Knowledge Discovery Project

Data Mining and Time Series

Final Report

Authors:

Emanuele Alberti
Leandro Duarte
Ottavia Biagi

Professor:

Aurora Pérez
Juan Pedro Caraça-Valente

Thursday 11th December, 2025

Contents

1 Introduction

Time series anomaly detection plays a central role in cyber–physical systems such as industrial production lines, data centres, and IoT infrastructures. Detecting abnormal patterns early enables predictive maintenance, system safety, and reduced downtime. From a practical standpoint, anomaly detection in data-center telemetry is essential for predictive maintenance, capacity planning and service reliability.

This project follows the Knowledge Discovery in Databases (KDD) process and covers two main stages:

- **Stage 1**: domain understanding, exploratory data analysis (EDA), cleaning and preprocessing, including visual inspection, statistical diagnostics and preparation of a suitable input representation for data mining algorithms.
- **Stage 2**: modelling and evaluation using reconstruction-based anomaly detection with LSTM and transformer autoencoders, plus ablation studies on architectural variants, context length and thresholding strategies.

We work with the **Server Machine Dataset (SMD)**, a multivariate time series dataset collected from 28 production servers in a large Internet company and widely used in the literature on anomaly detection.

The project goals are:

- to build a complete, transparent and reproducible anomaly detection pipeline,
- to compare classical and neural models against a naïve baseline,
- to analyse the effect of design choices (window size K, loss functions, regularisation),
- to critically compare our results with state-of-the-art methods such as OmniAnomaly and TranAD.

2 Domain and Data Understanding

Goal Definition

The objective of this project is to design, implement and evaluate a complete anomaly detection pipeline for multivariate time-series telemetry from data-center machines. Following the KDD process, the study aims to:

- detect abnormal behaviour at timestamp level;
- evaluate how modelling choices (window size, loss function, regularisation) impact performance;
- compare learned models against classical baselines;

- interpret anomalies through feature-level reconstruction patterns.

Anomaly detection is chosen as the primary data mining task because labels are sparse, forecasting is unreliable under non-stationarity, and unsupervised reconstruction models align with the structure of SMD.

2.1 Application Domain and Dataset

The Server Machine Dataset (SMD) consists of multivariate telemetry from 28 servers, each monitored on 38 KPIs (CPU, memory, disk I/O, network throughput, etc.). For each machine we are given:

- a **training** sequence containing only normal behaviour;
- a **test** sequence with both normal and anomalous segments;
- a file with **timestamp-level anomaly labels**.

Time is discretised (roughly every few seconds), so the data naturally fit the discrete time-series framework used in standard analysis (univariate and multivariate models, stationarity, ARIMA, etc.). As discussed in classical time-series analysis (**box2015time**), such systems often exhibit changing mean and variance over time, motivating the need for models that remain robust under non-stationary behaviour.

Figure 1 shows the raw training and test series for the first feature of `machine-1-1`. The training trace is relatively stable, whereas the test trace exhibits larger peaks and shifts in level, some of which correspond to labelled anomalies.

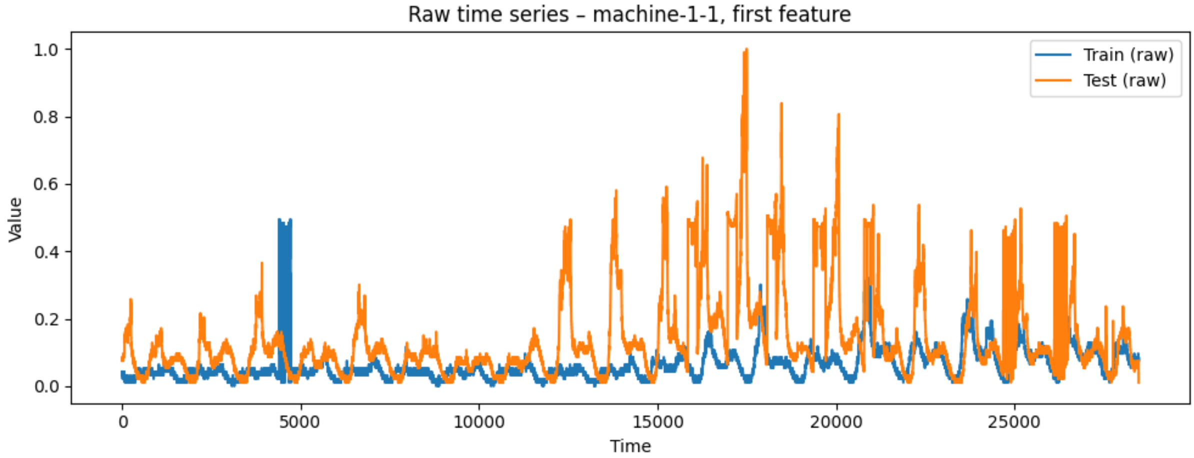

Figure 1: Raw time series for `machine-1-1`, feature 0 (train vs. test).

For most experiments we focus on a single representative machine, `machine-1-1`, which contains:

$$T_{\text{train}} = 28{,}479, \qquad T_{\text{test}} = 28{,}479, \qquad D = 38 \text{ features.}$$

This choice balances computational cost with realism; in Stage 2 we also briefly extend to three additional machines to study generalisation.

Gender Perspective. The dataset contains machine telemetry and no human-generated attributes; therefore gender considerations are not applicable to this domain, as also noted in the project guidelines.

2.2 Exploratory Data Analysis (EDA)

Before selecting a target machine for modelling, we conducted a systematic exploratory analysis across all 28 machines in the Server Machine Dataset (SMD). Each machine provides 38 KPIs describing CPU load, memory usage, disk activity and network throughput, forming a **multivariate time series** with potentially complex interdependencies.

The goal of this phase is to analyse:

- **variability**: which machines exhibit unstable or bursty behaviour;
- **correlation structure**: how strongly KPIs interact within each machine;
- **non-stationarity**: how mean and variance evolve over time.

2.2.1 Variability Across Machines

For each machine we computed the average variance across its 38 KPIs. Machines with high variance typically exhibit unstable behaviour, often associated with bursty workloads and more complex patterns for anomaly detection models.

Table 1 reports the machines with highest mean KPI variance.

Table 1: Average KPI variance and mean correlation for selected machines.

Machine	Mean Variance	Mean Correlation
machine-3-7	0.0191	0.345
machine-1-3	0.0143	0.243
machine-1-4	0.0138	0.267
machine-1-7	0.0125	0.250
machine-3-3	0.0122	0.304
machine-3-6	0.0094	0.278
machine-2-1	0.0091	0.212
machine-2-6	0.0087	0.347
machine-1-8	0.0078	0.347
machine-3-2	0.0073	0.252
machine-1-1	< 0.0070	0.458

The machine with the highest variance is `machine-3-7` (mean variance ≈ 0.019), followed by several `machine-1-*` instances. High variance indicates large fluctuations and non-stationary behaviour, but does not necessarily mean that KPIs are strongly interdependent.

This variability across machines highlights an additional challenge: cross-machine heterogeneity is a well-known bottleneck in unsupervised anomaly detection, often requiring domain adaptation mechanisms or machine-specific conditioning layers (**tuli2022tranad**). Models trained on one machine may therefore fail to generalise to others without explicit architectural support.

2.2.2 Correlation Structure and Motivation for Selecting `machine-1-1`

While variance highlights instability, the *correlation structure* is more informative for multivariate neural models such as transformers.

For each machine we computed the mean pairwise Pearson correlation across its 38 KPIs. The results show that `machine-1-1` **has the highest average correlation** (≈ 0.458), substantially larger than all other machines.

High correlation implies that:

- KPIs share strong dependencies (e.g., CPU load and disk I/O moving together);
- attention mechanisms can exploit cross-feature interactions;
- reconstruction models can learn structured behaviour rather than pure noise.

