Weakly-supervised Multi-sensor Anomaly Detection with Time-series Foundation Models

Zelin He[1]*, **Matthew Reimherr**[1,2], **Sarah Alnegheimish**[3]*, **Akash Chandrayan**[2]
[1]Department of Statistics, Pennsylvania State University
[2]Reliability Maintenance Engineering, Amazon
[3]Electrical Engineering and Computer Science, MIT

Abstract

Anomaly detection in industrial sensor data is challenging as sensor readings are frequently affected by routine operations, leading to sudden changes that may not indicate actual issues. This makes it difficult to distinguish between normal and anomalous behavior. With a few expert-labeled anomalies, we aim to leverage these sparse labels to improve sensor anomaly detection. Besides the issue of limited labels, since these labels are collected from heterogeneous sensors across different machines, we need a framework that can learn general anomaly patterns across sensors and then adapt to the unique behavior of each individual sensor. In this paper, we propose a weakly-supervised multi-sensor anomaly detection (*WMAD*) framework that leverages deep networks, including foundation models, to construct a data-enclosing hypersphere, effectively separating normal from anomalous time windows. By incorporating two-level importance sampling and meta-learning, *WMAD* effectively handles both label sparsity and sensor heterogeneity. The experiment shows that our method outperforms state-of-art competing methods on both a large proprietary industrial amperage dataset spanning over 700K hours of time-series data from Amazon and a public telemetry dataset.

1 Introduction

One of the key challenges in industrial sensor anomaly detection is the presence of irregular sensor readings, as sensor readings are frequently affected by machine operating conditions and routine maintenance events. These external factors often lead to sudden fluctuations in sensor data—such as drops or spikes—that may not necessarily indicate actual issues with equipment. At the same time, during maintenance activities, technicians create a limited set of high-quality, manually labeled anomalies on some sensors. How to effectively leverage these sparse labeled anomalies to improve anomaly detection across different sensors is the key problem we aim to solve. Since sensors deployed across machines and under varying conditions produce diverse patterns, models trained from scratch may struggle to learn robust representations. To address this, we explore using time-series foundation models to enhance anomaly detection performance.

Prior to this work, there were several lines of research exploring using limited labeled anomalies to enhance detection. One line of research focuses on mapping input data into a representation space that separates normal from anomalous samples [1, 2]. Another direction involves directly optimizing anomaly scores in an end-to-end manner [3–5]. Additionally, some approaches reformulate the problem as a classification task, minimizing cross-entropy loss [6]. The majority of these works are developed for computer vision tasks, with relatively limited work in multi-sensor time-series anomaly detection. Meanwhile, recent advancements in time-series foundation models—such as TimeGPT [7], Lag-Llama [8], GTT [9], TimesFM [10], and Chronos [11]—have shown strong performance in

*Work done during an internship at Amazon.

NeurIPS 2024 Workshop on Time Series in the Age of Large Models.

tasks like zero-shot forecasting. Nevertheless, adapting these models to anomaly detection remains a largely unexplored field, with only a few initial works exploring this problem [12]. Adapting these models for weakly-supervised anomaly detection with heterogeneous sensors remains an open research question.

In this paper, we introduce a weakly-supervised multi-sensor anomaly detection framework (WMAD) that learns a deep neural network to construct a data-enclosing hypersphere, which can effectively distinguish normal from anomalous time windows. The framework employs two-level importance sampling and meta-learning techniques to handle both the sparsity of anomalies and the heterogeneity of sensor data. We evaluate both backbone networks trained from scratch and those fine-tuned from pre-trained foundation models. Our approach achieves promising performance on a large industrial amperage dataset as well as on a public telemetry dataset, outperforming a range of baselines. The code is available at https://github.com/ZLHe0/WMAD-Sensor-Anomaly-Detection.

