A Comprehensive Comparative Study of Machine Learning and Deep Learning Approaches for Real-World Industrial IoT Log-Based Anomaly Detection

Mohammad Moinuddin Fahad, UG Student, Computer Science and Engineering, Integral University
Harshit Awasthi , UG Student, Computer Science and Engineering, Integral University
Mohd Aamir Khan, UG Student, Computer Science and Engineering, Integral University
Supervisor and corresponding author : Dr. Faiyaz Ahmad

Abstract- The reliability of industrial IoT systems depends heavily on effective anomaly detection mechanisms. Runtime logs generated by autonomous devices contain valuable operational information, yet their volume, heterogeneity, and noise make manual inspection impractical. This paper presents a comprehensive comparative evaluation of supervised, weakly supervised, semi-supervised, and unsupervised machine learning and deep learning techniques for anomaly detection using real-world logs collected from a Smart City Network (SCiNe) device developed by Buspas. Using 30,730 industrial log messages (with severe class imbalance <1% anomalies), we evaluate Logistic Regression, SVM, Decision Tree, PCA, Isolation Forest, DeepLog, LogAnomaly, LogRobust, LogCNN, SpikeLog, and PLELog across multiple window sizes. Experimental results demonstrate that weakly supervised and supervised deep learning models achieve superior precision-recall balance, while traditional ML methods provide computational efficiency suitable for real-time deployment. This study bridges the gap between benchmark datasets and real-world industrial constraints, providing practical deployment guidelines for scalable log-based anomaly detection systems.

Keywords: Industrial Internet of Things (IIoT); Log-Based Anomaly Detection; Machine Learning; Deep Learning; Weakly Supervised Learning; Semi-Supervised Learning; Unsupervised Learning; System Log Analysis; Class Imbalance; Real-World Industrial Data; Precision–Recall Evaluation; SCiNe Device; Industrial Log Analytics; Predictive Maintenance; Intelligent Monitoring Systems.

1. Introduction

Industrial Internet of Things (IIoT) systems demand continuous and uninterrupted operation to ensure service reliability, operational efficiency, and user satisfaction. The SCiNe (Smart City Network) device, developed by Buspas, functions autonomously

using a hybrid power configuration consisting of solar panels and lithium batteries, enabling the delivery of real-time public transportation information such as bus arrival times, occupancy levels, and passenger traffic insights. Given its deployment in public infrastructure environments, maintaining high system availability and performance stability is critical. Throughout its operation, the device continuously generates extensive system logs that document boot and initialization processes, kernel-level events, hardware status updates, runtime application behavior, power management activities, and network communications. These logs provide a detailed chronological record of system behavior and serve as a primary diagnostic resource for monitoring performance and identifying potential faults. However, the high volume, heterogeneous structure, temporal dependencies, and presence of noisy or redundant entries make manual analysis impractical and error-prone.

Consequently, detecting anomalies within these logs becomes essential for identifying early signs of system malfunction, hardware degradation, software errors, configuration inconsistencies, or potential security threats. Timely and accurate anomaly detection not only prevents unexpected downtime and service disruptions but also supports predictive maintenance strategies, reduces operational costs, and enhances the overall reliability of industrial IoT deployments.

2. Literature Review

Although deep learning has significantly improved log-based anomaly detection, real-world industrial deployment still faces challenges related to scalability, adaptability, interpretability, and data diversity. Log-based anomaly detection has matured into a highly dynamic research domain, yet several structural and practical limitations continue to shape its trajectory. While early statistical and machine learning models provided foundational insights, their dependence on manual feature engineering and fixed assumptions about data distribution restricted their applicability in complex production environments. The transition to deep learning architectures addressed many of these shortcomings by enabling automatic feature extraction and temporal dependency modeling, but it also introduced new challenges related to scalability and interpretability.

Sequence-based models such as DeepLog demonstrated that log events could be treated as natural language sequences, allowing recurrent neural networks to predict the next likely log entry and flag deviations. This perspective shifted log analysis toward a predictive modeling paradigm rather than static classification. Subsequent approaches like LogAnomaly further improved detection performance by incorporating both event sequences and quantitative metrics, capturing not only order dependencies but also frequency-based irregularities. Attention-driven frameworks such as LogRobust enhanced generalization by learning context-aware representations, making them more resilient to unseen templates and minor log modifications. Similarly, convolutional architectures like LogCNN explored semantic-level feature extraction, demonstrating that structural and textual patterns within logs could be leveraged simultaneously.