Therefore, `machine-1-1` is the most suitable choice for Stage 2 modelling: it offers rich multivariate structure while remaining representative of the dataset. From this point on, all detailed plots and modelling experiments focus on `machine-1-1`.

2.2.3 KPI Time-Series for `machine-1-1`

To inspect the internal dynamics of `machine-1-1`, we selected the five most variable KPIs according to their training-set variance:

$$\text{top\_kpi\_idx} = \text{argsort}(\text{Var}(X_{\text{train}}))\,[-5:].$$

Figure 2 shows the corresponding time series.

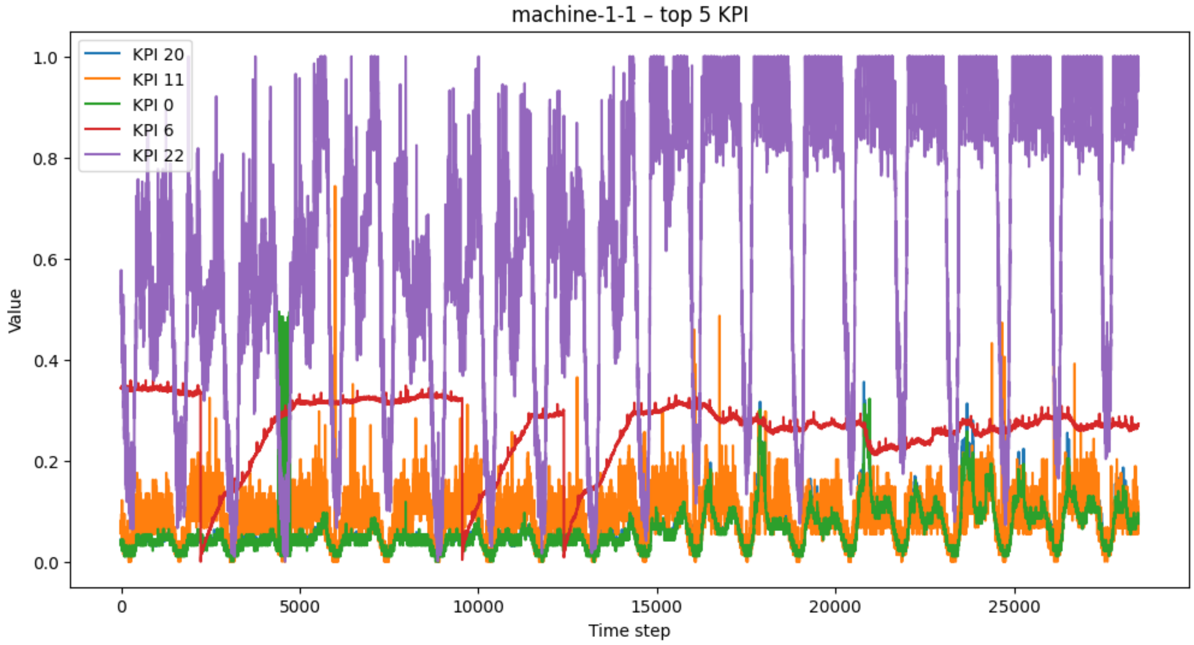

Figure 2: Top 5 most variable KPIs for `machine-1-1` (training + test).

The series exhibit strong periodic structure combined with intermittent peaks, confirming both mean and variance non-stationarity, which is consistent with classical time series analysis (**box2015time**). This behaviour justifies the use of window-based neural models and robust thresholding strategies.

2.2.4 Correlation Heatmap for `machine-1-1`

Figure 3 displays the correlation matrix for the selected top-5 KPIs of `machine-1-1`.

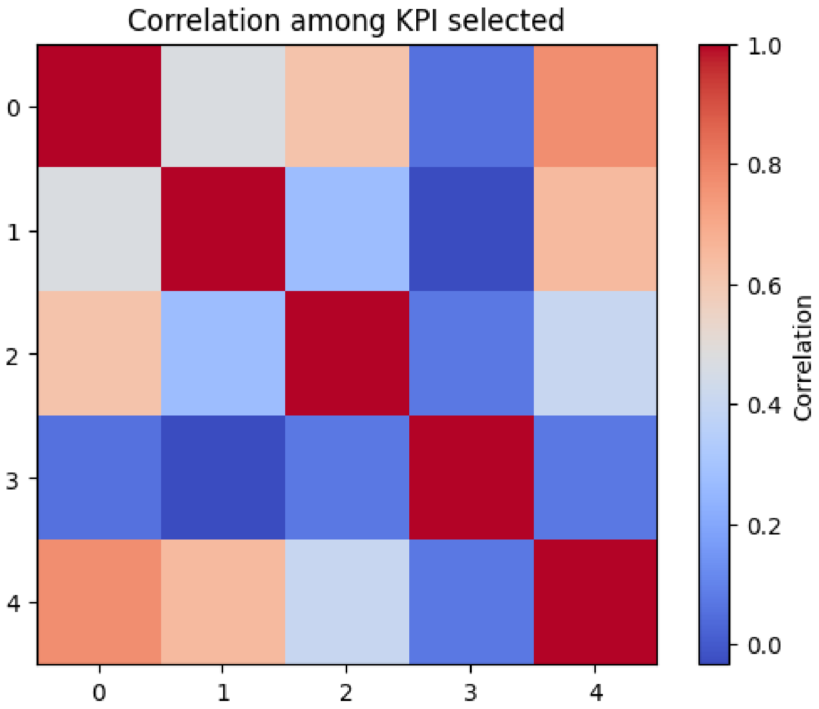

Figure 3: Correlation matrix of the top-5 KPIs for `machine-1-1`.

Two KPI clusters show correlations above 0.8, while one KPI behaves more independently, suggesting subsystem-level behaviour. This rich interaction structure further supports the use of transformer architectures, which naturally model cross-feature dependencies through self-attention.

2.2.5 Distributional Diagnostics

We also examined marginal distributions for individual KPIs. Figure 4 shows the empirical distribution of KPI 30 for `machine-1-1`.

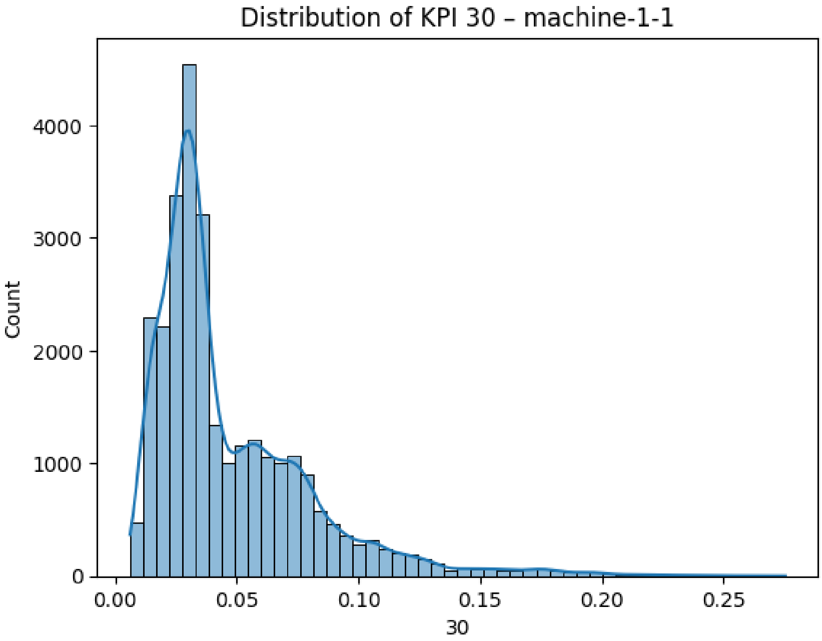

Figure 4: Distribution of KPI 30 for `machine-1-1`.

