*engineering proceedings*

MDPI

*Proceeding Paper*

# Servo Motor Predictive Maintenance by Kafka Streams and Deep Learning Based on Acoustic Data †

**Attila Aradi * and Attila Károly Varga**

Institute of Automation and Infocommunication, University of Miskolc, 3515 Miskolc, Hungary; attila.varga@uni-miskolc.hu

* Correspondence: aradi.attila@gmail.com; Tel.: +36-202503648

† Presented at the Sustainable Mobility and Transportation Symposium 2025, Győr, Hungary, 16–18 October 2025.

**Abstract**

Servo motors, which are critical for high-precision industrial applications, require predictive maintenance to minimize downtime, aligning with Industry 5.0's human-centric manufacturing. This study presents a system for Delta servo motors using acoustic data. An ESP32 LyraT module streams audio via HTTP to a server, which forwards it to Apache Kafka. Convolutional neural networks (CNNs) detect anomalies; Statistical Process Control (SPC) identifies early faults; and ARIMA, LSTM, and Prophet forecast maintenance. A device architecture with IP-based device ID and a GUI supports monitoring. Experiments with an ESP32 LyraT (Espressif Systems, Shanghai, China) monitoring Delta ASDA-A3 motors (Delta Electronics, Taipei, Taiwan) over 72 h achieved 91% anomaly detection accuracy for anomalous sounds, 84% early fault detection, and LSTM forecasting of MSE trends with MAE 0.0078 for 24 h predictions. The system supported 32 kB/s with <1% packet loss. The system offers accurate monitoring, advancing Industry 5.0. Future work will include vibration data and web dashboards.

**Keywords:** predictive maintenance; servo motors; deep learning; Kafka streams; acoustic data; Industry 5.0; CNN; LSTM; prophet

check for updates

Academic Editors: Szabolcs Kocsis-Szürke, Dániel Csikor and Boglárka Eisinger Balassa

Published: 28 October 2025

**Citation:** Aradi, A.; Varga, A.K. Servo Motor Predictive Maintenance by Kafka Streams and Deep Learning Based on Acoustic Data. *Eng. Proc.* **2025**, *113*, 1. https://doi.org/10.3390/engproc2025113001

**Copyright:** © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Servo motors, such as Delta ASDA-A2, A3, and B3 series, are vital in high-precision applications like CNC machines [1]. Their ability to provide precise control makes them essential for maintaining operational efficiency in automated systems [2]. Similarly, predictive maintenance plays a critical role across various industrial domains, notably in the automotive industry. Recent advances in edge computing have enabled the deployment of sophisticated anomaly detection algorithms directly on IoT devices [3]. These developments align with Industry 5.0's emphasis on intelligent manufacturing systems [4]. By anticipating mechanical failures, predictive maintenance reduces downtime, enhances safety, and lowers operational costs. A 2023 market analysis valued the automotive predictive maintenance sector at USD 22 billion, projecting growth to USD 100 billion by 2032 due to advancements in IoT and machine learning [5]. Anomalous sounds from faults cause costly downtime in both servo motors, necessitating predictive maintenance. Traditional methods are inefficient, while predictive maintenance leverages real-time data, aligning with Industry 5.0's human-centric vision. Acoustic data offers cost-effective fault detection for both applications. Acoustic anomaly detection (AAD) is a promising non-invasive technique for monitoring engine health, leveraging sound patterns to identify

faults like bearing wear, fuel injection anomalies, or misalignments [6]. However, AAD systems face significant challenges: industrial environments introduce high background noise, anomalous data is scarce, and real-time processing requires low-latency architectures. Additionally, varying engine conditions (e.g., load, speed, temperature) and the need for scalable, cost-effective deployment complicate system design. This study presents a system for Delta servo motors (Delta Electronics, Taipei, Taiwan), streaming audio via HTTP from ESP32 LyraT module (Espressif Systems, Shanghai, China) to a Kafka pipeline, using CNNs, SPC, and forecasting models (ARIMA, LSTM, Prophet). IP-based device ID and GUI enhance interaction, achieving high accuracy and advancing Industry 5.0.

