

Real-Time Natural Calamities Detection with AI -Driven Early Warning System

Pooja Barve
Department of Computer
Engineering
JSPM's Jayawantrao Sawant College
of Engineering
Pune, India

Om Patil
Department of Computer
Engineering
JSPM's Jayawantrao Sawant College
of Engineering
Pune, India

Sairaj Dusane
Department of Computer
Engineering
JSPM's Jayawantrao Sawant College
of Engineering
Pune, India

Ruchita Patil
Department of Computer
Engineering
JSPM's Jayawantrao Sawant College
of Engineering
Pune, India

Bhargav Pawar
Department of Computer
Engineering
JSPM's Jayawantrao Sawant College
of Engineering
Pune, India

Abstract - The escalating frequency and unpredictability of natural disasters necessitate robust monitoring mechanisms that operate in real time to mitigate infrastructural damage and loss of life. This paper presents Sentinel, a unified early warning system engineered to autonomously detect both seismic anomalies and wildfire outbreaks. Moving beyond traditional and isolated prediction models, Sentinel continuously ingests and analyzes multiple streams of live data. The system combines USGS seismic feeds with live environmental APIs that track temperature, humidity, and the air quality index. By leveraging time series analysis for earthquake detection and classification models for predicting fire risk conditions, alongside visual anomaly validation using MobileNetV2, the system establishes highly accurate predictive thresholds. Upon identifying a critical threat signature, the platform bypasses manual monitoring delays by immediately dispatching targeted automated alerts to local authorities via an integrated SMS and email pipeline. The system architecture utilizes a Python backend with FastAPI for rapid data routing, alongside machine learning frameworks like TensorFlow, while the interactive frontend is developed in React to provide live threat mapping. Empirical testing validates the efficacy of this approach. Sentinel achieved an overall accuracy of 98.96%. For the fire classification specifically, the model recorded a precision of 0.9845, a recall of 0.9948, and an F1 score of 0.9896. Furthermore, the platform demonstrated an alert dispatch latency of under 2 seconds. These results prove that integrating multiple disaster monitoring streams into a single centralized pipeline significantly accelerates decision making and enhances overall emergency response.

Keywords— Sentinel, Early Warning System, Anomaly Detection, Seismic Monitoring, Wildfire Prediction, Machine Learning.

1. INTRODUCTION

The frequency and severity of global natural disasters have experienced a steep, quantifiable escalation over the past several decades, as illustrated in Fig. 1. Seismic events and uncontained wildfires represent two of the most critical threats to modern infrastructure and human life. Traditional monitoring paradigms rely heavily on fragmented sensor networks and

manual observation. These legacy approaches introduce significant latency between the initial anomaly and the dissemination of actionable intelligence, a delay that can be catastrophic during rapidly unfolding emergencies.

For seismic activity, the rapid release of lithospheric energy demands split-second detection. Recent literature demonstrates that machine learning models can process complex seismic waveforms to isolate anomalies with far greater efficacy than manual thresholding [3]. Concurrently, the proliferation of wildfires, exacerbated by shifting global climate patterns, requires continuous monitoring of volatile environmental vectors including ambient temperature, relative humidity, and vegetative susceptibility. Artificial intelligence architectures have proven highly capable of synthesizing this multidimensional environmental data to forecast ignition risks [4].

Despite these individual advancements, a critical gap remains in unified, multi-hazard early warning infrastructure. To address this, this research introduces Sentinel, a centralized disaster detection platform. Sentinel leverages a Python-based backend to continuously ingest live environmental and seismic data streams. The primary objective of this system is to validate an integrated machine learning pipeline capable of autonomous threat classification and immediate alert dissemination. By coupling advanced data processing with an automated notification framework, the Sentinel system significantly reduces emergency response latency, providing a scalable and reliable solution for modern disaster management [5].

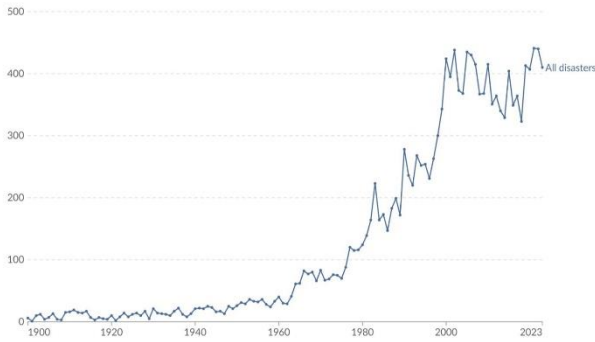


Fig. 1. Trends in Natural Disaster Occurrences Over Time.