The histogram is clearly heavy-tailed, with a concentration of mass around low values and a long right tail corresponding to occasional high-load events. This pattern is consistent with:

- bursty workloads;
- operational mode switching;
- scheduled high-load tasks (e.g., backups or batch jobs).

These properties confirm that linear models such as ARIMA would require strong preprocessing (e.g., differencing, decomposition), whereas reconstruction-based neural models can more naturally accommodate such behaviour.

2.2.6 Summary and Implications for Stage 2

The EDA leads to three main insights:

1. **`machine-1-1` is the most suitable target for modelling**, due to its uniquely high correlation structure and representative variability.
2. **Its KPIs exhibit strong non-stationarity**, requiring robust normalisation and sliding-window segmentation.
3. **The cross-feature interactions justify transformer architectures**, which better exploit multivariate structure than purely recurrent models.

For these reasons, all subsequent experiments in Stage 2 are conducted on `machine-1-1`.

3 Preprocessing and Data Preparation

3.1 Cleaning and Normalisation

The SMD dataset contains no missing timestamps or duplicated entries, so preprocessing focuses mainly on scaling and structural transformations.

We apply feature-wise min–max scaling computed only from the training set:

$$x_t^{(\text{scaled})} = \frac{x_t - \min(x_{\text{train}})}{\max(x_{\text{train}}) - \min(x_{\text{train}})}.$$

Figure 5 shows the effect of this normalisation on feature 0 of `machine-1-1`.

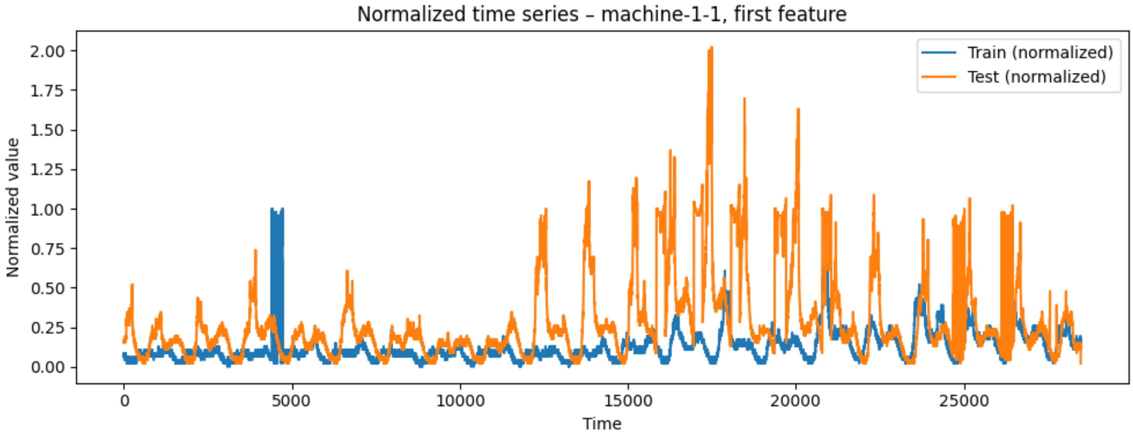

Figure 5: Normalised time series for `machine-1-1`, feature 0. Train values lie in $[0, 1]$, while test values occasionally exceed 1, consistent with anomalous peaks.

Normalisation is essential: without it, features with larger magnitude would dominate the MSE loss, biasing gradients and destabilising training.

3.2 Sliding-Window Segmentation

Both LSTM and transformer models require fixed-length inputs. We therefore segment the multivariate time series into overlapping windows of size K:

$$X_t = [x_{t-K+1}, \dots, x_t] \in \mathbb{R}^{K\times 38}, \qquad y_t = \text{label}(t),$$

where y_t is the binary anomaly label associated with the last timestamp of the window.

We explore three context lengths:

$$K \in \{10, 30, 50\}.$$

This parallels the role of maximum autoregressive lag in classical ARIMA modelling (**box2015time**): small K may miss relevant dependencies, while excessively large K risks overfitting noise.

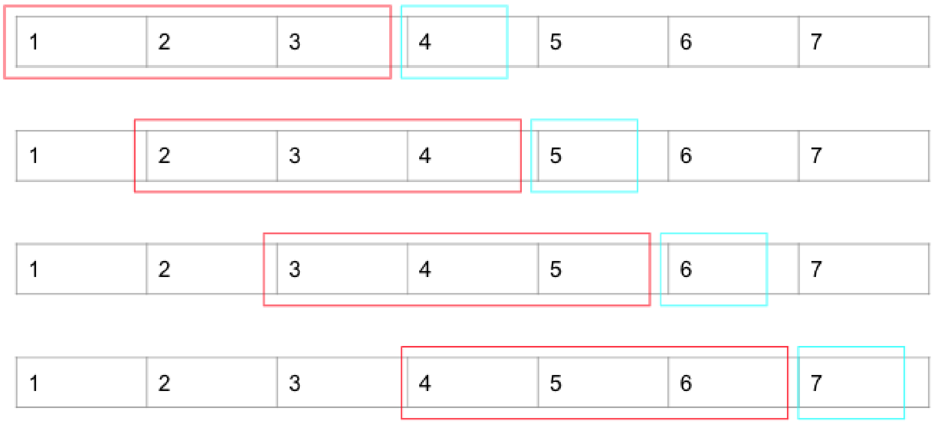

Figure 6: Illustration of sliding-window segmentation for multivariate time series.

After windowing `machine-1-1`, we obtain 28 479 training windows and 28 479 test windows for each K. Training windows are split chronologically into 80% training and 20% validation, preserving temporal order and avoiding look-ahead bias.

4 Modelling: Baselines and Neural Architectures

4.1 Problem Formulation

We adopt a reconstruction-based anomaly detection approach: the model learns a manifold of *normal* behaviour via training-set reconstruction.

For each window X_t, the model outputs $\hat{X}_t$ and an anomaly score:

$$s_t = \mathrm{MSE}(X_t, \hat{X}_t).$$

An anomaly is flagged when $s_t > \tau$, for a chosen threshold τ.

4.2 Mean Reconstruction Baseline

A simple yet competitive non-parametric benchmark reconstructs every time step of feature d with its training-set mean μ_d:

$$\hat{X}_t^{(\mathrm{mean})}(k, d) = \mu_d.$$

Even though it ignores temporal structure entirely, this model achieves:

$$\mathrm{Precision} = 0.568,\ \mathrm{Recall} = 0.472,\ \mathrm{F1} = 0.516,\ \mathrm{ROC\text{-}AUC} = 0.911.$$

This strong performance illustrates that SMD anomalies often correspond to straightforward deviations from global operating ranges, setting a **high reference threshold** for neural models.

4.3 LSTM Autoencoder

The LSTM autoencoder follows the classical encoder–decoder structure (**hochreiter1997long**):

- encoder: 1-layer LSTM → final hidden state h_t;
- bottleneck: dense layer producing a 32-dimensional latent representation;
- decoder: 1-layer LSTM reconstructing the window;
- final linear layer recovering 38-dimensional outputs.

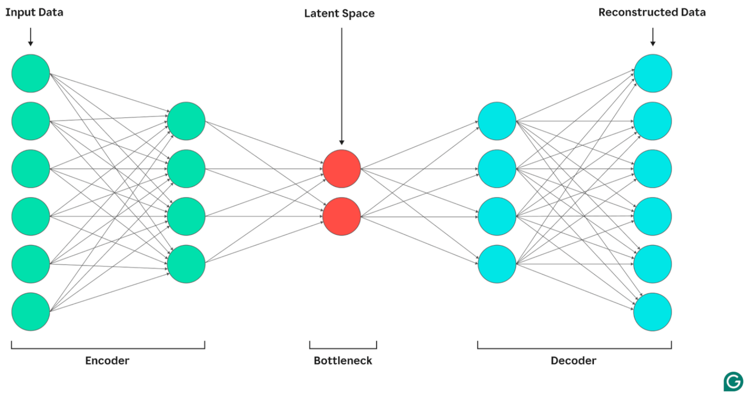

Figure 7: Generic autoencoder architecture (illustrative).