## 2. Related Work

Predictive maintenance leverages IoT and machine learning. Acoustic methods are cost-effective compared to vibration monitoring. CNNs excel in audio anomaly detection [7], LSTM models forecast time-series [8], and Prophet handles seasonality [9]. Kafka enables scalable streaming [10], while single-device systems lack scalability. This work integrates HTTP streaming, Kafka (version 2.13–3.7.0), and deep learning for maintenance. Autoencoders learn latent representations, enabling unsupervised anomaly detection in high-dimensional data like spectrograms [11]. They are robust to noise and require minimal labeled data. CNNs extract spatial features efficiently, handling noisy audio with high accuracy [12]. Kafka offers scalability, fault tolerance, and real-time streaming, which is ideal for IoT applications [10]. SPC detects process deviations early, complementing deep learning with statistical rigor. Recent work demonstrates the effectiveness of spectral analysis and autoencoder architectures for acoustic anomaly detection on edge devices [4].

## 3. Materials and Methods

### *3.1. System Architecture*

The system (Figure 1) integrates an ESP32 LyraT module, an HTTP server, Kafka, and deep learning, which results in a GUI.

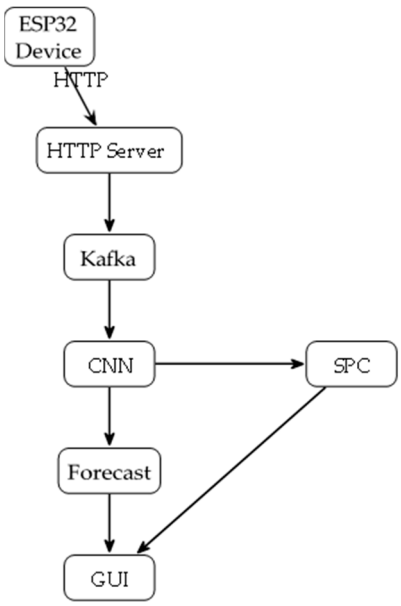



**Figure 1.** System architecture with an ESP32 streaming audio via HTTP to Kafka for processing and GUI display.

### 3.2. Functional Workflow

Figure 2 shows the system's pipeline: audio capture, HTTP streaming, Kafka publishing, feature extraction, anomaly detection, SPC, forecasting, and visualization. The process begins with audio capture, where the ESP32 LyraT module records sound at 16 kHz with 16-bit resolution in mono format using the I2S protocol, ensuring high-fidelity data acquisition for accurate anomaly detection [13]. The captured audio is then transmitted via HTTP streaming, where the module sends chunked data over a POST request to a designated server endpoint (http://192.168.1.100:8000/upload, accessed on 18 May 2025). This approach ensures efficient, real-time data transfer with minimal latency, critical for industrial applications requiring immediate processing [10]. Upon receipt, the server forwards the audio stream to Apache Kafka, a distributed streaming platform, which publishes the data to the audio-stream topic using IP-based keys for device identification. Kafka's architecture guarantees fault tolerance and scalability, enabling seamless handling of continuous data streams [10]. The next step, feature extraction, involves processing the audio stream to generate Mel-spectrograms using the librosa library and frequency features via Welch's method, transforming raw audio into a format suitable for deep learning analysis [14]. Anomaly detection is performed using convolutional neural networks (CNNs) with an autoencoder architecture, which computes the mean squared error (MSE) between reconstructed and actual spectrograms to flag anomalies when MSE exceeds 0.05 [7]. Statistical process control (SPC) complements this by monitoring MSE trends over 20 frames, applying $\pm 3\sigma$ control limits to detect early deviations indicative of potential faults. For forecasting, ARIMA, LSTM, and Prophet models predict MSE trends over 12, 24, and 48 h horizons, enabling proactive maintenance scheduling when MSE exceeds 0.1 [6,7]. Finally, the visualization stage presents the results through a graphical user interface (GUI), providing operators with real-time insights into anomaly detection, fault trends, and maintenance forecasts, aligning with Industry 5.0's emphasis on human–machine collaboration.