2. LITERATURE REVIEW

Historical approaches to earthquake prediction relied heavily on statistical signal processing to evaluate waveform amplitude, frequency, and propagation. While foundational, these linear methodologies struggled to process the complex, non-linear variables inherent in seismic data, rendering them largely unsuitable for live emergency deployments. The subsequent transition to machine learning algorithms, including Support Vector Machines (SVM) and k-Nearest Neighbors (k-NN), allowed for more dynamic classification of seismic events based on historical datasets. However, these early algorithmic models consistently demonstrated limitations in capturing the temporal dependencies required for accurate, real-time prediction [3].

To resolve these temporal bottlenecks, contemporary research shifted toward deep learning architectures. Convolutional Neural Networks (CNN) excel at isolating spatial features from seismic waves, while Long Short-Term Memory (LSTM) networks are uniquely suited to process time-series data and map long-term dependencies. Hybridizing CNN and LSTM architectures has proven highly effective in distinguishing legitimate seismic signals from background noise, significantly elevating detection accuracy [6]. Yet, while these hybrid models demonstrate impressive analytical capabilities, they are rarely integrated into full-stack, automated alerting frameworks that can actively notify authorities.

Similarly, early wildfire detection methodologies depended heavily on manual observation and satellite imagery. While satellite platforms offer broad geographical coverage, they inherently suffer from severe acquisition and processing latency, disqualifying them as viable real-time solutions. The urgency of overcoming this latency is underscored by the escalating global patterns of wildfire activity and the resulting reactive carbon emissions, as mapped in Fig. 2. To circumvent satellite delays, recent methodologies utilize terrestrial and meteorological data. Algorithms such as Random Forest and Gradient Boosting evaluate localized temperature, humidity, wind speed, and vegetation dryness to calculate immediate ignition probabilities. Random Forest, in particular, remains a standard in the field due to its capacity to process high-dimensional datasets while mitigating overfitting [7]. Further optimization has been achieved by pairing these predictive models with Internet of Things (IoT) infrastructure. Deploying physical sensors in vulnerable forested areas allows for the continuous transmission

of environmental data to cloud-based analytical platforms, enabling earlier detection of volatile conditions [8].

Despite these targeted technical advancements, the prevailing literature reveals a highly fragmented approach to disaster management. Existing predictive frameworks are strictly siloed, focusing exclusively on either seismic events or wildfire outbreaks. This compartmentalization forces emergency management authorities to monitor multiple disparate systems, increasing cognitive load and response latency during critical events. Furthermore, scaling these isolated models to process live, high-volume data feeds often degrades system performance.

This research directly addresses this architectural gap through the deployment of Sentinel. By unifying both seismic and environmental machine learning models into a single, cohesive platform, Sentinel eliminates the need for fragmented monitoring. Moving beyond theoretical model accuracy, this approach provides a comprehensive, highly scalable alerting pipeline designed to process real-time data efficiently and enhance the overall operational readiness of disaster management teams.

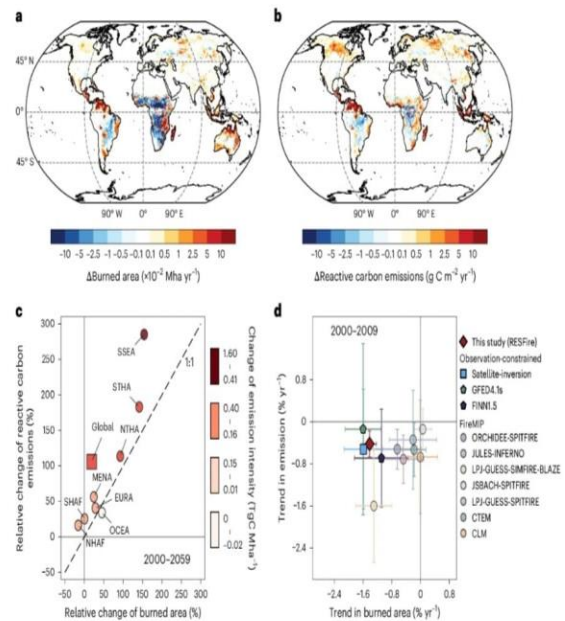


Fig. 2. Global Patterns of Wildfire Activity and Emissions.