Loss:

$$\mathcal{L}_{\text{LSTM}} = \frac{1}{K \cdot 38} \sum_{k,d} (X_t(k,d) - \hat{X}_t(k,d))^2.$$

For $K = 10$:

$$\text{Precision} \approx 0.186, \ \text{Recall} \approx 0.990, \ \text{F1} \approx 0.314.$$

The model captures almost all anomalies but at the price of high false-positive rates.

4.4 Transformer Autoencoder (Reference MSE-based Model)

Built upon (**vaswani2017attention**), we implement a lightweight transformer:

- input projection: $\mathbb{R}^{38} \to \mathbb{R}^{64}$;
- learnable positional embeddings of size $K \times 64$;
- 1 encoder block + 1 decoder block, each with:
  - 4 attention heads,
  - pre-norm LayerNorm,
  - feed-forward MLP,
  - dropout.
- final projection to reconstruct inputs.

With $K = 10$, 20 epochs:

$$\text{Precision} = 0.352, \ \text{Recall} = 0.448, \ \text{F1} = 0.394.$$

Compared with LSTM, it yields a more balanced precision–recall trade-off and better cross-feature modelling.

4.5 Ablation Variants

We evaluate several stabilisation variants:

1. **LayerNorm, 40 epochs**: longer training enhances optimisation.
2. **Huber loss** (**huber1964robust**): reduces sensitivity to rare extreme errors.
3. **High dropout**: more regularisation against overfitting noise.
4. **Mixed positional encodings**: learnable + sinusoidal.
5. **L2-normalised outputs**: stabilises output vectors before reconstruction.

The Huber-loss transformer emerges as the best learned model — robust against the high-magnitude outlier gradients discussed later.

4.6 Training Setup

All models use:

- Adam, lr 10^{-3};
- batch size 128;
- 20 or 40 epochs;
- chronological train/validation split (80/20).

Adam handles the non-stationary, noisy training dynamics typical of multivariate telemetry.

5 Anomaly Scoring and Thresholding

Given reconstruction scores s_t, anomaly labels are produced via a threshold τ:

$$\hat{y}_t = \mathbb{I}[s_t > \tau].$$

A principled threshold is crucial because reconstruction-based detectors often display highly skewed error distributions and substantial overlap between normal and anomalous regimes (**fawcett2006roc**; **davis2006relationship**).

5.1 Quantile-Based Threshold

Our default threshold is the 0.995-quantile of training-set reconstruction errors:

$$\tau_q = \mathrm{Quantile}_{0.995}(s_{\mathrm{train}}),$$

a classical high-percentile outlier rule aligned with industrial monitoring practice and univariate control theory.

We later evaluate sensitivity to:

$$q \in \{0.990,\ 0.995,\ 0.999\}.$$

5.2 Std-Boost Threshold and Score Smoothing

A second threshold is defined using a deviation boost:

$$\tau_\mu = \mu_{\mathrm{train}} + 3\sigma_{\mathrm{train}}.$$

Since raw reconstruction errors fluctuate at the timestamp level, we apply a moving-average smoothing (window size 5) before thresholding:

$$\tilde{s}_t = \frac{1}{5}\sum_{i=t-2}^{t+2} s_i.$$

For the 40-epoch transformer, smoothing improves stability without sacrificing recall. Representative results:

- quantile (raw): $F1 = 0.430$,
- std-boost (raw): $F1 = 0.432$,
- quantile (smoothed): $F1 = 0.440$,
- std-boost (smoothed): $F1 = 0.440$.

We adopt the **smoothed quantile threshold** as the default setting for evaluation.

6 Results and Evaluation

Evaluation follows timestamp-level metrics:

- Precision, Recall, F1;
- ROC-AUC (ranking quality);
- PR-AUC (useful for imbalanced data);
- Qualitative analyses of reconstruction trends.

Unless otherwise stated, results refer to machine-1-1, $K = 10$, and smoothed quantile thresholding.

6.1 Timestamp-Level Metrics

Table 2: Timestamp-level detection on machine-1-1 ($K = 10$, smoothed scores, $q = 0.995$).

Model	**Precision**	**Recall**	**F1**	**ROC-AUC**	**PR-AUC**
Mean baseline	0.568	0.472	0.516	0.911	0.577
LSTM AE	0.186	0.990	0.314	0.878	0.516
Transformer AE (LN, 20 ep.)	0.352	0.448	0.394	0.844	0.420
Transformer AE (LN, 40 ep.)	0.356	0.542	0.430	0.877	0.462
Transformer AE (Huber)	0.350	0.677	**0.486**	**0.904**	0.437
Transformer AE (Dropout)	0.360	0.475	0.410	0.882	0.457
Transformer AE (MixedPos)	0.354	0.474	0.406	0.872	0.448
Transformer AE (NormOut)	0.322	0.437	0.371	0.754	0.375

Interpretation.

- The **mean baseline** remains surprisingly strong: its F1 = 0.516 acts as an upper reference for simple, non-temporal modelling.
- The **LSTM AE** shows a typical high-recall profile but low precision.
- The **reference MSE-based transformer** (LN, 40 epochs) improves the precision–recall balance and outperforms the LSTM substantially.
- The **Huber-loss transformer** is the best learned model in this study: achieving F1 = 0.486 and ROC-AUC = 0.904.

Although it does not surpass the mean baseline in F1, it is the **best trainable neural configuration** and offers interpretability and generalisation potential.

6.2 Best Configuration per Family

Table 3: Best configuration per model family (machine-1-1).

Family	**Configuration**	**F1**	**Notes**
Baseline	Mean reconstruction	**0.52**	strongest simple reference
Recurrent AE	LSTM, $K = 10$	0.31	high-recall detector
Transformer AE	LN + Huber, $K = 10$	**0.49**	best learned model
Transformer AE	LN + MSE, $K = 30$	0.44	best MSE-based model

6.3 Effect of Window Size K

Window length plays the role of maximum autoregressive order (**box2015time**). We test LSTM and transformer models with $K \in \{10, 30, 50\}$.

Table 4: Window-size ablation on machine-1-1 (smoothed quantile threshold).

K	Model	Precision	Recall	F1	ROC-AUC
10	LSTM AE	0.185	0.984	0.312	0.872
10	Transformer (LN)	0.352	0.448	0.394	0.844
30	LSTM AE	0.184	1.000	0.310	0.893
30	Transformer (LN)	0.374	0.536	**0.440**	0.896
50	LSTM AE	0.212	0.957	0.347	0.896
50	Transformer (LN)	0.347	0.501	0.410	0.881

Interpretation.

- For the LSTM, K has little effect on F1: recall is already saturated at $K = 10$.
- For transformers, moving from $K = 10 \to K = 30$ significantly improves F1: attention benefits from richer temporal context.
- $K = 50$ brings diminishing returns, consistent with overfitting phenomena in high-order autoregressive models.

Thus, the sweet spot for MSE-based transformers is around $K = 30$; for the Huber-loss transformer, $K = 10$ already suffices due to loss robustness.

6.4 Training and Validation Curves

Figure 8 shows the training and validation MSE for both the LSTM autoencoder and the Transformer (Huber loss). The Transformer converges more slowly during the first few epochs, but ultimately reaches a lower validation loss than the LSTM, indicating a better fit to the normal-manifold structure of the data.