2 Methods

2.1 Problem Statement

We address the problem of multi-sensor, univariate time-series anomaly detection. Assume that we have K *independent* sensors, where each sensor collects a time series $\mathcal{Z}^{(k)} = \{z_t^{(k)}\}_{t\in[T_k]}$, with $k \in [K]$ and T_k is the time length recorded by the kth sensor. Our objective is to detect time windows in $\mathcal{Z}^{(k)}$ that exhibit persistent anomalous behavior. To achieve this, we transform the original time series into h-length moving time windows $\mathcal{X}^{(k)} = \{\boldsymbol{x}_t^{(k)} = (z_t^{(k)}, z_{t+1}^{(k)}, \ldots, z_{t+h-1}^{(k)})\}_{t\in[T_k-h+1]}$ and aim to assign a single anomaly label $y_t^{(k)} \in \{0, 1\}$ to each window $\boldsymbol{x}_t^{(k)}$. One challenge in this setting is the extremely limited number of labeled anomalous windows, denoted as $\mathcal{A}^{(k)} = \{t : y_t^{(k)} = 1\}$, compared to the vast number of unlabeled windows, denoted as $\mathcal{U}^{(k)} = \{t : y_t^{(k)} = 0\}$. In fact, for many sensors we have $|\mathcal{A}^{(k)}| = 0$, meaning that there are no labeled anomalies. This scarcity of labeled anomalies necessitates a strategy to gain global knowledge from the data across all K sensors.

2.2 Weakly-supervised Multi-sensor Anomaly Detection (*WMAD*)

We now introduce *WMAD*, a framework designed to leverage sparse labeled anomalies for multi-sensor anomaly detection. Using importance sampling and meta-learning, *WMAD* learns an adaptive hypersphere to separate normal and anomalous data in different sensors with minimal labeled data.

Hypersphere Learning with Pre-trained Representation. To incorporate labeled anomalies, we adopt a semi-supervised anomaly detection (SAD) framework [2]. For each sensor, our objective is to learn a neural network, $\phi(\cdot; \mathcal{W})$, to construct a data-enclosing hypersphere that effectively separates normal from anomalous data:

$$\min_{\mathcal{W}=\mathcal{W}_0+\Delta\mathcal{W}} \frac{1}{|\mathcal{U}^{(k)}|} \sum_{i\in\mathcal{U}^{(k)}} \left\|\phi(\boldsymbol{x}_i^{(k)}; \mathcal{W}) - \mathbf{c}\right\|^2 + \frac{\eta}{|\mathcal{A}^{(k)}|} \sum_{j\in\mathcal{A}^{(k)}} \left\|\phi(\boldsymbol{x}_j^{(k)}; \mathcal{W}) - \mathbf{c}\right\|^{-2}. \quad (1)$$

Here, $\phi(\boldsymbol{x}_i^{(k)}; \mathcal{W}) = \phi(\boldsymbol{x}_i^{(k)}; \mathcal{W}_0 + \Delta\mathcal{W})$ represents the mapping from the input space to the embedding space, with the weights $\mathcal{W}$ initialized from a pre-trained network, $\mathcal{W}_0$, and further optimized as $\mathcal{W}_0 + \Delta\mathcal{W}$. $\mathbf{c}$ is the hypersphere center, which is computed as the average of all representations from the first forward pass through the pre-trained network. The first term in (1) sums over the unlabeled (mostly normal) data points, aiming to minimize their representation's distance to the center $\mathbf{c}$ of the hypersphere. The second term focuses on the labeled anomalous points indexed by $\mathcal{A}$, maximizing their representation's distance to the center. The weighting term η represents the relative importance of the anomalous class and is incorporated using importance sampling. By learning such a hypersphere, we create a compact representation that clearly distinguishes normal and abnormal data. During inference, data mapped far from the center $\mathbf{c}$ are considered anomalous.

Importance Sampling and Meta-Learning. In such a weakly supervised setting, one key challenge is the sparsity of labeled anomalies. Many sensors may lack any labeled anomalies, and the labeled anomalies that do exist may be from sensors with very different normal patterns. Therefore, it's necessary to develop a training paradigm that produces a generalizable global model that can capture the global knowledge about anomalies and pass it to different sensors. To address this, we implement a

two-level sampling strategy and a meta-learning paradigm for the *WMAD* model training. Specifically, we modify (1) into the following objective:

$$\min_{\mathcal{W}=\mathcal{W}_0+\Delta\mathcal{W}} \underbrace{\sum_{k\sim p_s}}_{\text{sensor-level}} \underbrace{\sum_{i\sim p_{s,w}}}_{\text{window-level}} \left[\left\| \phi[\boldsymbol{x}_i^{(k)}; \mathcal{W} - \underbrace{\alpha\,\mathrm{Grad}_k(\mathcal{W})}_{\text{adaptation}}] - \mathbf{c} \right\|^{2\cdot[\mathbb{I}(i\in\mathcal{U}^{(k)})-\mathbb{I}(i\in\mathcal{A}^{(k)})]} \right]. \quad (2)$$