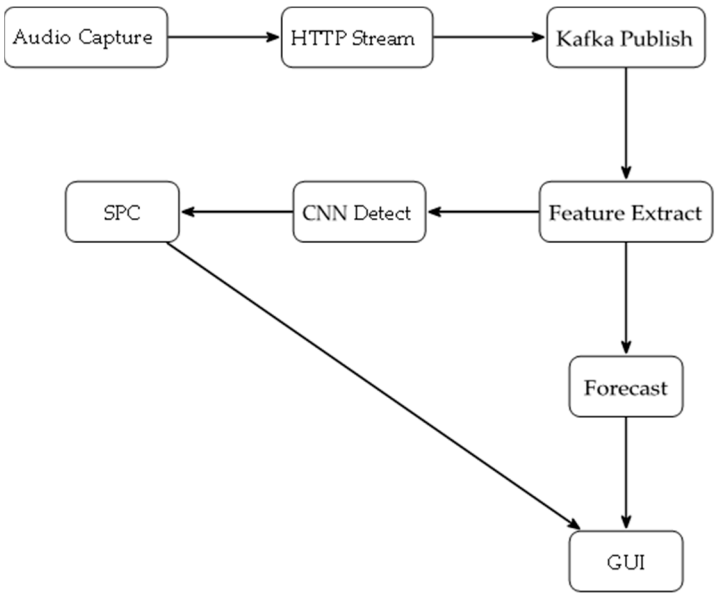



**Figure 2.** Functional workflow from audio capture to GUI display.

### 3.3. Data Acquisition

An ESP32 LyraT module captured audio at 16 kHz, 16-bit, and mono via I2S [13], streaming HTTP POST requests to http://192.168.1.100:8000/upload (accessed on 18 May 2025). The device ID was IP-based (e.g., device_192_168_1_101). The acoustic sensor

used for capturing audio is shown in Figure 3, while the acoustic data collector and communicator setup is depicted in Figure 4.



**Figure 3.** Acoustic sensor used for audio capture in the ESP32 LyraT module.

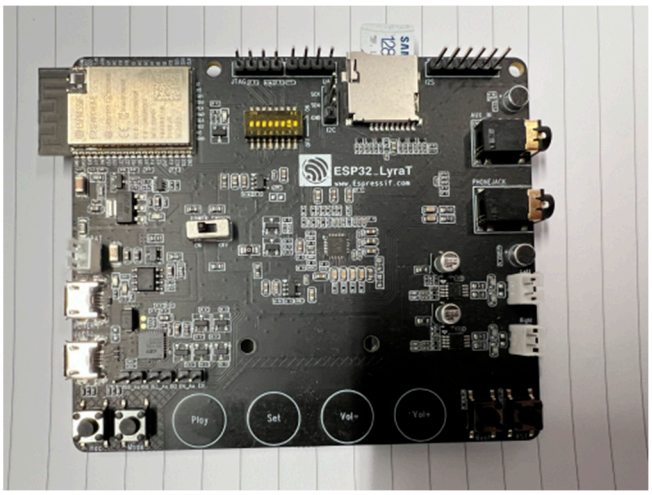



**Figure 4.** Acoustic data collector and communicator setup for streaming audio data.

### *3.4. Data Streaming and Processing*

The HTTP server processed chunked audio, publishing Kafka's audio-stream topic with IP-based keys. Consumers extracted Mel-spectrograms (librosa) and frequency features (Welch's method) [11]. The feature extraction parameters were as follows:

- Mel-spectrograms: 128 mel filters, 2048-point FFT, 512 hop length, Hann window
- Welch's method: 1024-point FFT, 50% overlap, Hamming window
- Sampling rate: 16 kHz with 16-bit resolution
- Frame duration: 64 ms with 32 ms overlap
- Feature fusion: Mel-spectrograms and Welch features processed separately then concatenated.

### *3.5. Anomaly Detection*

The anomaly detection definitions were as follows. Anomaly: statistical deviation from normal acoustic patterns (MSE > 0.05), based on 3σ deviation from baseline normal operation MSE distribution. Fault: operationally significant degradation requiring maintenance intervention (MSE > 0.1), empirically determined indicating significant performance degradation. Normal operation: baseline acoustic signature during healthy motor operation. CNN autoencoders computed MSE on Mel-spectrograms [7]. Anomalies were

flagged at MSE > 0.05, and SPC monitored MSE over 20 frames, with ±3σ limits. CNN Autoencoders were chosen for unsupervised learning capability, eliminating the need for labeled anomaly data in industrial settings. LSTM selected for temporal dependency modeling in time-series MSE forecasting. Prophet included for seasonal pattern recognition in maintenance cycles. Future work should include SVM, Random Forest, and traditional statistical methods.