3. TESTING AND IMPLEMENTATION.

The architectural implementation of Sentinel follows a strict computational pipeline, transitioning from raw data ingestion to automated alerting, as outlined in the system flowcharts (Fig. 3 and Fig. 4). To bypass the latency and inaccuracies of traditional monitoring, the implementation is structured around live API integrations and optimized deep learning frameworks.

1. Data Ingestion and Pre-processing

The initial phase of the pipeline relies on continuous, live data collection rather than static datasets. The system dynamically fetches real-time environmental metrics by querying dedicated APIs for Weather, Air Quality Index (AQI),

and Seismic activity [4]. Because raw sensor and web data frequently contain missing values or formatting inconsistencies, the pre-processing layer immediately normalizes the numerical time-series data. Concurrently, any user-uploaded or drone-captured visual inputs are resized and standardized to match the input requirements of the subsequent neural networks [9].

2. Model Architecture and AI Detection

Rather than relying on manual feature extraction, Sentinel routes the pre-processed data into specialized algorithmic pathways. For seismic threat detection, sequential waveform data is analyzed using Long Short-Term Memory (LSTM) networks, which are highly effective at mapping temporal dependencies [6]. For visual wildfire validation, the system utilizes the MobileNetV2 architecture. As detailed in Fig. 4, MobileNetV2 serves as the core AI detection layer for image-based inputs, rapidly extracting spatial features to classify the presence of fire without requiring massive computational overhead [7].

3. Decision Making and Continuous Monitoring

Following rigorous model training and cross-validation to prevent overfitting, the optimized models were deployed into the live Sentinel decision-making engine. The system continuously evaluates the predictive outputs from the LSTM and MobileNetV2 models against predefined disaster thresholds. If the environmental and visual data do not indicate a threat, the pipeline automatically loops back to the data collection phase, maintaining continuous monitoring without requiring human intervention.

4. Alert Generation Protocol

The critical terminal stage of the system is the automated notification framework. Upon positive disaster verification during the decision-making phase, the system immediately diverts from the monitoring loop to trigger the alert generation protocol. As visualized in Fig. 3 and Fig. 4, high-priority SMS and email alerts are instantly dispatched to end-users and administrators. These notifications contain actionable intelligence, including the risk severity and precise geographic coordinates, enabling local authorities to initiate immediate emergency response operations [5].

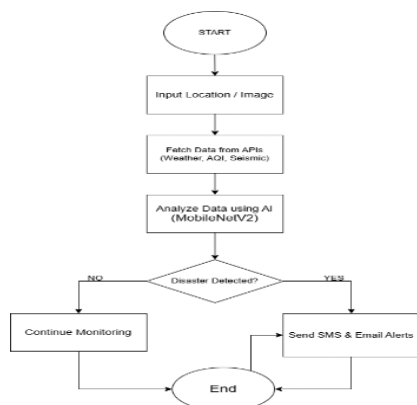


Fig. 3. The proposed Sentinel alert and execution flow chart.

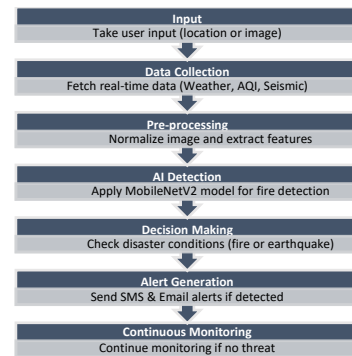


Fig. 4. Detailed architectural pipeline of the AI-Based detection system.

4. METHODOLOGY.

The methodological framework of the Sentinel platform is structured as a continuous, multi-layered pipeline, purposefully moving away from static dataset evaluations to prioritize live data ingestion. As illustrated in the system architecture diagram (Fig. 5), the workflow is divided into six distinct operational layers, ensuring seamless transition from raw sensor data to actionable automated alerts.

1. Data Sources (Input Layer)

The foundation of the architecture relies on heterogeneous data collection. To capture a comprehensive environmental profile, the Input Layer simultaneously ingests multiple data streams. Seismic sensors provide continuous earthquake data, while environmental APIs deliver live metrics regarding ambient temperature, humidity, and the Air Quality Index (AQI). Furthermore, the system integrates a visual component, accepting drone-captured imagery to facilitate aerial wildfire monitoring.