The sensor-level sampling prioritizes sensors with labeled anomalies by assigning them higher probabilities to those sensors during training. Within each sampled sensor, the window-level sampling further emphasizes windows containing labeled anomalies, ensuring that the model allocates sufficient focus on the sparse labels rather than diluting them across numerous unsupervised sensors.

To enhance generalization across diverse sensors, we integrate Model-Agnostic Meta-Learning (MAML) [13] into the training process to construct a highly generalizable global model ϕ. For each sensor sampled via the sensor-level strategy, a *support set* of windows is selected using window-level sampling. The model then adapts to the sensor-specific patterns through gradient updates, modifying its parameters to $\mathcal{W} - \alpha\,\mathrm{Grad}_k(\mathcal{W})$. A separate *query set* from the same sensor evaluates the adapted model by calculating the loss on unseen data, simulating testing conditions. This loss is then used to update the global model, aggregating insights across sensors iteratively. During testing, the global model undergoes a similar fine-tuning process, where a small amount of data from a new sensor is used to adapt the global model to the specific conditions of that sensor. The resulting local models, enriched with the knowledge gained from the global model, are better equipped to recognize anomalies while also being finely tuned to the normal patterns of their respective sensors.

2.3 Adapting Foundation Models for *WMAD*

To initialize the weights $\mathcal{W}_0$ for *WMAD* framework (2), we consider two primary approaches. The first approach involves training an autoencoder from scratch on the dataset and then using the encoder part of the network as the initialization. The second approach is to use pre-trained time-series foundation models, which provide highly generalizable representation that could both enhance the learning performance and avoiding the need for training from scratch. To serve as the backbone model for our weakly supervised *WMAD* method, the pre-trained model should have an encoder structure and fully open-sourced model weights. Recently there are some pioneering efforts in building time-series foundation models including TimeGPT [7], Lag-Llama [8], GTT [9], and TimesFM [10] and Chronos [11]. Among these, the General Time Transformer (GTT) meets these criteria. GTT is an open-source transformer-based model pre-trained on a dataset of 200M high-quality time series samples. GTT exhibits excellent zero-shot time series forecasting performance on various benchmark datasets. For our application, we adopt the entire architecture of the largest available GTT version, GTT-small with 22M parameters as the initial weight configuration, $\mathcal{W}_0$ for our *WMAD* framework. The final layer of GTT, originally designed for forecasting, is fine-tuned specifically for the *WMAD* objective. Further details on the fine-tuning process are discussed in Appendix A.

3 Experiments

We evaluate *WMAD* with two different backbones: *WMAD-GTT*, which uses the pre-trained General Time Transformer (GTT) model, and *WMAD-AE*, which uses an autoencoder (AE) trained from scratch. We evaluate our method against several baselines, which include unsupervised counterparts of the proposed method and other strategies for adapting the GTT foundation model to the task. *AE* is a deep CNN autoencoder that reconstructs input data from a compressed latent space, with reconstruction error serving as the anomaly measure. *SVDD-GTT* fine-tunes GTT in an unsupervised, one-class setting [14], and is essentially the unsupervised version of *WMAD-GTT*, where no labeled anomalies are incorporated. *FLOS-GTT* fine-tunes the final layer of GTT as an imbalanced binary classifier [4], using focal loss to emphasize the minority class and increase sensitivity to anomalies. *DevNet-GTT* fine-tunes GTT for few-shot anomaly scoring [3], assigning higher anomaly scores to data points deviating from normal patterns. We evaluate the performance of these methods on two real-world sensor anomaly detection tasks, as detailed in Table 1. For each method, we report the number of true positives (TP), false positives (FP), precision, and recall. Additionally, we compute the F0.5 score to emphasize precision over recall, as minimizing false alarms is critical to avoid

unnecessary resource allocation. Additional experiment details, including model implementation, anomaly thresholding, and evaluation metrics can be found in Appendix A.