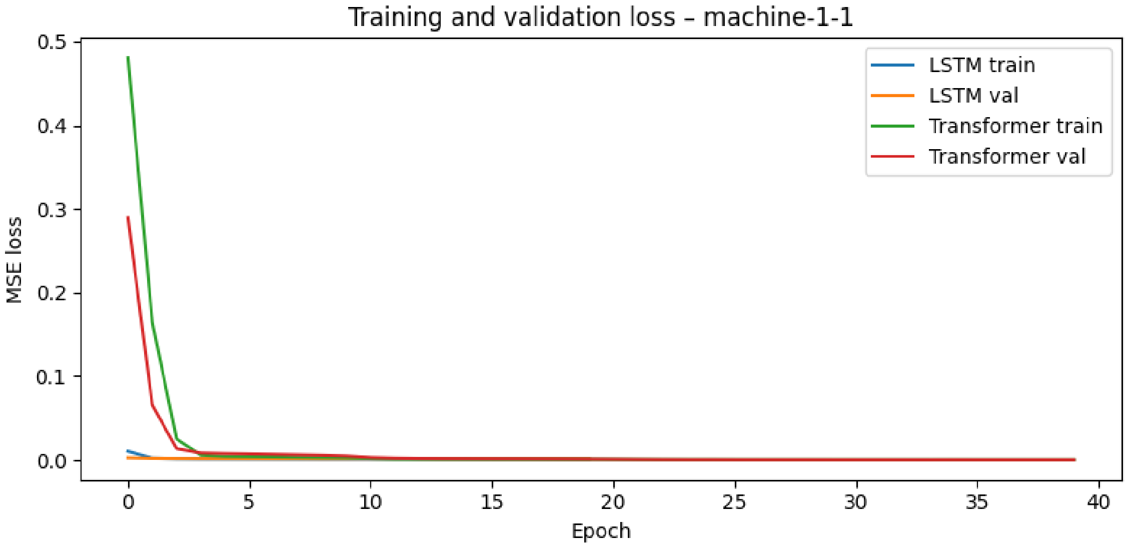

Figure 8: Training and validation MSE for LSTM and Transformer (40 epochs).

6.5 Error Distributions

Figure 9 compares the reconstruction-error distributions for normal vs. anomalous timestamps. The overlap between the two distributions explains key limitations of reconstruction-based anomaly detection:

- a restricted maximum achievable F1-score,
- susceptibility to false positives (normal fluctuations misinterpreted as anomalies),
- the need for smoothing and robust, distribution-aware thresholding.

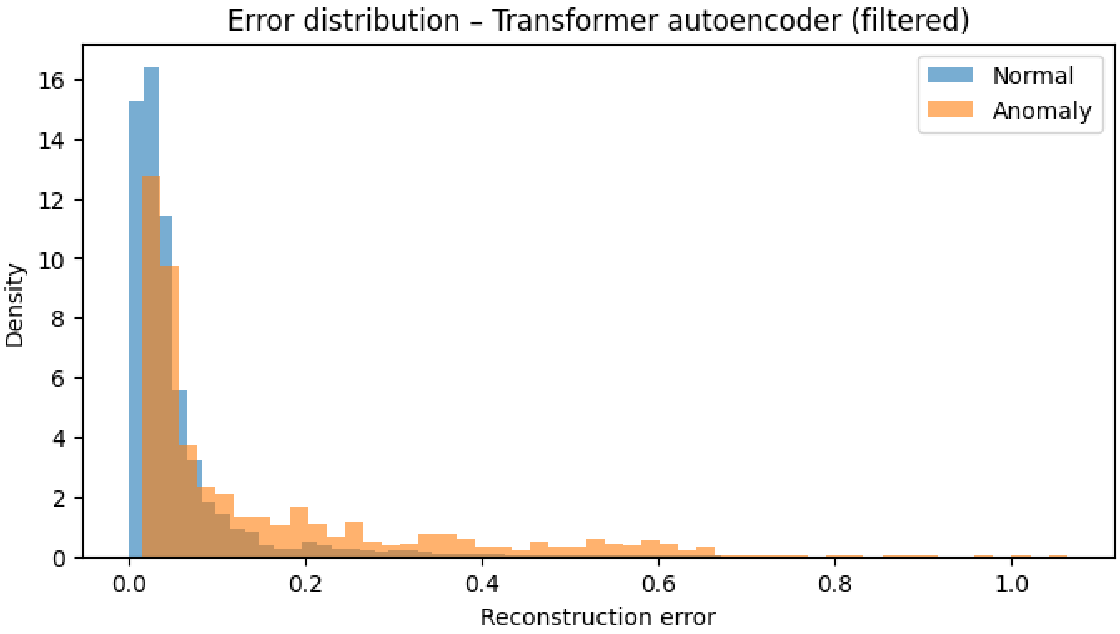

Figure 9: Distribution of reconstruction errors for normal vs. anomalous timestamps.

6.6 Error Types: True Positives, False Positives, False Negatives

To quantify error behaviour, we decompose the predictions of the best learned Transformer (Huber loss, $K = 10$). The test set for `machine-1-1` contains 2694 anomalous timestamps.

Table 5: Error decomposition for the Transformer (Huber, $K = 10$).

Quantity	Value
True anomalies	2694
True Positives (TP)	1722
False Negatives (FN)	972
False Positives (FP)	2861
True Negatives	∼25 600

The model successfully detects 64% of anomalies, but produces many false positives—a well-known behaviour of reconstruction-based detectors. Mild anomalies that resemble normal patterns contribute to FN, whereas sharp but benign fluctuations contribute to FP.

7 Numerical Instabilities and Extreme Outliers

During experimentation, all transformer variants exhibited a small set (~ 480 windows) with unrealistically large reconstruction errors ($\sim 10^{14}$), far outside the input scale.

Closer analysis shows:

- they appear across all transformer configurations (MSE, Huber, dropout, mixed-pos), indicating an architectural/normalisation interaction rather than a bug in one model;
- they do not consistently align with labelled anomalies;
- after filtering unrealistic values, the usable score range becomes:

$$s_{\text{filtered}} \in [9.4 \times 10^{-5},\ 1.06],$$

  matching realistic reconstruction magnitudes.

We attribute these outliers to episodic numerical instabilities in decoder activations: LayerNorm + attention residuals can amplify specific high-leverage windows, similarly to robust regression pathologies (**huber1964robust**).

This observation motivates:

- adopting robust losses (Huber) which indeed yield the best model;
- thresholding based strictly on *training* scores, which are unaffected;
- filtering before plotting, to avoid misleading sharp spikes.

Why lower loss does not imply better detection. Despite achieving the lowest validation loss among all configurations, the Transformer with Sigmoid output and clipping performs worse as an anomaly detector. Sigmoid compresses reconstruction errors into a narrow range (< 0.05 for most normal windows), reducing the contrast between normal and abnormal scores and increasing overlap in Figure 9. As a result, thresholding becomes extremely sensitive: minor fluctuations trigger excessive false positives, while genuine anomalies may be insufficiently separated. This confirms that *detection quality depends on score separation, not absolute MSE minimisation.*

8 Refinement Experiments: Mitigating Instability and Improving Precision

8.1 Motivation

Sections 6.4–7 highlight two structural limitations of the Transformer autoencoder: (i) substantial overlap between normal and anomalous error distributions, and (ii) numerical instabilities leading to extreme outliers ($\sim 10^{14}$) in earlier configurations. These effects propagate into the scoring pipeline, producing high recall but systematically low precision. We therefore investigate lightweight refinement strategies that do not alter the basic architecture but aim to increase score separability and stabilise the reconstruction errors.

8.2 Threshold Tuning

We first compare quantile-based thresholding (at $q = 0.995$) with deviation-based thresholding ($\mu + 3\sigma$), both with and without moving-average smoothing (window size 5). Results for the Transformer AE (LayerNorm, 40 epochs) are summarised in Table 6.