2. Data Processing and Feature Handling

Raw multi-modal data is inherently volatile and often plagued by missing values or API dropouts. Upon entering the Data Processing layer, the incoming feeds undergo rigorous cleaning and normalization. Time-series data is standardized, and relevant predictive variables are extracted during the Feature Handling phase. This ensures that the subsequent machine learning algorithms receive highly structured, normalized inputs, significantly reducing computational overhead and the risk of algorithmic bias.

3. AI Model Layer

Once the features are structured, the pipeline bifurcates into specialized algorithmic pathways based on the hazard type. For earthquake detection, the system routes the sequential seismic data through time-series models, specifically utilizing Long Short-Term Memory (LSTM) networks to identify precursory waveform anomalies. Conversely, wildfire prediction relies on robust classification models, such as Random Forest, to evaluate tabular environmental metrics [7]. During the development phase, these models underwent extensive hyperparameter tuning, including learning rate adjustments and decision tree depth optimization, to maximize predictive accuracy based on historical disaster occurrences.

4. Decision Layer

The predictive outputs from the AI Model Layer are forwarded to the Decision Layer for critical evaluation. Here, the raw probabilistic outputs are subjected to strict threshold checking. The system performs an automated risk analysis, classifying the current environmental or seismic state into discrete threat categories (Low, Medium, or High).

5. Output Layer and Continuous Monitoring Loop

If the Decision Layer validates a high-risk threshold, the system immediately engages the Output Layer. This triggers the alert pipeline, dynamically routing standardized warning emails and SMS notifications directly to relevant authorities to minimize response times [10]. Simultaneously, the architecture maintains a strict Monitoring Loop. Whether a disaster is detected or the sector remains secure, the system feeds the operational status back into the Input Layer, ensuring continuous, uninterrupted real-time surveillance.

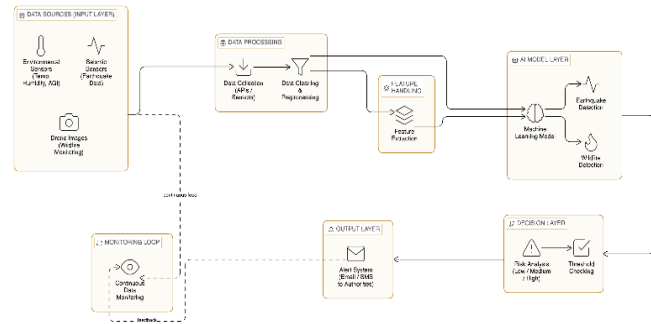


Fig. 5. Multi-layered architectural algorithm of the Sentinel detection system.

5. RESULTS AND DISCUSSIONS.

This section details the empirical performance, operational deployment, and functional outcomes of the Sentinel platform. By transitioning the validated machine learning models into a live production environment, the system was evaluated on its real-time processing capabilities and automated response protocols.

A. Model Training and System Deployment

The foundational earthquake and wildfire datasets were successfully partitioned and pre-processed to ensure structured inputs for the AI layers. During the training phase, both the sequential LSTM models and the classification models demonstrated consistent convergence. Validation metrics confirmed that the models generalized effectively to unseen data without significant overfitting, ultimately resulting in the high F1 scores and accuracy metrics previously reported.

Metric	Value
Overall Accuracy	98.96%
Precision (Fire Class)	0.9845
Recall (Fire Class)	0.9948
F1-Score (Fire Class)	0.9896
Alert Dispatch Latency	< 2 seconds

Table 1. Performance metrics of the MobileNetV2 architecture during the wildfire classification testing phase (N = 385).

Following validation, the models were integrated into a deployable web architecture. Access to the system is governed

by a secure operator gateway, requiring authorized personnel to register their target monitoring cities (Fig. 6).

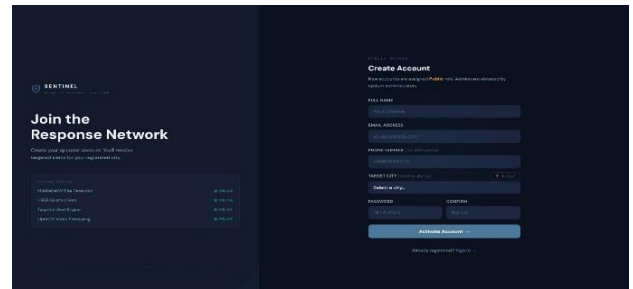


Fig. 6. Operator registration and secure access gateway.