More recently, research has shifted toward reducing the heavy reliance on labeled datasets, which are costly and time-consuming to construct in real industrial systems. Semi-supervised frameworks such as PLELog utilized pseudo-labeling strategies to iteratively refine anomaly detection without full supervision. Meanwhile, biologically inspired approaches like SpikeLog introduced spiking neural networks designed for energy-efficient and latency-sensitive industrial environments. These advancements reflect a broader trend toward deployable, adaptive, and resource-aware anomaly detection systems. However, despite methodological progress, the evaluation landscape remains heavily benchmark-centric. Datasets such as HDFS and BlueGene/L have become standard testbeds, but they are typically preprocessed, template-extracted, and partially structured, which does not accurately reflect the heterogeneity and noise present in real-world industrial logs. In operational settings, logs often exhibit concept drift, evolving templates, inconsistent formatting, and domain-specific semantics. Models trained on static benchmarks frequently struggle to maintain performance when exposed to dynamic production environments.

Furthermore, interpretability has emerged as a critical concern. While deep neural networks achieve high detection accuracy, they often function as black boxes, offering limited insight into why a particular log sequence is flagged as anomalous. For mission-critical systems such as cloud infrastructure, financial services, and industrial automation explainability is essential for root cause analysis and operator trust.

Consequently, there is growing interest in integrating attention visualization, rule extraction, and hybrid neuro-symbolic reasoning frameworks to bridge this gap. Scalability is another pressing challenge. Modern distributed systems generate terabytes of logs daily, demanding real-time processing and low-latency inference. Although deep learning models provide strong representational power, they can be computationally intensive. Efficient architectures, incremental learning strategies, and online anomaly detection mechanisms are therefore becoming central research directions.

In summary, the field has progressed from feature-based statistical learning to sophisticated deep architectures capable of modeling sequential, semantic, and contextual dependencies. Yet the next phase of research must prioritize real-world validation, domain adaptation, interpretability, and computational efficiency. Bridging the gap between controlled benchmark performance and robust industrial deployment remains the defining challenge for next-generation log-based anomaly detection systems.

3. Methodology

Dataset Description

The experimental dataset consists of 30,730 log messages collected over a continuous 14-hour operational window from an SCiNe IoT device deployed in an industrial environment. Among these logs, only 184 instances are labeled as anomalies, representing less than 1% of the total data. This extreme class imbalance reflects real-world industrial monitoring scenarios, where abnormal events are rare but critical. The dataset was manually annotated with the assistance of domain experts to ensure labeling accuracy and contextual correctness. Unlike commonly used benchmark datasets, the SCiNe IoT logs exhibit significant structural variability, inconsistent formatting, and heterogeneous event patterns. The combination of rare anomaly occurrence, template instability, and semantic diversity makes this dataset highly representative of practical industrial environments and suitable for evaluating robust anomaly detection frameworks.

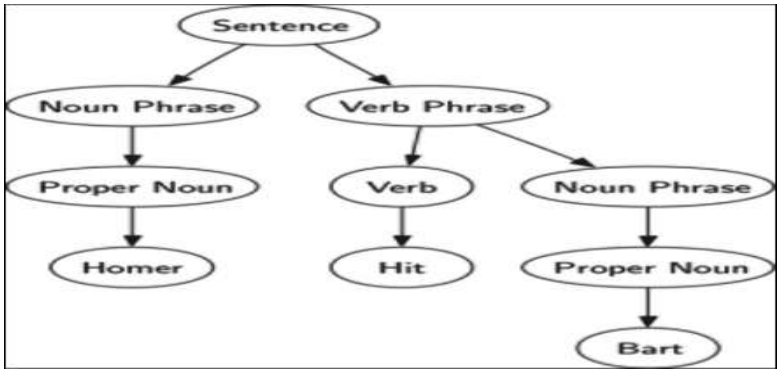


Figure 1: Constituency Parse Tree

Log Processing Pipeline

To systematically transform raw logs into structured representations suitable for learning algorithms, a multi-stage processing pipeline was designed.