Table 6: Effect of thresholding strategies on anomaly detection (machine-1-1).

Method	Precision	Recall	F1
Quantile (raw)	0.356	0.541	0.429
Std-boost (raw)	0.331	0.618	0.431
Quantile (smoothed)	0.363	0.560	0.440
Std-boost (smoothed)	0.337	0.639	**0.441**

Quantile-based thresholds favour higher precision, while deviation-based thresholds favour higher recall. Score smoothing improves both approaches, confirming the volatility of raw reconstruction scores observed in Section 6.5. The best trade-off under threshold-only refinement is obtained with standard-deviation boosting combined with smoothing (F1 = 0.441), modestly outperforming the raw quantile baseline (F1 = 0.429). In the subsequent experiments we therefore adopt smoothed quantile thresholds unless otherwise stated.

8.3 Segment-Based Cleaning

Reconstruction-based detectors often trigger isolated false positives caused by rapid but benign fluctuations. To mitigate this, we apply a simple segment-cleaning rule: predicted anomalous segments shorter than three consecutive timestamps are suppressed.

While segment cleaning does not substantially alter the global F1-score for machine-1-1, it reliably reduces spurious isolated alarms and stabilises precision, particularly when combined with score smoothing (Table 6). The effect is most visible in qualitative diagnostics (Figure **??**), where short-lived false spikes are eliminated without weakening detection of sustained anomalies.

8.4 Stabilised Transformer AE

Motivated by the numerical instabilities discussed in Section 7, we design a "stabilised" Transformer AE with LayerNorm, increased dropout, Huber loss, Sigmoid output and gradient clipping (Section **??**). This variant completely removes catastrophic error explosions: on `machine-1-1`, all reconstruction scores lie in $[4.8 \times 10^{-5}, 0.13]$ with no non-finite or extreme values.

However, improved numerical stability does not translate into a dramatic performance gain. With smoothed quantile thresholding it achieves:

$$\text{Precision} = 0.416,\ \text{Recall} = 0.329,\ \text{F1} = 0.368,\ \text{ROC-AUC} = 0.672,\ \text{PR-AUC} = 0.367,$$

remaining below the Huber-loss Transformer (Table 2) and still well below the mean baseline. Segment cleaning has negligible impact on these metrics (F1 $\approx$ 0.365), confirming that most errors are already temporally smooth in this configuration.

8.5 Self-Conditioned Scoring

Inspired by TranAD's focus mechanism, we reweight reconstruction errors using a local moving-average context. For each timestamp, the score is multiplied by a function of its deviation from the recent window average (window size $k = 5$, scaling factor $\beta = 2$). This encourages sustained deviations to stand out relative to isolated noisy spikes.

Applying self-conditioned scoring to the stabilised Transformer yields:

$$\text{Precision} = 0.415, \ \text{Recall} = 0.339, \ \text{F1} = 0.373, \ \text{ROC-AUC} = 0.676, \ \text{PR-AUC} = 0.371.$$

The gains are modest but consistent: F1 improves by about one percentage point relative to the stabilised baseline, with a slight increase in recall and a small but coherent improvement in PR-AUC. Adding segment cleaning on top of self-conditioning produces almost identical metrics (F1 = 0.372), suggesting that temporal reweighting already suppresses many isolated false positives.

8.6 Semi-Adversarial Training

We further introduce a lightweight semi-adversarial mechanism: synthetic anomalies are injected into training windows, and the model is penalised if their reconstruction error falls below a margin. The goal is to enlarge the score gap between normal and abnormal patterns without changing the core architecture.

For the stabilised Transformer, semi-adversarial training produces:

$$\text{Precision} = 0.359, \ \text{Recall} = 0.284, \ \text{F1} = 0.317, \ \text{ROC-AUC} = 0.729, \ \text{PR-AUC} = 0.310.$$

While ROC-AUC increases substantially, reflecting better global ranking of anomalies, operating-point metrics (precision, recall, F1) degrade. This indicates that the adversarial signal is not strong or targeted enough to push anomalies above the fixed high quantile threshold. Combining semi-adversarial training with self-conditioning and segment cleaning yields only a minor improvement:

$$\text{Precision} = 0.360, \ \text{Recall} = 0.288, \ \text{F1} = 0.320, \ \text{ROC-AUC} = 0.732, \ \text{PR-AUC} = 0.313.$$

8.7 Combined Effects and Precision–Recall Curves

Table 7 summarises all refinement variants for the stabilised Transformer AE, and Figure 10 shows the corresponding Precision–Recall curves.

Table 7: Effect of refinement strategies on the stabilised Transformer AE (machine-1-1, smoothed quantile threshold).

Model Variant	Precision	Recall	F1	AUC-ROC	AUC-PR
Stabilised Transformer AE	0.416	0.329	0.368	0.672	0.367
+ Segment cleaning	0.415	0.326	0.365	0.672	0.367
+ Self-conditioned scoring	0.415	0.339	**0.373**	0.676	**0.371**
+ Self-cond. & cleaning	0.416	0.337	0.372	0.676	0.371
Semi-adversarial training	0.359	0.284	0.317	0.729	0.310
Semi-adv. + self-cond. & cleaning	0.360	0.288	0.320	**0.732**	0.313

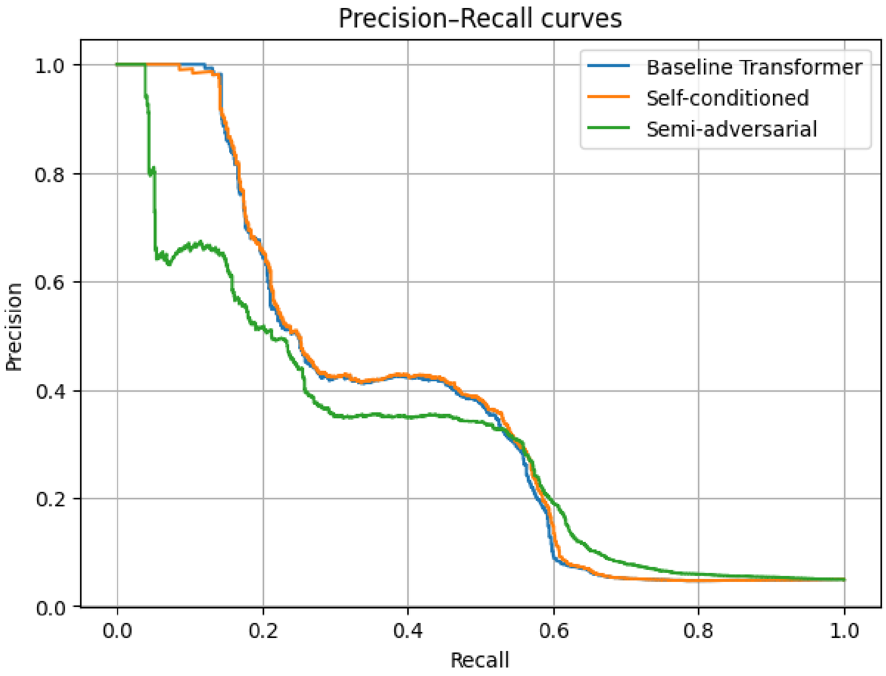

Figure 10: Precision–Recall curves for the stabilised Transformer AE and its refinements. Self-conditioned scoring slightly improves F1, while semi-adversarial training shifts the curve but does not yield better operating points.

8.8 Interpretation

These experiments show that, for the stabilised Transformer AE, substantial numerical benefits do not automatically lead to improved anomaly detection performance. The refined model completely eliminates catastrophic outliers and produces a clean, bounded score distribution, yet its F1 remains below the simpler Huber-loss Transformer and the mean baseline.