Table 1: Dataset Information. "# Labels" refers to the total number of labeled anomalies for training.

Dataset	Type	# Sensors	Sampling Rate	Total Series Length	# Labels
Amperage	Proprietary	330	15 minutes	728,640 hours	5
Telemetry	Public	82	1 minute	8,274 hours	5

Dataset Information. As shown in Table 1, the Amperage dataset is a proprietary industrial dataset collected from an Amazon fulfillment center, consisting of data from 330 sensors spanning over 700,000 total hours, monitoring the amperage in VFD motors. It includes high-quality, manually labeled anomalies linked to equipment breakdowns, with performance evaluated based on the work orders generated within the specified time range. The Telemetry dataset is a public dataset derived from the NASA telemetry dataset [15], comprising 82 sensors covering over 8,000 total hours. Expert-labeled anomalies validated by engineers are provided.

Evaluation Results. The performance comparison across different methods is shown in Table 2. We begin by analyzing the results on the amperage dataset, where *WMAD-GTT*, the proposed weakly-supervised anomaly detection model based on a foundation model, achieved the highest precision at 0.580, a 22.9% improvement over the autoencoder-based model, *WMAD-AE*, and a 38.3% improvement over the next best weak-supervised baselines. In industrial anomaly detection, precision is typically prioritized over recall to minimize false alarms, making *WMAD-GTT* particularly favorable. Despite a slight drop in recall (0.103 for *WMAD-GTT* vs. 0.111 for *WMAD-AE*), *WMAD-GTT* maintained a strong F0.5 score due to its precision. Notably, *WMAD-GTT* excelled in detecting critical breakdown events (Damaged and Intfail), outperforming other models in these categories. When labeled anomalies were not incorporated, performance dropped significantly: from *WMAD-GTT* to *SVDD-GTT*, the F0.5 score decreased by 64.8%, and from *WMAD-AE* to *AE*, it dropped by 66.4%. This underscores the value of leveraging sparse labeled anomalies to improve both precision and recall. Results on the telemetry dataset followed a similar pattern, with *WMAD-GTT* achieving the highest precision at 0.722, surpassing all competing methods. It also maintained the highest F0.5 score of 0.383. Overall, *WMAD*-based methods uniformly outperformed other models, particularly in scenarios where precision is critical.

Table 2: Performance comparison across different methods. Methods with (U) denote unsupervised learning approaches. For the Telemetry dataset, reported values represent the mean and standard deviation (subscript) over 10 replicates. The best performer is highlighted in bold.

Dataset	Metric	Category	WMAD-GTT	WMAD-AE	FLOS-GTT	DevNet-GTT	SVDD-GTT(U)	AE(U)
Amperage	TP	Damaged	**43**	40	9	19	8	11
		Intfail	**28**	26	5	9	2	9
		Others	38	**51**	4	46	17	5
		Total	109	**117**	18	74	27	25
	FP		**79**	131	25	258	26	36
	Precision		**0.580**	0.472	0.419	0.223	0.509	0.410
	Recall		0.103	**0.111**	0.017	0.070	0.025	0.024
	F0.5		**0.301**	0.286	0.073	0.155	0.106	0.096
Telemetry	Precision		$\mathbf{0.722_{0.076}}$	$0.550_{0.087}$	$0.124_{0.007}$	$0.337_{0.044}$	$0.460_{0.077}$	$0.456_{0.104}$
	Recall		$0.137_{0.016}$	$0.152_{0.040}$	$0.234_{0.017}$	$\mathbf{0.322_{0.035}}$	$0.118_{0.019}$	$0.092_{0.037}$
	F0.5		$\mathbf{0.383_{0.022}}$	$0.358_{0.070}$	$0.137_{0.007}$	$0.333_{0.041}$	$0.289_{0.040}$	$0.246_{0.058}$

Case Studies and Embedding Analysis. We present case studies demonstrating *WMAD*'s effectiveness in detecting anomalies in noisy amperage data, and an embedding analysis showing its ability to produce informative latent representations for anomaly diagnosis. Full details are in Appendix B.

