Introduction and Overview of Anomaly Detection Methods Research

Purpose

This memo introduces the Machine Learning Algorithm Research team and summarizes our exploration of advanced anomaly detection methods applied to time series data, as part of our ongoing effort to enhance data-driven monitoring and predictive maintenance capabilities.

Overview

During this cycle, the research team focused on evaluating and comparing four machine learning algorithms for anomaly detection in complex time series scenarios, particularly in multivariate sensor data.

Explored Methods

1. Z-Score (Statistical Baseline)

- Simple yet effective for univariate data.
- Highlights point anomalies based on standard deviation thresholds.

Isolation Forest

- Tree-based ensemble method that isolates anomalies through recursive partitioning.
- Effective for both univariate and multivariate data without assuming data distribution.

One-Class SVM

- Kernel-based method learning the boundary of normal data.
- Sensitive to parameter tuning and computationally intensive for large datasets.

LSTM (Long Short-Term Memory)

- Recurrent neural network model capturing temporal dependencies.
- Applied both as predictive and autoencoder-based anomaly detector.
- Particularly powerful for detecting complex temporal anomalies and patterns missed by traditional methods.

Next Steps

Building upon the current research, in the future, the team can focus on refining anomaly detection strategies by:

Algorithm Selection:

Selecting the most appropriate anomaly detection method based on specific data characteristics, operational requirements, and system complexity (e.g., using lightweight statistical methods for simple cases, and LSTM or Isolation Forest for complex, multivariate time series).

Sensor-Level Strategy Optimization:

Assessing whether sensors should be monitored individually or jointly, based on their correlations and operational interdependencies. This will ensure that both single-point anomalies and multi-sensor correlated anomalies are effectively detected.

Time Window Optimization:

Incorporating time windowing techniques (e.g., rolling statistics, lag features) to improve the robustness of anomaly detection, enabling the detection of both sudden spikes and gradual drifts in sensor behavior.

These enhancements aim to create a more context-aware, adaptive, and robust anomaly detection framework, capable of supporting scalable deployment across different IoT and industrial monitoring scenarios.

Appendix:

Comparison of Anomaly Detection Methods for Time Series

Method	Type	Strengths	Weaknesses	Best Use Cases	Contributors	Github
Z-Score	Statistical	- Simple to implement - Fast computation	- Assumes normal distribution - Univariate only	Quick checks on single-sensor data	ALIREZA MONTAZERI DEVANSHI TYAGI	https://github.com/DataBytes-Organisation/Intelligent-IoT-Data-Management/tree/feature/zscore-anomaly-detection/algorithms/zscore_anomaly_detection
Isolation Forest	Tree-based Ensemble	- Handles high-dimensional data - No distribution assumption - Robust to outliers	- Ignores time dependencies - Sensitive to contamination parameter	Multivariate anomaly detection without temporal patterns	LI WAN ARNAV AHUJA	https://github.com/DataBytes-Organisation/Intelligent-IoT-Data-Management/tree/feature/isolation-forest/isolation_forest
One-Class SVM	Kernel-based	- Captures complex data boundaries - Can model nonlinear patterns	- High computational cost on large datasets - Parameter tuning sensitive	Anomaly detection on small or medium-sized structured datasets	JOY JAYESH PATEL MARIAM SULEMANA BANDA	https://github.com/DataBytes-Organisation/Intelligent-IoT-Data-Management/tree/feature/one-class-svm
LSTM	Neural Network (RNN)	- Captures temporal dependencies - Handles complex, sequential	- Requires more data - Longer training time - Less interpretable	Time series data with temporal dependencies and complex patterns (e.g., sensor fusion,	HAI NAM LE GEORGIA QUACH	https://github.com/DataBytes-Organisation/Intelligent-IoT-Data-Management/tree/georgiaquach-feature-visualization/LSTM

Method	Type	Strengths	Weaknesses	Best Use Cases	Contributors	Github
		patterns - Suitable for multivariate sequences		predictive maintenance)		