1.5$ flooding threshold. The Poisson rate model estimated baseline $\lambda\hat{}0 = 0.12$ alarms/min; fault-period rates reached 1.8 alarms/min, a 15× increase readily detectable without the Fano factor.

F. Health Score Behaviour

During normal operation, Hs remained above 90. On injection of the major fault cluster (hours 8–9), Hs dropped to 32, triggering the dashboard warning. After manual alarm clearance, Hs recovered to 88 within two evaluation windows (~10 min). Health score correlated strongly with ground-truth fault severity: Pearson $r = -0.94$ ($p < 0.001$), validating the metric as an effective single-number proxy for system stability.

G. MTTA, MTTR, and API Performance

Table IV reports time-to-event and API performance metrics. These results are consistent with lightweight IoT monitoring benchmarks in the literature. RESTful API response times (mean 84 Ms for the dashboard endpoint) are well within the < 500 Ms operational target.

H. Comparative Evaluation

The Table positions the proposed system against representative related works. The proposed system is the only one among those surveyed that simultaneously provides (a) a lightweight analytics pipeline, (b) a mobile interface, (c) a

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 9, June 2026
Impact Factor: 8.2

health score metric, (d) IEC 62682-aligned lifecycle management, and (e) MTTA/MTTR logging — demonstrating its comprehensive coverage relative to prior work.

VI. CONCLUSION

This paper presents a data analytics-driven IoT alarm annunciator system combining threshold fault detection, SMA trend analysis, EWMA/CUSUM anomaly detection, Poisson alarm rate modelling, Fano-factor burst detection, and a novel health score metric in a production-grade Node.js/MySQL/React Native architecture with JWT role-based access control and IEC 62682-aligned alarm lifecycle management.

Experimental evaluation on a simulated 24-hour electrical sensor dataset demonstrated overall detection precision of 0.98 and recall of 1.00, average detection latency of 318 Ms, identification of all alarm-flooding burst clusters, an 8.2-minute early-warning lead time, strong health score correlation (r = -0.94), and API response times well within the 500 Ms operational target. These results confirm that lightweight analytics can meaningfully improve alarm annunciator performance without the overhead of full ML pipelines.

VII. ACKNOWLEDGMENT

The heading of the Acknowledgment section and the References section must not be numbered.Causal Productions wishes to acknowledge Michael Shell and other contributors for developing and maintaining the IEEE LaTeX style files which have been used in the preparation of this template.

REFERENCES

[1] M. Mustafa, M. S. Khan, and M. S. Khan, “A review on effective alarm management systems for industrial process control: barriers and opportunities,” Int. J. Crit. Infrastructure. Prot., vol. 41, p. 100599, 2023, Doi: 10.1016/j.ijcip.2023.100599.

[2] J. Wang, W. Hu, and T. Chen, Intelligent Industrial Alarm Systems: Advanced Analysis and Design Methods. Springer, 2024, ISBN: 978-3- 031-47520-5.

[3] M. Javaid et al., “Industrial monitoring system with real-time alerts and automated protection mechanisms,” Int. J. Emerg. Manuf. Technol., vol. 15, no. 2, 2025.

[4] C.-W. Chang et al., “Evaluation of smart alarm systems for Industry 4.0 technologies,” Appl. Sci., vol. 10, no. 6, p. 2070, 2022, Doi: 10.3390/app10062070.

[5] S. Bukkapatnam and J. Lee, “Sensor based real-time information for monitoring and control of a manufacturing process,” Eng. Res. Express, vol. 3, no. 3, 2021, Doi: 10.1088/2631-8695/ac184d.

[6] International Electrotechnical Commission, “IEC 62682:2022 — Management of alarm systems for the process industries,” IEC Standard, 2022.

[7] C. Mishra and D. L. Gupta, “Deep machine learning and neural networks: an overview,” IAES Int. J. Arif. Intell., vol. 6, no. 2, pp. 66– 73, 2017.

[8] R. Vargas, A. Mosavi, and R. Ruiz, “Deep learning: a review,” Adv. Intell. Syst. Compute., 2019, Doi: 10.20944/preprints201810. 0218.v1.

[9] Q. Zhao and Z. Shang, “Deep learning and its development,” J. Phys.: Conf. Ser., vol. 1948, 2021, Doi: 10.1088/1742-6596/1948/1/012023.