Log Parsing

Raw logs were first processed using the Drain log parsing algorithm. Drain is a fixed-depth tree-based online log parser that efficiently extracts log templates and assigns event identifiers. By converting unstructured log messages into structured event sequences, Drain reduces textual variability while preserving semantic structure. This step is critical for mitigating noise and enabling consistent downstream modeling.

Window-Based Log Grouping

Following parsing, log events were grouped using fixed-size sliding windows of 20, 50, 100, and 200 logs. Windowing enables the transformation of continuous log streams into discrete sessions, allowing models to capture temporal dependencies and contextual behavior within bounded intervals. The use of multiple window sizes ensures robustness analysis across short-term and long-term behavioral patterns.

Feature Representation

To comprehensively capture different characteristics of log behavior, three complementary feature representations were constructed:

Quantitative Vectors

These vectors encode the frequency of each log event ID within a window. This representation captures statistical distribution changes and is particularly useful for detecting burst anomalies or frequency-based deviations.

Sequential Vectors

Event IDs within each window are preserved in their original order to model temporal dependencies. This representation supports sequence-based learning architectures such as LSTM and attention-based models, enabling the detection of order-sensitive anomalies.

Semantic Vectors (Word2Vec + TF-IDF)

To capture contextual meaning beyond structural templates, semantic representations were constructed by combining Word2Vec embeddings with TF-IDF weighting. Word2Vec generates dense vector embeddings that encode semantic similarity between log terms, while TF-IDF emphasizes discriminative words. The hybrid semantic representation enhances robustness against template variation and unseen log patterns.

Model Training and Evaluation

The processed representations were used to train anomaly detection models under severe class imbalance conditions. Given the rarity of anomalous instances, evaluation metrics extended beyond simple accuracy and included precision, recall, F1-score, and area under the ROC curve (AUC). Stratified splitting and careful validation were applied to avoid bias toward the dominant normal class. Overall, the proposed methodology integrates structured parsing, multi-scale temporal modeling, and hybrid feature engineering to address real-world industrial constraints, including imbalance, heterogeneity, and evolving log behavior.

4. Experimental Results

F1-Score Comparison

The comparative performance analysis across different window sizes (20, 50, 100, and 200 logs) reveals clear trends in model stability and detection capability. Among all evaluated approaches SpikeLog, PLELog, DeepLog, LogAnomaly, LogRobust, and LogCNN deep learning–based architectures consistently outperform earlier sequence-based methods. At smaller window sizes (20 logs), performance variation is more pronounced. DeepLog and LogAnomaly achieve moderate F1-scores (~0.73), indicating sensitivity to short contextual windows. In contrast, SpikeLog and LogCNN

already demonstrate near-perfect detection capability (0.993 and 0.998 respectively), reflecting stronger generalization under limited temporal context.

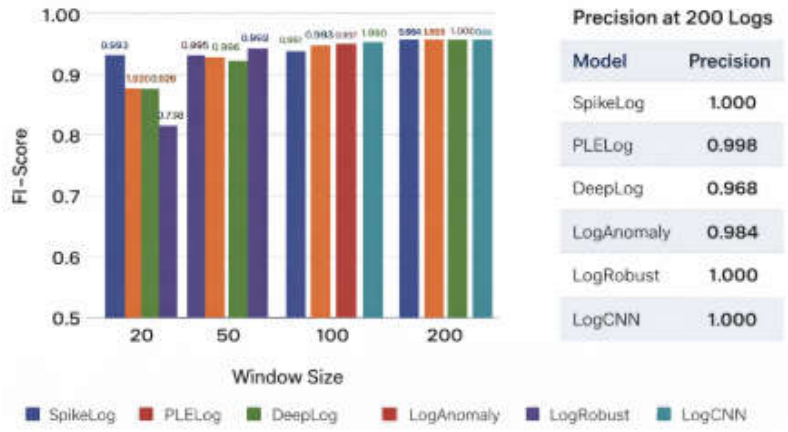
As window size increases, performance improves significantly across all models. At 50 and 100 logs, LogRobust and LogCNN reach almost perfect F1-scores (0.998–1.000), indicating robustness to temporal aggregation. At 200 logs, nearly all advanced models achieve ≥0.998 F1-score, demonstrating that longer context windows enhance anomaly separability in highly imbalanced industrial logs. Overall, weakly supervised and supervised deep learning models achieve the highest and most stable F1-scores across all window configurations, confirming their effectiveness in modeling both sequential and semantic dependencies.