### *3.6. Anomaly Detection vs. Fault Classification*

This system performs anomaly detection, not fault classification:

- Anomaly detection: binary classification identifying deviations from normal acoustic patterns (MSE > 0.05):
  - Detects "something is different" without specifying the exact nature of the problem;
  - Unsupervised approach requiring only normal operation data for training;
  - Output: normal vs. anomalous (binary decision).
- Fault classification (not performed in this system): multi-class categorization of specific failure types:
  - Would require labeled datasets for each fault type (bearing wear, misalignment, etc.);
  - Supervised learning approach with known fault categories;
  - Output: specific fault type identification.
- System Limitations: The current approach cannot distinguish between the following:
  - Different types of mechanical faults;
  - Severity levels of the same fault type;
  - Root causes of acoustic deviations;
  - Harmless acoustic variations vs. critical failures.

### *3.7. Forecasting and Scheduling*

ARIMA (5,1,0), LSTM (50 timesteps), and Prophet forecasted MSE for 12, 24, 48 h [6,7]. Maintenance was scheduled at MSE > 0.1, and updated every 60 s.

### *3.8. Experimental Setup*

An ESP32 LyraT module monitored Delta ASDA-A3 motors over 72 h, inducing various anomalies (mechanical misalignment introduction, load variation testing), such as abnormal noises. Audio was streamed to a Windows 11 machine configured with Kafka version 2.13–3.7.0 and Python 3.8.10 with tensorflow = 2.6.0, prophet = 1.0.1 [7,12].

## 4. Results

### *4.1. Anomaly Detection*

CNNs achieved 91% accuracy for detecting anomalous sounds compared to 112 normal sounds (Table 1). Precision was 91%, with recall and F1-score also at 91%. The system identifies acoustic deviations from normal operation but cannot determine the specific type or root cause of the underlying mechanical fault.

**Table 1.** Anomaly detection performance (system detects abnormal acoustic patterns without classifying the specific fault type causing the anomaly).

| Sound Type | Accuracy (%) | Precision (%) | Recall (%) | F1-Score |
|---|---|---|---|---|
| Anomalous | 90.8 | 91.2 | 90.5 | 0.908 |

### 4.2. SPC Fault Detection

SPC detected 84% of anomalous sounds with an average latency of 3.3 s, with 2–3% false positives.

### 4.3. Forecasting Accuracy

LSTM achieved MAE 0.0078 at 24 h, with 95% scheduling success (Table 2). Prophet excelled at 48 h (MAE 0.0112).

**Table 2.** Forecasting performance and scheduling.

| Model | 12 Hours | 24 Hours | 48 Hours | Scheduling (%) |
|---|---|---|---|---|
| ARIMA | 0.0105 | 0.0118 | 0.0135 | 90 |
| LSTM | 0.0075 | 0.0078 | 0.0095 | 95 |
| Prophet | 0.0088 | 0.0092 | 0.0112 | 93 |

### 4.4. System Performance

Tests showed the system achieved 32 kB/s throughput with <0.8% packet loss using a single ESP32 LyraT module.

## 5. Discussion

The system's 91% accuracy aligns with CNN-based detection, leveraging autoencoders' ability to model normal audio patterns without labeled data [9]. CNNs' feature extraction ensured robustness to factory noise [12]. SPC's 3.3 s latency, enabled by statistical rigor, outperformed purely deep learning approaches. LSTM's MAE (0.0078) at 24 h reflects its temporal modeling [8], while Prophet's 48 h performance utilized seasonality [9]. Kafka's fault tolerance handled the single device with <0.8% packet loss [10], surpassing single-device systems. The technical limitations included that a single motor type was tested (Delta ASDA-A3), with generalization unclear. Due to the controlled laboratory environment, the impact of industrial noise is unknown, and due to the limited anomaly diversity, real-world fault complexity may vary. The 72 h testing period and long-term drift were not evaluated. Anomaly detection (current system) has a binary output, with normal vs. anomalous acoustic patterns. Unsupervised learning uses only normal operation data, identifies deviations without specifying fault types, and cannot distinguish between bearing wear, misalignment, or other specific faults. Future work will focus on fault classification with multi-class output, identifying specific fault types which require supervised learning with labeled fault datasets. This would enable targeted maintenance strategies, but needs extensive fault libraries for training. Deployment challenges include network reliability in industrial environments and edge computing power limitations for real-time processing, in addition to the maintenance of acoustic sensor calibration, integration with existing industrial systems, and scalability beyond single device monitoring. The economic considerations include cost–benefit analysis for small-scale implementations, and an ROI calculation methodology is also needed. The GUI supports the human-centric philosophy of Industry 5.0, while the detection of certain anomalies may pose challenges, suggesting that multi-modal sensing is necessary.