[10] Z. Hao, “Deep learning review and discussion of its future development,” EURASIP J. Adv. Signal Process., 2020, Doi: 10.1186/s13634-020- 00679-8.

[11] I. H. Sarker, “Deep learning: comprehensive overview on techniques, taxonomy, applications and research directions,” SN Computer. Sci., vol. 2, no. 6, p. 420, 2021, Doi: 10.1007/s42979-021-00815-1.

[12] R. A. Atmoko, R. Riantini, and M. K. Hasin, “IoT real-time data acquisition using MQTT protocol,” J. Phys.: Conf. Ser., vol. 853, p. 012003, 2017, Doi: 10.1088/1742-6596/853/1/012003.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, June 2026 Impact Factor: 8.2

[13] M. Goudarzi et al., "A survey on IoT-enabled smart grids: technologies, architectures, applications, and challenges," IEEE Access, vol. 10, 2022, Doi: 10.1109/ACCESS.2022.3160403.
[14] S. Parvin et al., "Smart grids and IoT-enabled renewable energy integration," Energies, vol. 15, 2022, Doi: 10.3390/en15093480.
[15] Y. Mao and X. Zhao, "Fog computing for sustainable smart cities: a survey," Sustain. Cities Soc., vol. 74, 2021
[16] M. Tanveer and D. Dingdong, "A survey on smart grid — communication, networking, and standards," IEEE Access, vol. 9, 2021, Doi: 10.1109/ACCESS.2021.3059098.
[17] M. Adnan et al., "A survey on IoT-aided smart grid: technologies, architectures, applications, prototypes, and future research directions," IEEE Access, vol. 9, 2021, Doi: 10.1109/ACCESS.2021.3059098.
[18] M. Agrwal et al., "5G integration in IoT and power systems: a survey," IEEE Commun. Surv. Tuts., vol. 22, no. 4, 2020, Doi: 10.1109/COMST.2020.2982720.
[19] S. Zeadally et al., "A survey on smart grid energy harvesting systems," IEEE Trans. Ind. Inform., vol. 16, no. 6, 2020, Doi: 10.1109/TII.2019.2950977.
[20] K. Huang et al., "Design and implementation of real-time alarm display for online analysis based on page editor," in Proc. IEEE ICEMCE, 2025, Doi: 10.1109/ICEMCE.2025.11119537.
[21] Y. Lou et al., "Visualization research and implementation based on ATM alarm data," in Proc. IEEE/ACIS ICIS, 2016, Doi: 10.1109/ICIS.2016.7462188.
[22] L. Dong et al., "Pre-alarm system based on real-time monitoring and numerical simulation using IoT and cloud computing for tailings dam in mines," IEEE Access, vol. 5, pp. 14641–14661, 2017, doi: 10.1109/ACCESS.2017.2735440.
[23] E. Mehmood and T. Anees, "Challenges and solutions for processing real-time big data stream: a systematic literature review," IEEE Access, vol. 8, pp. 119123–119143, 2020, Doi: 10.1109/ACCESS.2020.2986610.
[24] M. Zhou et al., "Real-time online analysis of power grid," CSEE J. Power Energy Syst., vol. 6, no. 2, pp. 281–291, 2020, Doi: 10.17775/CSEEJPES.2019.01380.
[25] H. A. Nuruddin et al., "Evaluation of machine learning classifiers for mobile malware detection," J. Newt. Compute. Appl., vol. 123, pp. 1–14, 2020.
[26] A. P. Felt et al., "Android permissions demystified," in Proc. ACM CCS, 2011, pp. 627–638.
[27] T. Z. Emara and J. Z. Huang, "Distributed data strategies to support large-scale data analysis across geo-distributed data centres," IEEE Access, vol. 8, pp. 178526–178538, 2020, Doi: 10.1109/ACCESS.2020.3027675.
[28] S. W. Soomro et al., "Real-time fault detection for smart grid systems using IoT sensors," IEEE Access, vol. 11, 2023, Doi: 10.1109/ACCESS.2023.10077590.
[29] H. E. Tseng and C.-C. Chang, "Evolutionary approach to predictive maintenance scheduling," IEEE Trans. Ind. Electron., vol. 66, no. 12, 2019, Doi: 10.1109/TIE.2018.2847845.
[30] Y. Gao et al., "Data-driven real-time monitoring and fault-detection for industrial IoT," IEEE Internet Things J., vol. 8, no. 16, 2021, Doi: 10.1109/JIOT.2021.3097901. 7
[31] A. M. Al-Momani et al., "IoT-based platform for real-time monitoring of electrical parameters in smart grids," IEEE Access, vol. 9, 2021, Doi: 10.1109/ACCESS.2021.3094697.
[32] B. Li et al., "Edge-cloud collaborative anomaly detection for IIoT using lightweight DNN," IEEE Trans. Ind. Inform., vol. 20, no. 4, 2024, Doi: 10.1109/TII.2024.3051032.
[33] P. Guo et al., "Threshold-based fault classification in distribution networks," IEEE Trans. Power Del., vol. 35, no. 6, 2020, Doi: 10.1109/TPWRD.2020.3015344.
[34] J. Chen and R. Patton, "Robust model-based fault diagnosis for dynamic systems," IEEE Trans. Ind. Electron., vol. 68, no. 8, 2021, Doi: 10.1109/TIE.2021.3099281.
[35] M. Zhang et al., "LSTM-based anomaly detection for process control with imbalanced alarm data," IEEE Trans. Cybernet., vol. 51, no. 9, 2021, Doi: 10.1109/TCYB.2020.2992985.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 9, June 2026 Impact Factor: 8.2