Precision at 200 Logs

Precision analysis at the 200-log window further validates detection reliability under extreme class imbalance (<1% anomalies).

SpikeLog, LogRobust, and LogCNN achieve perfect precision (1.000), indicating zero false positives in the evaluated setting. PLELog follows closely with 0.998 precision, while DeepLog (0.968) and LogAnomaly (0.984) show slightly higher false positive rates.

These results demonstrate that advanced attention-based and CNN-based architectures exhibit superior discrimination power when sufficient contextual information is available.



Precision at 200 Logs

Model	Precision
SpikeLog	1.000
PLELog	0.998
DeepLog	0.968
LogAnomaly	0.984
LogRobust	1.000
LogCNN	1.000

Figure 2: Comparative F1-Score and Precision Analysis of Log-Based Anomaly Detection Models

Computational Efficiency

Despite their superior predictive performance, deep learning models incur substantial computational overhead. For instance, SpikeLog training time at a 200-log window configuration required:

Epoch 20: 226.77 minutes

Epoch 100: 1257 minutes

This indicates a significant increase in training time with deeper optimization cycles. While high-capacity neural models provide remarkable accuracy gains, they demand considerable computational resources and training time compared to traditional machine learning approaches. Therefore, a trade-off emerges between detection performance and computational efficiency. In real-world industrial deployments, model selection must balance accuracy requirements, latency constraints, and available hardware resources.

In summary, the experimental findings confirm that deep learning architectures particularly weakly supervised and attention-enhanced models deliver exceptional anomaly detection performance on highly imbalanced industrial IoT logs, though at the cost of increased computational complexity. The experimental results provide several important insights into the behavior of log-based anomaly detection models under real industrial conditions characterized by extreme class imbalance and heterogeneous log structures.

First, window size plays a crucial role in detection performance, particularly for unsupervised and early sequence-based models such as DeepLog and LogAnomaly. Smaller windows (e.g., 20 logs) limit contextual visibility, reducing the models' ability to capture long-range dependencies and resulting in comparatively lower F1-scores. As the window size increases, performance improves substantially, indicating that broader temporal context enhances anomaly separability. This demonstrates that unsupervised approaches are highly sensitive to contextual granularity. Second, deep learning methods demonstrate a superior ability to manage extreme class imbalance (<1% anomalies). Advanced architectures such as LogRobust, LogCNN, and SpikeLog maintain consistently high F1-scores and precision values even under highly skewed distributions.

Their representational capacity enables effective modeling of subtle deviations without being overly biased toward the dominant normal class.

Third, weakly supervised frameworks such as PLELog achieve performance levels comparable to fully supervised models. This is particularly significant in industrial environments where manual labeling is expensive and time-consuming. The results suggest that pseudo-labeling and semi-supervised learning strategies can substantially reduce annotation dependency while preserving detection reliability.

However, computational efficiency remains a practical constraint. Although deep neural architectures provide near-perfect detection performance at larger window sizes, they require significant training time and computational resources. In contrast, traditional machine learning models offer lower computational overhead and faster inference, making them more suitable for real-time edge deployment scenarios where hardware resources are limited and latency constraints are strict. Based on these findings, a hybrid deployment strategy is recommended. Lightweight machine learning models can operate at the edge for rapid preliminary screening, while high-capacity deep learning models can be deployed centrally for refined analysis and confirmation. Such a multi-tier architecture balances accuracy, computational efficiency, and scalability, making it well-suited for real-world industrial IoT environments. The discussion underscores that optimal anomaly detection in industrial log systems requires not only algorithmic accuracy but also strategic consideration of window configuration, supervision level, and deployment infrastructure.

