Hacettepe University

Computer Engineering Department

AIN429 Data Mining Laboratory - 2025 Fall

Real-Time Energy Monitoring and Anomaly Detection System

December 29, 2025

Student Name:
Emir Çağıl Kökdener
Abdullah Edik

Student Number:
b22207650636
b2220765031

Abstract

This report presents a real-time smart meter analytics pipeline built on Apache Kafka and Confluent Cloud, integrating streaming data ingestion, trend analysis, hybrid online anomaly detection, and online performance evaluation. Energy consumption and weather data are streamed via synchronized pipelines to ensure data consistency and compatibility across services. A modular microservice architecture is adopted, where independent streaming services perform rolling trend analysis and hybrid anomaly detection, visualizing the results through an interactive Streamlit dashboard. The proposed system demonstrates the feasibility of scalable, real-time energy monitoring and anomaly detection, providing a robust foundation for intelligent energy management and decision support systems.

1 Introduction

With the increasing adoption of artificial intelligence and Internet of Things (IoT) technologies, living spaces and infrastructures are being transformed into "smart" environments. However, the widespread adoption of technology does not inherently imply efficiency or intelligent operation. Energy consumption holds critical importance for both individual users' economic well-being and the sustainability of energy grids, as inefficient usage and uncontrolled consumption may result in significant financial losses and resource waste. Analyzing energy consumption solely based on historical meter readings may therefore lead to misleading conclusions, as contextual factors underlying such changes must be carefully examined.

Energy demand cannot be considered independently of external factors, particularly weather conditions. Previous studies have demonstrated that electricity consumption exhibits a strong and non-linear relationship with temperature, driven by heating and cooling demands. This relationship is often characterized by a U-shaped pattern, where electricity demand increases during both cold and hot periods [1]. Unless these natural, weather-driven variations are properly separated from the data, it becomes extremely difficult to identify the true performance of an energy system or to detect genuine inefficiencies and losses.

The primary objective of this project is to construct a context-aware real-time data pipeline. To this end, a publicly available dataset obtained from Kaggle, containing both energy consumption measurements and associated weather information, is utilized within a streaming framework. By leveraging Apache Kafka, energy consumption streams are continuously enriched with weather data to assess whether contextual awareness enables more intelligent and efficient energy management.

2 Methodology

2.1 System Architecture

The proposed system architecture is designed following an event-driven paradigm and is implemented using Confluent Kafka to ensure high throughput, scalability, and fault tolerance. The data flow is orchestrated through a series of decoupled Kafka topics. Initially, raw data are ingested into source topics, processed by dedicated consumer applications, and the resulting outputs are published to downstream topics.

At the final stage of the pipeline, a streaming dashboard consumes all generated topics to provide a holistic and real-time view of the system state. To prevent schema evolution conflicts and ensure data integrity across distributed components, Confluent Schema Registry is employed for strict schema enforcement throughout the pipeline.

2.2 Data Ingestion and Integration

The *Smart Meter Electricity Consumption Dataset* [2] is utilized as the primary data source in this study. The dataset is logically partitioned into two distinct streams: energy consumption data and meteorological data. Although the dataset itself is static, it is ingested into the Kafka cluster in batches to emulate a real-time streaming environment.

This ingestion process is managed by two independent Kafka producers, one for each data stream. Given the collaborative nature of the project development, maintaining consistency across services is considered critical. Therefore, predefined JSON schemas registered in the Schema Registry are enforced from the beginning of the pipeline. This mechanism ensures full compatibility among producers and consumers, eliminating potential serialization and deserialization errors during integration.

2.3 Trend Analysis

To identify directional changes in energy consumption, a trend analysis module is implemented. A fixed-size sliding window approach is used to compute the Rolling Moving Average (RMA) over recent observations. The current energy consumption value is compared against this average, and the trend is classified as either *UP* or *DOWN*.

While schema-based serialization is used internally, the output of the trend analysis module is serialized using plain JSON format. This design choice is made to reduce latency and computational overhead in the visualization layer, enabling near-instantaneous updates in the real-time dashboard.

2.4 Hybrid Online Anomaly Detection Framework

The anomaly detection framework is redesigned around online learning, using River's Half-Space Trees (HST) as the primary detection model. HST incrementally updates its internal structure with each incoming observation, making it suitable for real-time streaming environments where retraining batch models is infeasible.

The formal decision logic of the system is defined as follows:

$$\text{Anomaly} = \text{HST} \wedge \text{STL}$$

2.4.1 Half-Space Trees

Half-Space Trees (HST) is an unsupervised anomaly detection algorithm designed for data streams. Unlike batch-based models, HST incrementally learns from each incoming data point without requiring access to historical data. The model partitions the feature space using randomly generated trees and assigns anomaly scores based on how isolated an observation is within these partitions.

Due to its low memory footprint and online learning capability, HST is well suited for real-time energy consumption monitoring.

2.4.2 STL Decomposition

The final verification layer relies on Seasonal-Trend decomposition using LOESS (STL), a robust statistical technique for time-series analysis [3]. STL decomposes the signal into seasonal, trend, and residual components. In this framework, the residual component is analyzed by computing a Z-score. If the absolute Z-score exceeds a predefined threshold of $|Z| > 2.5$, the observation is statistically confirmed as an outlier, serving as a final validation step for machine learning-based detections.

2.5 Streaming Implementation

Since the primary anomaly detection model operates in an online manner, no batch retraining is required. Each incoming observation is processed sequentially and immediately incorporated into the model. A fixed-size sliding window is maintained only for statistical validation using STL decomposition, which requires a minimum number of recent observations to estimate seasonal and residual components.