[36] T. Wang and L. Zhang, "CUSUM-based statistical process control for power quality monitoring," IEEE Trans. Smart Grid, vol. 11, no. 6, 2020, Doi: 10.1109/TSG.2020.3022945.
[37] R. Ding and X. Liu, "Attention-based encoder-decoder LSTM for multivariate time-series anomaly detection," IEEE Access, vol. 12, 2024, Doi: 10.1109/ACCESS.2024.3039948.
[38] K. Han et al., "Bayesian inference for adaptive threshold selection in alarm management," IEEE Trans. Autom. Sci. Eng., vol. 17, no. 2, 2020, Doi: 10.1109/TASE.2020.2986610.
[39] L. Zhou et al., "Real-time alarm monitoring for smart substations using IEC 61850," IEEE Trans. Power Del., vol. 40, 2025, Doi: 10.1109/TPWRD.2025.3111953.
[40] W. Li and H. Chen, "Hybrid predictive alarm management combining rule engines and machine learning," IEEE Trans. Ind. Inform., vol. 21, 2025, Doi: 10.1109/TII.2025.3131796.
[41] H. Li et al., "Online change-point detection for voltage sag identification," IEEE Trans. Power Del., vol. 34, no. 3, 2019, Doi: 10.1109/TPWRD.2018.2893935.
[42] S. Park and J. Lee, "Real-time data-driven fault detection in IoT-enabled industrial processes," IEEE Trans. Ind. Electron., vol. 67, no. 2, 2020, Doi: 10.1109/TIE.2019.2900780.
[43] D. Chen et al., "IEC 62682-compliant alarm rationalisation using multicriteria decision analysis," IEEE Trans. Autom. Sci. Eng., vol. 19, no. 3, 2022, Doi: 10.1109/TASE.2022.3179356.
[44] P. Meng et al., "Mobile application framework for real-time alarm notification via RESTful microservices," IEEE Access, vol. 4, 2016, Doi: 10.1109/ACCESS.2016.2546288.
[45] Y. Liu et al., "Role-based access control for industrial IoT dashboards," IEEE Internet Things J., vol. 10, no. 20, 2023, Doi: 10.1109/JIOT.2023.3027495.
[46] Q. Zhang et al., "REST versus MQTT for IoT-based industrial monitoring: latency and bandwidth," IEEE Access, vol. 3, 2016, Doi: 10.1109/ACCESS.2016.2547428.
[47] T. Hu et al., "Anomaly detection in industrial time-series using isolation forests," IEEE Trans. Cybern., vol. 50, no. 4, 2020, Doi: 10.1109/TCYB.2018.2880968.
[48] N. Zhao et al., "EWMA-based control chart for current imbalance in three-phase power systems," IEEE Trans. Instrum. Meas., vol. 63, 2014, Doi: 10.1109/TIM.2014.2298584.
[49] X. Guo et al., "Scalable microservices architecture for industrial IoT data processing," IEEE Trans. Ind. Inform., vol. 20, 2024, Doi: 10.1109/TII.2024.3092904.
[50] Z. Shen et al., "React Native cross-platform framework for industrial process monitoring," IEEE Access, vol. 11, 2023, Doi: 10.1109/ACCESS.2023.3031448