4 Conclusion

In this paper, we proposed *WMAD*, an anomaly detection framework designed to handle label sparsity and sensor heterogeneity in industrial datasets. By incorporating two-level importance sampling and meta-learning and leveraging the power of foundation models, *WMAD* achieved promising performance on both industrial amperage and public telemetry datasets.

References

[1] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. Learning representations of ultrahigh-dimensional data for random distance-based outlier detection. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2041–2050, 2018.

[2] Lukas Ruff, Robert A. Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations*, 2020.

[3] Guansong Pang, Chunhua Shen, and Anton Van Den Hengel. Deep anomaly detection with deviation networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 353–362, 2019.

[4] Guansong Pang, Choubo Ding, Chunhua Shen, and Anton van den Hengel. Explainable deep few-shot anomaly detection with deviation networks. *arXiv preprint arXiv:2108.00462*, 2021.

[5] Yingjie Zhou, Xucheng Song, Yanru Zhang, Fanxing Liu, Ce Zhu, and Lingqiao Liu. Feature encoding with autoencoders for weakly supervised anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2454–2465, 2021.

[6] Choubo Ding, Guansong Pang, and Chunhua Shen. Catching both gray and black swans: Open-set supervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7388–7398, 2022.

[7] Azul Garza and Max Mergenthaler-Canseco. Timegpt-1. *arXiv preprint arXiv:2310.03589*, 2023.

[8] Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Arian Khorasani, George Adamopoulos, Rishika Bhagwatkar, Marin Biloš, Hena Ghonia, Nadhir Vincent Hassen, Anderson Schneider, et al. Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

[9] Cheng Feng, Long Huang, and Denis Krompass. Only the curve shape matters: Training foundation models for zero-shot multivariate time series forecasting through next curve shape prediction. *arXiv preprint arXiv:2402.07570*, 2024.

[10] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2023.

[11] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

[12] Gastón García González, Pedro Casas, Emilio Martínez, and Alicia Fernández. On the quest for foundation generative-ai models for anomaly detection in time-series data. In *2024 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 252–260. IEEE, 2024.

[13] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[14] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pages 4393–4402. PMLR, 2018.

[15] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 387–395, 2018.

A Additional Experiment Details

Data Preprocessing and Sampling. For each of the 330 sensors in the Amperage dataset, the raw time series data is aggregated into 15-minute intervals using the median of all sensor readings within each interval. The data is then segmented into sliding time windows of size 256 with stride 1. This window size is chosen as it's large enough to capture essential temporal patterns in the amperage data, providing enough context for distinguishing between normal and anomalous behavior, while also being short enough to avoid diluting anomalous patterns. We then apply median-MAD normalization, where each value is normalized by subtracting the median and dividing by the mean absolute deviation (MAD). This is followed by a scaled sigmoid transformation to map the data between -1 and 1. The use of median-MAD combined with the scaled sigmoid ensures that the processed data remains centered around zero, offering robustness against frequent jumps and drops in sensor data caused by routine maintenance and operational events. Median and MAD values are computed during training and saved for consistent preprocessing during inference. Missing data is interpolated using the median of the previous 256 time points.

In addition to the basic preprocessing, a two-level importance sampling strategy is employed to make sure that the model is trained on a balanced representation of normal and anomalous data and facilitate meta-training. Following the meta-learning training paradigm [13], we create 2,000 tasks for training by first sampling 2,000 sensors with replacements from the pool of 330 sensors. The sampling probability is set to 0.2 for anomalous sensors and 0.8 for normal sensors. For each sampled sensor, we then perform a second level of sampling, drawing 20 windows (shots) for adaptation and 30 windows (queries) for task loss calculation and model backpropagation. If the sensor contains anomalous windows, the sampling ratio of anomalous to normal windows is 1:9. For *DevNet-GTT* and *FLOS-GTT*, this ratio is 1:1 following the set up of the original paper [4]. This same process is applied to the Telemetry dataset, with a few differences: the window size is 128 instead of 256, the sampling rate is 1 minute rather than 15, and 1,000 tasks are sampled for training.