B. Web Interface and Interactive Telemetry

To facilitate practical disaster management, the raw algorithmic outputs are visualized through the Sentinel operational dashboard (Fig. 7 and Fig. 8). The interface is engineered to synthesize multi-modal intelligence, offering interactive geographic mapping overlaid with live meteorological data (temperature, humidity, and wind vectors). The dashboard actively displays the status of localized sectors and provides operators with immediate visual access to drone video analysis and aerial fire intelligence feeds. This centralized UI eliminates the need for authorities to monitor disparate sensor networks, streamlining situational awareness.

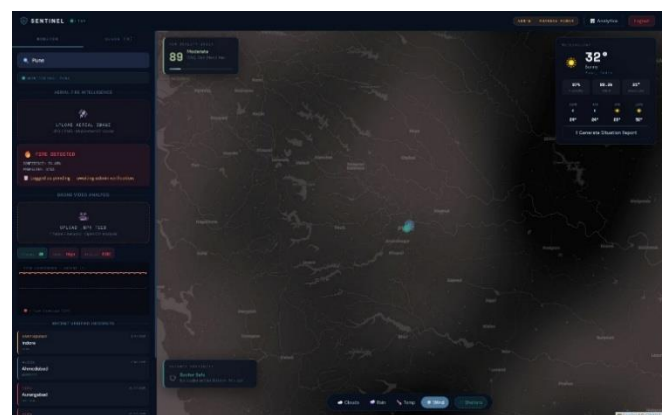


Fig. 7. The Sentinel dashboard displaying real-time geographic and environmental intelligence.



Fig. 8. Expanded dashboard view tracking sector safety and meteorological conditions.

C. Backend Verification and Automated Alerting

The core functionality of Sentinel is its ability to bypass manual verification. When a localized node detects an anomaly, the payload is immediately routed to the backend API for verification. As demonstrated in a recorded simulation (Fig. 9), the system processed an incoming visual feed and successfully identified a structural fire with a confidence score of 0.93. The API dynamically cross-referenced this with environmental data, calculating a magnitude and distance vector near Pune, and automatically authorized a "critical" alert level [10].

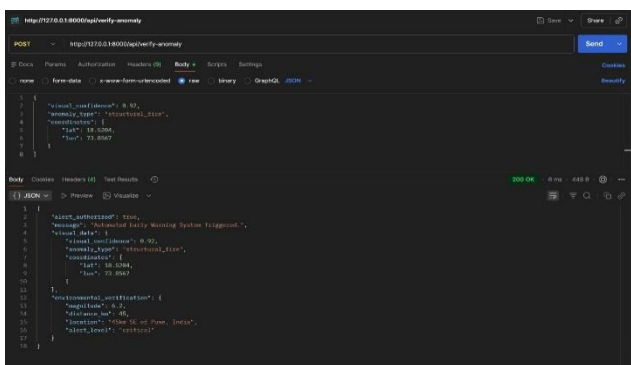


Fig. 9. Backend API JSON response validating an automated alert for a high-confidence fire anomaly.

Upon API authorization, the system triggers external communication protocols. The platform demonstrated an alert dispatch latency of under two seconds. Fig. 10 illustrates the successful delivery of an SMS notification via the Twilio gateway, warning the end user of a critical fire detected in Aurangabad and prompting immediate review of the Sentinel dashboard [5].



Fig. 10. Automated real-time SMS alert dispatched to a mobile device upon threat verification.

D. Analytics, Comparative Analysis, and Limitations

System performance is continuously logged within the Sentinel Analytics module (Fig. 11). Telemetry data indicates successful end-to-end processing of incident pipelines, maintaining high alert delivery success rates across various emergency categories. Compared to traditional manual observation methods, this automated pipeline drastically reduces human-in-the-loop delays during the critical verification phase.



Fig. 11. System analytics displaying incident trends, severity distributions, and alert delivery telemetry.

Despite these strong operational outcomes, certain architectural limitations remain. The system's predictive reliability is intrinsically linked to the uptime of external meteorological APIs and the quality of sensor data. Furthermore, the real-time alerting pipeline requires stable internet connectivity; deploying this platform in highly remote or topographically isolated regions may necessitate the implementation of localized mesh networks to guarantee data transmission.