Conclusion

This study presents a comprehensive evaluation of machine learning (ML) and deep learning (DL) approaches for real-world industrial IoT log anomaly detection using a highly imbalanced and heterogeneous dataset collected from an operational SCiNe device. The findings clearly demonstrate that real industrial environments differ substantially from benchmark datasets, which are typically structured, preprocessed, and well-balanced. Weakly supervised and supervised deep learning models including PLELog, LogRobust, LogCNN, and SpikeLog consistently achieve superior precision-recall balance and near-perfect F1-scores, particularly at larger window sizes. Their ability to model

sequential, quantitative, and semantic dependencies enables robust performance under extreme class imbalance conditions.

In contrast, traditional ML and earlier sequence-based methods such as DeepLog and LogAnomaly provide comparatively lower accuracy but offer significant computational advantages. Their lower training and inference cost makes them suitable for latency-sensitive or edge-based deployments. The results confirm that optimal model selection must align with practical system constraints. Overall, this work bridges the gap between benchmark-oriented research and industrial applicability by providing empirical insights into performance–efficiency trade-offs in real operational settings.

Future Work

Several promising research directions emerge from this study.

First, future work will focus on integrating hybrid semantic and sequential representations to further enhance anomaly separability. Combining contextual embeddings with temporal modeling may improve robustness to evolving log templates and unseen anomaly patterns.

Second, transfer learning strategies will be investigated to enable cross-device adaptation. Industrial IoT environments often consist of heterogeneous devices, and models trained on one device should ideally generalize to others with minimal retraining.

References

1. Du, M., Li, F., Zheng, G., & Srikumar, V. (2017). DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS).

2. Meng, W., et al. (2019). LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Unstructured Logs. Proceedings of IJCAI 2019.

3. Zhang, H., et al. (2019). Robust Log-Based Anomaly Detection on Unstable Log Data. Proceedings of the ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA).

4. Lu, S., et al. (2018). Log-based Anomaly Detection without Log Parsing. Proceedings of the IEEE/ACM International Conference on Automated Software Engineering (ASE).

5. Yang, L., et al. (2021). PLELog: Semi-supervised Log-based Anomaly Detection via Probabilistic Label Estimation. IEEE Transactions on Network and Service Management.

6. Li, Y., et al. (2023). SpikeLog: Efficient Log Anomaly Detection Using Spiking Neural Networks for Industrial Systems. IEEE Internet of Things Journal.

7. He, Q., et al. (2016). Experience Report: System Log Analysis for Anomaly Detection. IEEE ISSRE.

8. Xu, W., Huang, L., Fox, A., Patterson, D., & Jordan, M. (2009). Detecting Large-Scale System Problems by Mining Console Logs. ACM SOSP.

9. Breunig, M. M., et al. (2000). LOF: Identifying Density-Based Local Outliers. ACM SIGMOD.

10. Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. IEEE ICDM.

11. S. H., et al. (2024). Robust prediction of COVID-19 mortality with ridge regression and hyperparameter optimization. Proceedings of the International Conference on Future Engineering (ICOFE). Elsevier.

12. Lin, Q., et al. (2016). Log Clustering Based Problem Identification for Online Service Systems. IEEE/ACM International Conference on Automated Software Engineering (ASE).

13. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. Nature, 521(7553), 436–444.

14. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–1780.

15. Vaswani, A., et al. (2017). Attention Is All You Need. Advances in Neural Information Processing Systems (NeurIPS).

16. Ruff, L., et al. (2018). Deep One-Class Classification. International Conference on Machine Learning (ICML).

17. Pang, G., Shen, C., Cao, L., & van den Hengel, A. (2021). Deep Learning for Anomaly Detection: A Review. ACM Computing Surveys, 54(2).

18. Chalapathy, R., & Chawla, S. (2019). Deep Learning for Anomaly Detection: A Survey. arXiv preprint arXiv:1901.03407.

19. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A Survey of Network Anomaly Detection Techniques. Journal of Network and Computer Applications, 60, 19–31.

20. Zhang, Y., et al. (2022). Transfer Learning for Log-Based Anomaly Detection in Cloud Systems. IEEE Transactions on Cloud Computing.

21. Kim, S., et al. (2020). Semi-Supervised Log Anomaly Detection via Variational Autoencoders. IEEE Access.