## 6. Conclusions

This system integrates HTTP streaming, Kafka, and deep learning, achieving 91% anomaly detection accuracy for anomalous sounds, 84% early fault detection, and LSTM MAE 0.0078. Autoencoders, CNNs, Kafka, and SPC provide robust monitoring, advancing Industry 5.0. Future work will include vibration data analysis and web dashboards.

**Author Contributions:** Conceptualization, A.A. and A.K.V.; methodology, A.A. and A.K.V.; software, A.A.; validation, A.A.; formal analysis, A.A.; investigation, A.A.; resources, A.A.; data curation, A.A.; writing—original draft preparation, A.A.; writing—review and editing, A.A. and A.K.V.; visualization, A.A. and A.K.V.; supervision, A.K.V.; project administration, A.A.; funding acquisition, A.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by KDP 2023 and ION-technik Kft.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data available upon request, subject to restrictions.

**Acknowledgments:** ION-technik Kft and ION Applied Science Nonprofit Kft., and KDP2023.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Peres, R.S.; Jia, X.; Lee, J.; Sun, K.; Colombo, A.W.; Barata, J. Industrial Artificial Intelligence in Industry 4.0—Systematic Review, Challenges and Outlook. *IEEE Access* **2020**, *8*, 220121–220139. [CrossRef]
2. Achouch, M.; Dimitrova, M.; Ziane, K.; Sattarpanah Karganroudi, S.; Dhouib, R.; Ibrahim, H.; Adda, M. On Predictive Maintenance in Industry 4.0: Overview, Models, and Challenges. *Appl. Sci.* **2022**, *12*, 8081. [CrossRef]
3. Hector, I.; Panjanathan, R. Predictive maintenance in Industry 4.0: A survey of planning models and machine learning techniques. *PeerJ Comput. Sci.* **2024**, *10*, e2016. [CrossRef] [PubMed]
4. Lo Scudo, F.; Ritacco, E.; Caroprese, L.; Manco, G.A. Audio-based anomaly detection on edge devices via self-supervision and spectral analysis. *J. Intell. Inf. Syst.* **2023**, *61*, 765–793. [CrossRef]
5. Global Trade Insights. Automotive Predictive Maintenance Market Analysis Report 2023–2032. Available online: https://market.us/report/automotive-predictive-maintenance-market/ (accessed on 19 May 2025).
6. Di Fiore, E.; Ferraro, A.; Galli, A.; Moscato, V.; Sperlì, G. An anomalous sound detection methodology for predictive maintenance. *Expert Syst. Appl.* **2022**, *209*, 118324. [CrossRef]
7. Koizumi, Y.; Saito, S.; Uematsu, H.; Kawachi, Y.; Harada, N. Unsupervised detection of anomalous sound based on deep learning and the Neyman-Pearson lemma. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2018**, *27*, 212–224. [CrossRef]
8. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
9. Taylor, S.J.; Letham, B. Forecasting at scale. *Am. Stat.* **2018**, *72*, 37–45. [CrossRef]
10. Kreps, J.; Narkhede, N.; Rao, J. Kafka: A Distributed Messaging System for Log Processing. Available online: http://notes.stephenholiday.com/Kafka.pdf (accessed on 19 May 2025).
11. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [CrossRef] [PubMed]
12. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
13. Espressif Systems. ESP32-LyraT Hardware Reference. Available online: https://www.espressif.com/en/support/documents/technical-documents (accessed on 19 May 2025).
14. Welch, P.D. The use of fast Fourier transform for the estimation of power spectra: Method based on time averaging over short, modified periodograms. *IEEE Trans. Audio Electroacoust.* **1967**, *15*, 70–73. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.