Hyperparameter Setups and Training Details. We adopt two types of background networks for the *WMAD* framework: a CNN-based architecture trained from scratch and the General Time Transformer (GTT), an encoder-based pre-trained model. While we also experimented with LSTM architectures, CNNs outperformed them due to their ability to handle the frequent fluctuations present in amperage sensor data.

The CNN autoencoder consists of an encoder with two Conv1D layers (8 and 4 filters, kernel size 5, stride 2), followed by a fully connected layer that maps the flattened output to a latent vector. The decoder mirrors this with ConvTranspose1D layers for upsampling and reconstruction. The encoder part of this autoencoder is used for the *WMAD* framework, referred to as *WMAD-AE*, and is further trained for the anomaly detection task. We evaluated various filter sizes and configurations, and found that this specific architecture offered the most robust and efficient performance for the sensor data. For the *WMAD-GTT* model, we leverage the GTT foundation model, which consists of six encoder layers, each with multi-head self-attention (eight heads). The model's final embedding size is 512, and the original forecasting head, which produces a 64-dimensional output, is fine-tuned to minimize the *WMAD* objective (2). The entire GTT architecture is adopted, with only the final layer fine-tuned. The GTT architecture is also employed for the *SVDD-GTT*, *DevNet-GTT*, and *FLOS-GTT* baselines, with each model differing in how the final layer is fine-tuned. *SVDD-GTT* fine-tunes the last layer in a similar manner to *WMAD* but without utilizing labeled anomalies. *DevNet-GTT* fine-tunes the final layer to build an anomaly scoring network. The confidence margin is set to be 0.01 to adapt to the scale of time series data. The Gaussian prior-based reference scores, originally used in DevNet, were replaced by original scores to mitigate training instability observed in time-series data. The rest of the setup is the same as the original paper. *FLOS-GTT* uses a focal loss-based cross-entropy to fine-tune the last layer, transforming the GTT into a classifier. The focal loss parameters were set to $\alpha = 0.1$, aligning with the first-level sampling ratio in our framework, and $\gamma = 2$, balancing the focus between minority (anomalous) and majority (normal) classes [4].

For all models on the Amperage dataset, we used the Adam optimizer with MAML training paradigm [13], training the model on sampled tasks for 50 epochs. The weight dess was set to 10^{-6}, with a meta-adapting learning rate of 0.01 and a learning rate of 0.001. For the telemetry dataset, the same configuration was applied except that the model was trained for 20 epochs. Testing different epoch settings showed minimal effect on the final performance. All methods were implemented in PyTorch, except for the GTT foundation model, which was pre-trained in TensorFlow.

Anomaly Detection and Evaluation Procedure. For the amperage dataset, anomaly scores from the previous month's data for each sensor are first calculated, with the 99th quantile of the Gaussian tail set as the testing data anomaly threshold. An alarm is triggered if a sensor's anomaly score exceeds this threshold for 36 consecutive time windows (9 hours). The detected anomalous intervals are compared against work orders created within the same month for each sensor. Anomalies occurring within a window of 10 days prior to or 3 days after a work order are considered true positives; otherwise, they are marked as false positives. Any work orders without corresponding raised anomalies in the defined time frame are regarded as false negatives. Precision, recall and F0.5 scores are calculated based on these true positive and false positive rates. The evaluation is conducted monthly and results are aggregated over three months. For the telemetry dataset, the anomaly threshold is determined using the provided training set and subsequently applied to the test set for each channel. We observe that models fine-tuned from foundation models, such as *WMAD-GTT*, tend to be more conservative in their predictions compared to models trained from scratch, such as *WMAD-AE*. For a better comparison between the two, we control *WMAD-GTT* and *WMAD-AE* to achieve a comparable level of recall for a comparison of their precision. This results in a threshold at the 95th quantile for all models trained from scratch and at the 85th for all fine-tuned foundation models. In addition to relative thresholding, to enhance the robustness of anomaly detection across all methods, we impose an absolute score threshold by requiring detected anomalies to exceed the 25th quantile of all collected scores. An alarm is triggered if a sensor's anomaly score exceeds this threshold for 30 consecutive time windows (30 minutes). A true positive is defined as an anomaly detected up to 5 hours before the labeled anomalous interval or overlapping with it. Evaluation metrics are similarly calculated.