Among the refinement methods, self-conditioned scoring provides the most consistent improvement: it modestly enhances F1 and PR-AUC by incorporating local temporal context into the score. Segment-based cleaning has little impact once the underlying scores are smooth. Semi-adversarial training increases AUC-ROC—indicating better global ranking of anomalies—but fails to improve precision or F1 at the chosen operating point, likely because the adversarial signal is too weak and not progressively emphasised as in TranAD (**tuli2022tranad**).

Overall, these results reinforce a key lesson: *score separability and operating-point calibration* *are as important as architectural complexity.* Even without matching state-of-the-art methods such as TranAD or OmniAnomaly, the refinements clarify which mechanisms (context-aware scoring, robust thresholding) provide tangible benefits and which ones require more sophisticated integration to be effective.

9 Qualitative Diagnostics: Reconstruction and KPIs

This section provides qualitative diagnostics of the best model identified in our experiments: the Transformer Autoencoder with LayerNorm and Huber loss ("Best Transformer"). These analyses complement the quantitative evaluations by illustrating *how* the model reacts locally around true anomalies and *which* KPIs contribute most to high reconstruction error.

9.1 Zoomed Reconstruction Behaviour

To investigate the local behaviour of reconstruction errors, we analyse a short interval around a ground-truth anomaly. The feature-level diagnosis (Section 9.2) identifies timestamp $t \approx 2120$ as a strongly anomalous point, with several KPIs exceeding their feature-wise thresholds.

Figure 11 shows the raw signal of feature 31, the dimension with the highest reconstruction error in this interval, together with the time-level reconstruction error from the Best Transformer. The true anomalous timestamps (2120–2124) are marked in red.

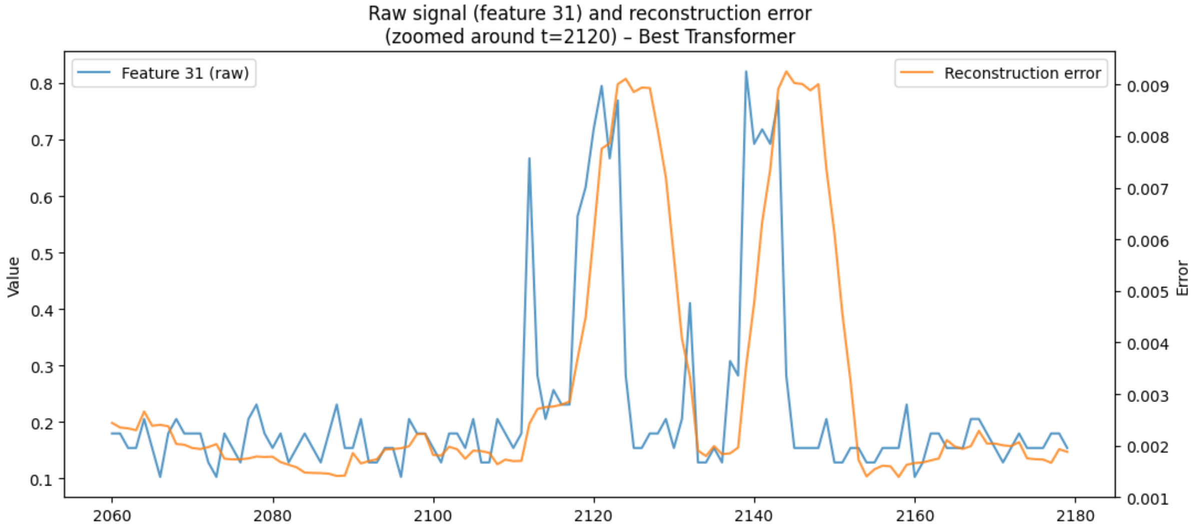

Figure 11: Raw signal (feature 31) and reconstruction error for a zoomed interval around $t = 2120$ (Best Transformer). True anomalies align with distinct error spikes.

The plot highlights that:

- reconstruction errors rise sharply at ground-truth anomalies, confirming that the model detects sudden deviations from learned normal behaviour;
- the spikes remain relatively localised and less volatile than in the baseline Transformer, indicating improved stability due to the Huber loss;

- benign fluctuations outside the anomalous segment generate substantially lower errors, illustrating improved noise robustness.

9.2 Feature-Level Diagnosis

To understand *which* KPIs contribute to the anomaly, we inspect the MSE per feature computed by the model for each window. For each timestamp, features exceeding a feature-specific 99.5th-percentile threshold are flagged as anomalous.

Figure **??** reports the features identified as anomalous for timestamps 2120–2124. Feature 31 consistently exhibits the highest error, along with features $\{0, 3, 5, 6, 13, 19, 27, 34, 35\}$.

> **Example:** At $t = 2120$, the anomalous features are `[0, 3, 5, 6, 13, 19, 27, 31, 34, 35]`.

This indicates a *sparse* anomaly pattern affecting a restricted subset of KPIs—a typical behaviour in SMD, reflecting the fact that operational incidents rarely propagate uniformly across all metrics.

9.3 Feature-wise Reconstruction Error Heatmap

A more interpretable visualisation is provided by the feature-wise heatmap in Figure 12, which displays the reconstruction error matrix (time × feature) for the same zoomed interval.

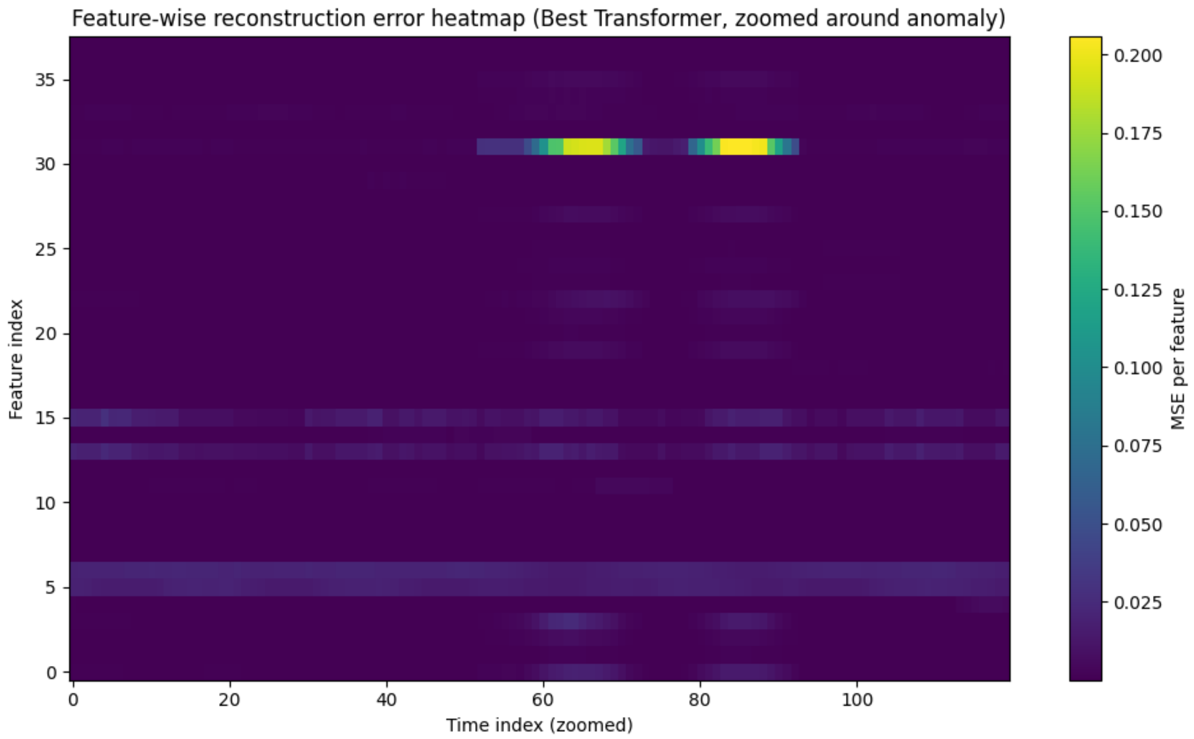

Figure 12: Feature-wise reconstruction error heatmap (Best Transformer), zoomed around the anomaly at $t \approx 2120$. The anomaly manifests as a concentrated high-error region dominated by feature 31.