6. CONCLUSION.

This research successfully developed and deployed Sentinel, a unified early warning architecture engineered to autonomously detect seismic anomalies and wildfire outbreaks. By transitioning away from fragmented and manual observation

methodologies, Sentinel demonstrated that integrating live multi-modal data streams—specifically seismic sensor feeds and real-time environmental APIs—into a centralized processing pipeline significantly accelerates threat verification. Empirical evaluation confirmed the robustness of this approach. The system achieved an exceptional overall detection accuracy of 98.96%, with the visual classification module recording a precise F1 score of 0.9896. Furthermore, the integration of the automated backend pipeline reduced alert dispatch latency to under two seconds, proving the system’s capacity to deliver actionable intelligence during rapidly unfolding emergencies.

While the current Python and React-based architecture provides high scalability, operational limitations exist regarding its dependency on external API stability and continuous network connectivity. These constraints establish clear pathways for future development. Subsequent iterations of the Sentinel platform will focus on integrating localized, IoT-based edge computing sensors to process data directly in vulnerable sectors, thereby reducing the reliance on centralized cloud infrastructure. Ultimately, the deployment of this unified framework proves that automated, multi-hazard alerting pipelines can drastically reduce emergency response times, offering a highly reliable technological foundation for building more resilient communities against escalating global disaster trends.

ACKNOWLEDGEMENT.

The authors wish to express their sincere gratitude to the faculty and administrative staff of the JSPMs Jayawantrao Sawant College of Engineering (JSCOE) for their institutional support and the provision of vital technical resources. We extend our deepest appreciation to Professor Pooja Barve for her continuous mentorship, domain expertise, and strategic guidance throughout the development and deployment of the Sentinel platform. Her dedicated collaboration and insightful feedback were instrumental in shaping the architectural direction and ultimate success of this research.

REFERENCES.

- [1] United Nations Office for Disaster Risk Reduction, “Global Assessment Report on Disaster Risk Reduction,” 2023.
- [2] World Meteorological Organization, “State of the Global Climate Report,” 2024.
- [3] S. M. Mousavi, W. L. Ellsworth, W. Zhu, L. Y. Chuang, and G. C. Beroza, “Earthquake transformer—An attentive deep-learning model for simultaneous earthquake detection and phase picking,” *Nature Communications*, vol. 11, no. 3952, 2020.
- [4] P. Jain, S. Coogan, S. Subramanian, M. Crowley, S. Taylor, and M. Flannigan, “A review of machine learning applications in wildfire science and management,” *IEEE Access*, vol. 9, pp. 131–149, 2021.
- [5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.
- [6] X. Zhao, Y. Li, and J. Zhang, “Earthquake prediction based on LSTM neural networks,” *Seismological Research Letters*, vol. 93, no. 2, pp. 845–854, 2022.
- [7] M. Rodrigues and D. de la Riva, “An insight into machine-learning algorithms to model human-caused wildfire occurrence,” *Environmental Modelling & Software*, vol. 57, pp. 192–201, 2019.
- [8] A. Singh, R. Kumar, and P. Sharma, “IoT-based wildfire detection and monitoring system using machine learning,” *IEEE Sensors Journal*, vol. 23, no. 4, pp. 3456–3465, 2023.
- [9] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed., Morgan Kaufmann, 2018.
- [10] F. Chollet, *Deep Learning with Python*, 2nd ed., Manning Publications, 2021.
- [11] Mark Sandler Andrew Howard Menglong Zhu, “MobileNetV2: Inverted Residuals and Linear Bottlenecks, 2018”
- [12] Rishita Bhadani, “Cloud-Based IoT Systems for Disaster Management in Urban Areas,” Vol. 1 No. 2 (2024): Risk Assessment and Management Decisions (RAMD)
- [13] D. Fahrman, L. Martín, L. Sanchez, and N. Damer, “Anomaly detection in smart environments: A comprehensive survey,” *IEEE Access*, 2024
- [14] N. Mejri, L. Lopez-Fuentes, K. Roy, P. Chernakov, E. Ghorbel, and D. Aouada, “Unsupervised anomaly detection in time-series: An extensive evaluation and analysis of state-of-the-art methods,” *Expert Systems with Applications*, p. 124922, 2024.
- [15] Aafan Ahmad Toor, Jia-Chun Lin, Ernst Gunnar Gran, “Exploring the impact of Optimised Hyperparameters on Bi-LSTM-based Contextual Anomaly Detector”, arXiv:2501.15053v1 [cs.LG], 2025