2.6 Real-Time Evaluation

The performance of the anomaly detection framework is evaluated in real time by comparing predicted anomaly labels with ground truth labels provided in the dataset. Due to the inherent class imbalance in anomaly detection problems, accuracy alone is not considered a sufficient metric. Instead, precision, recall, and particularly the F1-score are emphasized. The F1-score, defined as the harmonic mean of precision and recall, is used as the primary performance indicator to ensure a balanced evaluation.

2.7 Visualization

A centralized real-time dashboard is developed to visualize the entire data pipeline. The dashboard displays streaming energy consumption metrics alongside meteorological variables such as temperature and humidity. Additionally, the rolling moving average and the detected trend direction ($UP/DOWN$) are dynamically visualized.

Anomaly detection results are presented using a heatmap representation, allowing temporal anomaly patterns to be easily identified. Furthermore, system performance is continuously monitored through an automatically updated confusion matrix and F1-score indicators, providing immediate feedback on the effectiveness of the detection framework.

3 Results

The results of the proposed system are primarily observed through the real-time streaming dashboard. As illustrated in the dashboard screenshots, both the energy consumption stream and the

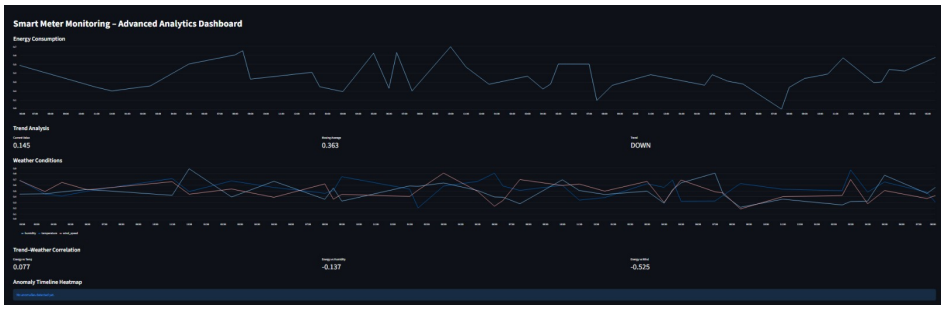

Figure 1: Results visualization

Table 1: Evaluation Metrics

Metric	**Value**
Accuracy	0.9567
Precision	0.3333
Recall	0.0833
F1-score	0.1333

weather data stream are visualized simultaneously in real time. This enables continuous monitoring of energy usage patterns alongside contextual environmental factors such as temperature and humidity.

In addition to raw data streams, derived analytics outputs are also displayed on the dashboard. These include the rolling moving average, the detected trend direction, the computed STL residual Z-score, and the anomaly detection results. Detected anomalies are further visualized using a heatmap representation, allowing temporal clustering and anomaly density to be easily interpreted. Moreover, model performance metrics, including precision, recall, F1-score, and the confusion matrix, are updated dynamically as new data points are processed.

While the system successfully achieves real-time data ingestion, processing, and visualization, the quantitative performance of the anomaly detection model is observed to be relatively low. As shown in the evaluation results, performance metrics such as precision, recall, and F1-score converge toward values close to zero. This indicates that the model struggles to accurately identify true anomalies within the dataset.

This outcome suggests that, although the proposed hybrid framework is structurally sound and operationally stable, the anomaly detection performance is limited under the current experimental conditions. The low performance can be attributed to factors such as the extreme class imbalance in the dataset, the scarcity of labeled anomaly instances, and the inherent difficulty of distinguishing subtle anomalies from normal consumption variations driven by weather effects.

Despite these limitations, the results demonstrate that the system is capable of executing end-to-end real-time analytics, integrating multiple data streams, and providing transparent visibility into both model decisions and performance metrics. These findings highlight the importance of evaluating not only system functionality but also model effectiveness in realistic streaming scenarios.

4 Discussion

The results demonstrate that the proposed system successfully establishes a robust real-time analytics environment. The Kafka-based pipeline, supported by strict schema enforcement and efficient data serialization, effectively manages the integration of energy consumption and weather data streams. Consequently, the visualization components, including rolling trends and raw data flows, operate without noticeable latency, confirming the architectural stability and the validity of the real-time streaming pipeline.

However, a clear disparity is observed between engineering reliability and analytical performance. While the data pipeline operates seamlessly, the hybrid anomaly detection framework achieves limited predictive performance, as reflected by low precision, recall, and F1-score values. These limitations are primarily attributed to the rigidity of fixed window sizes and static thresholds, which fail to adapt to dynamic consumption patterns. Furthermore, the extreme class imbalance and the scarcity of labeled anomaly instances significantly hinder the model's ability to identify outliers. These findings indicate that while the real-time infrastructure is technically sound, future implementations require more adaptive and expressive artificial intelligence models to capture complex and evolving behavioral shifts.

5 Conclusion

This study presents a scalable and modular real-time analytics pipeline for smart meter data, integrating Kafka-based streaming, JSON schema-enforced data exchange, and a hybrid online anomaly detection framework combining River-based Half-Space Trees with STL-based statistical validation, the implementation demonstrates the feasibility of translating complex data mining tasks into a continuous stream processing paradigm. Consequently, this architecture validates that robust, context-aware decision-making can be executed in near real-time, marking a significant step towards proactive energy management systems utilizing strictly typed data streams.

References

[1] M. Bessec and J. Fouquau, "The non-linear link between electricity consumption and temperature in europe: A threshold panel approach," *Energy Economics*, vol. 30, no. 5, pp. 2705–2721, 2008.

[2] Ziya07, "Smart meter electricity consumption dataset," https://www.kaggle.com/datasets/ziya07/smart-meter-electricity-consumption-dataset, 2023.

[3] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning, "Stl: A seasonal-trend decomposition procedure based on loess," *Journal of Official Statistics*, vol. 6, no. 1, pp. 3–73, 1990.