B Additional Experiment Results

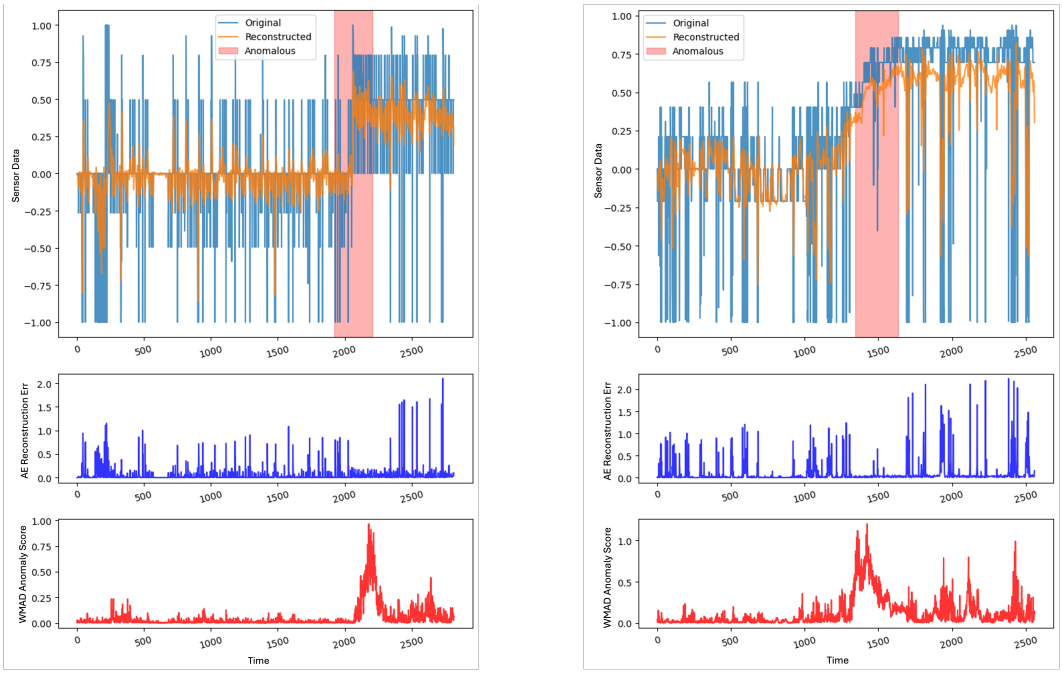
Case Study of *WMAD* Detections. Figure 1 presents case studies of amperage sensor data anomaly detection using the *WMAD-GTT* model. Both Figure 1a and 1b consist of three sub-figures, each illustrating different aspects of the detection result. In the top sub-figure of each, we show the processed sensor data, the reconstructed data from the autoencoder (AE) model, and the labeled anomalous time intervals, marked in red. The middle sub-figure visualizes the autoencoder reconstruction error, where each time point represents the error between the original sensor data and the reconstructed data. The bottom sub-figure shows the *WMAD* anomaly scores, where each score corresponds to the window ending at that time point, and the window length is of 256 time points.

In Figure 1a, we observe that the autoencoder fails to capture the labeled anomalous patterns effectively, as it does not exhibit high reconstruction error for the actual anomalous time window. Instead, it generates high errors in response to sudden drops and spikes in the sensor data, likely caused by routine maintenance events or changes in the working conditions, such as equipment shutdowns, restarts, or load shifts. To improve the performance of the autoencoder, one may consider addressing these fluctuations using pre-processing steps to remove outliers. However, such patterns are pervasive in the amperage dataset and exhibit diverse forms, making them difficult to isolate. Additionally, pre-processing could diminish important anomaly patterns and make the model more susceptible to outliers. By contrast, *WMAD* can directly work on raw sensor data and detect the switch in amperage patterns, raising alarms aligned with the labeled anomaly.