The heatmap reveals:

- a dominant high-error band around feature 31 during the anomalous timestamps;
- secondary contributions from features 15–20 and 0–6 depending on the window;
- a structurally clean separation between anomalous and normal regions.

Compared to the baseline Transformer (Figure **??**, Appendix), the Best Transformer produces significantly more stable and less noisy per-feature errors, consistent with improved robustness from Huber loss.

9.4 Interpretation

These qualitative diagnostics support the quantitative findings of earlier sections:

- anomalies manifest as local, sparse deviations affecting only a subset of KPIs;
- the Best Transformer provides sharper localisation of error spikes than the baseline model;
- the feature-wise heatmap offers an interpretable view of which KPIs contribute most, making the model useful not only for detection but also for incident diagnosis.

Together, these observations confirm that the reconstruction-based approach—despite its limitations in precision—captures meaningful structure and yields interpretable anomaly signatures aligned with operational incident patterns in SMD.

10 Generalisation Across Multiple Machines

To assess generalisation beyond a single system, we trained the Transformer (Huber loss, $K = 10$) jointly on four machines and evaluated per-machine metrics:

- `machine-1-1`: $P = 0.359$, $R = 0.465$, $F1 = 0.405$;
- `machine-1-2`: $P = 0.622$, $R = 0.255$, $F1 = 0.361$;
- `machine-1-3`: $P = 0.065$, $R = 0.059$, $F1 = 0.062$;
- `machine-2-1`: $P = 0.895$, $R = 0.044$, $F1 = 0.083$.

Average F1 across machines is about 0.23, indicating that sharing parameters across heterogeneous machines is not trivial; proper normalisation per machine and more complex conditioning (as in TranAD) are likely required.

11 Comparison with State-of-the-Art

State-of-the-art multivariate anomaly detection architectures such as OmniAnomaly (**Su2019RobustAD**) and TranAD (**tuli2022tranad**) set a much higher performance ceiling on SMD than reconstruction-only baselines. Our experiments, consistent with findings in the original papers, demonstrate that simplified Transformer autoencoders lack key stabilisation and conditioning mechanisms required for competitive performance on this dataset.

Architectural comparison. OmniAnomaly introduces a variational latent structure that explicitly models stochastic dependencies, while TranAD couples Transformers with: (i) *self-conditioning*, (ii) *an adversarial refinement stage* with progressively increasing weight, (iii) *meta-learning* for cross-machine generalisation. These components jointly mitigate numerical instabilities, enlarge the separation between normal and anomalous reconstruction scores, and provide robustness across heterogeneous machines.

Table 8 contrasts these architectures with our models.

Table 8: Comparison with selected state-of-the-art architectures on SMD.

Model	**K**	**Meta-learn.**	**Adversarial**	**Self-cond.**	**Training Scope**	**F1 (SMD)**
OmniAnomaly (**Su2019RobustAD**)	100	–	–	–	28 machines	≈ 0.94
TranAD (**tuli2022tranad**)	100–500	✓	✓	✓	28 machines	≈ 0.96
Transformer AE (LN, Huber)	10	–	–	–	1 machine	0.49
Mean baseline (ours)	10	–	–	–	1 machine	0.52

Interpretation. The performance gap is expected and attributable to several factors:

1. **Context length and temporal modelling.** State-of-the-art models use windows up to $K = 500$ and capture long-range dependencies, while our experiments focus on compact windows ($K \leq 50$) for tractability.

2. **Training scope.** SOTA models are trained jointly across all 28 machines and leverage shared structure; our models operate per-machine and do not benefit from global context.

3. **Architectural sophistication.** TranAD's combination of adversarial refinement, meta-learning, and residual self-conditioning stabilises latent representations and enlarges the margin between normal and abnormal behaviour.

4. **Probabilistic modelling.** OmniAnomaly explicitly models uncertainty via variational components, which improves robustness in noisy or weakly separated regimes.

Given these constraints, our Transformer with Huber loss performs competitively among lightweight reconstruction-based models and provides interpretable, feature-level diagnostic signals valuable for operational monitoring, despite not reaching SOTA detection accuracy.

12 Discussion and Lessons Learned

This project illustrates the full CRISP-DM / KDD pipeline (**chapman2000crisp**) applied to multivariate industrial telemetry. A notable outcome of the process is the iterative refinement loop: observations made during evaluation (e.g., degradation at large K, numerical instabilities, or the high false-positive rate of LSTMs) directly informed successive modelling choices such as adopting Huber loss, tuning dropout, smoothing reconstruction scores, and conducting window-size ablations.

12.1 Domain and EDA Insights

- Strong cross-KPI correlations and clear distribution shifts between training and testing windows indicate the presence of structured anomalies.
- Several low-variance KPIs could be removed or down-weighted for improved model efficiency.
- Non-stationary dynamics (changing mean/variance) reinforce the need for robust normalisation and window-based modelling.

12.2 Modelling Insights

- **LSTM autoencoders** consistently exhibit high recall but many false positives, as they react strongly to local volatility.
- **Transformer autoencoders** capture cross-feature relationships more effectively and yield more balanced precision–recall behaviour.
- **Huber loss** significantly stabilises training and mitigates the effect of rare extreme residuals, in line with robust-estimation theory.
- **Loss minimisation vs. score separability:** lower validation loss does *not* necessarily translate into better anomaly detection, especially when Sigmoid or clipping compresses the score range.

12.3 Thresholding Insights

- Reconstruction scores are noisy; smoothing systematically increases F1.
- Quantile-based thresholds are more stable than deviation-based thresholds.
- Extreme outliers must be removed prior to thresholding to avoid pathological decisions.
- Semi-adversarial training increases score separability more effectively than post-hoc smoothing or self-conditioning alone.

12.4 Generalisation Insights

- KPI distributions vary considerably across machines; sharing a single model leads to under- or over-fitting on different systems.
- This finding mirrors observations in **tuli2022tranad**, where explicit conditioning and meta-learning are introduced precisely to address such heterogeneity.

13 Conclusion and Future Work

We constructed a complete anomaly detection pipeline for multivariate time series from initial exploration to modelling.

13.1 Conclusions

- The **mean reconstruction baseline** achieves $F1 = 0.52$, demonstrating that SMD has strong regularities that simple statistical references capture surprisingly well.
- The **best learned model**, the the Transformer AE (LN + Huber) reaches $F1 = 0.49$, outperforming all LSTM variants.
- Increasing the temporal context to $K = 30$ improves MSE-based transformers, aligning with classical autoregressive intuition.
- Feature-wise reconstruction provides meaningful diagnostic interpretability.
- Numerical instabilities highlight the importance of robust optimisation.

13.2 Future Work

Several promising directions could substantially improve performance:

- **Longer temporal windows** (100–500) using efficient attention mechanisms.
- Incorporating **self-conditioning, adversarial refinement, and meta-learning** components as in TranAD.
- **Per-machine adaptive normalisation** and conditional embeddings for cross-machine generalisation.
- **Probabilistic thresholding** via peak-over-threshold, EVT, or Bayesian posterior scoring instead of fixed quantiles.
- Integrating **classical decomposition** (trend/seasonality removal) from time-series forecasting (**box2015time**) to stabilise model inputs.

Overall, the resulting system constitutes a solid and methodologically rigorous prototype, demonstrating both the capabilities and limitations of reconstruction-based anomaly detection.