A similar observation can be made from Figure 1b. While the autoencoder produces high reconstruction errors for abrupt drops in amperage, which are unrelated to breakdown events, the *WMAD* model accurately identifies the rise in amperage indicative of a breakdown and raises the alarm accordingly. This demonstrates the *WMAD* model's ability to integrate sparse labeled anomalies to enhance detection precision, even in noisy industrial sensor data.

Embedding Analysis of *WMAD* Representations. In Figure 2, we present an analysis of the learned latent space by our *WMAD-AE* model to investigate the potential clustering of anomalous data representation. The figure shows a 2D scatter plot of the latent representations, obtained by applying T-SNE for dimensionality reduction on the high-dimensional embeddings produced by the *WMAD* model. Points represent anomalous time windows from six different sensors, each assigned a distinct color. For each sensor, five anomalous windows are sampled, resulting in six sets of colored points. In the figure, we highlight three distinct clusters in the latent space. Each cluster corresponds to a specific type of anomalous pattern, which is further described through images adjacent to the clusters.

In the bottom-left cluster, two sensors are grouped together, both exhibiting a pattern of sharp jumps followed by periodic drops in amperage, which is related to a missing belt issue. Similarly, the central cluster corresponds to two sensors showing a sharp jump followed by stabilized amperage, associated with cracked belts and belt replacements. The top-right cluster includes two sensors characterized by a gradual rise in amperage, indicative of operational issues. This analysis reveals that the *WMAD* model not only distinguishes between normal and anomalous data in the latent space but also enables the clustering of different types of anomalies. This clustering capability facilitates anomaly diagnosis, as newly detected anomalies can be compared to previously labeled anomalies with known root causes. By mapping new anomalies to similar past events, the model can aid in diagnosing underlying issues, extending beyond simple anomaly detection. This represents an advantage of *WMAD* framework over scoring or classification-based approaches, which do not typically produce such meaningful representations for further interpretation.



(a) Shifts in amperage patterns.

(b) Rise in baseline amperage levels.

Figure 1: Case studies illustrating the detection performance of the *WMAD-GTT* model compared to the *AE* model on amperage sensor data. Each figure contains three sub-figures: the processed sensor data with labeled anomalies, the *AE* reconstruction error, and the *WMAD* anomaly score.

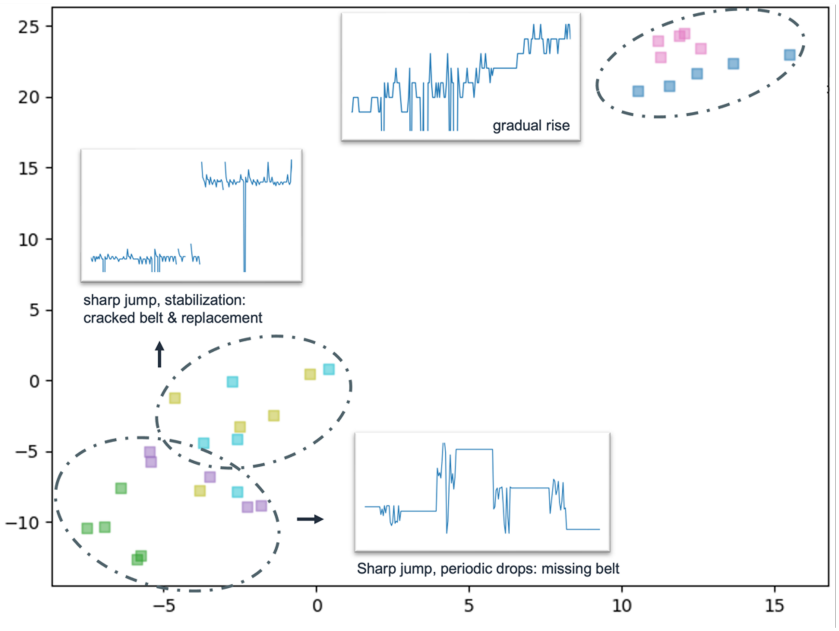


Figure 2: Visualization of the latent representation learned by the *WMAD-AE* model, where the high-dimensional representation is reduced to two dimensions using T-SNE. Each colored point represents the representation of an anomalous time window from one of six different sensors. The six different colors correspond to the six sensors. Clusters of similar anomalies are highlighted.