



Article

TWIN-ADAPT: Continuous Learning for Digital Twin-Enabled Online Anomaly Classification in IoT-Driven Smart Labs

Ragini Gupta , Beitong Tian , Yaohui Wang and Klara Nahrstedt *

Department of Computer Science, University of Illinois at Urbana-Champaign, Champaign, IL 61820, USA; raginig2@illinois.edu (R.G.); beitung2@illinois.edu (B.T.); yaohuiw2@illinois.edu (Y.W.)

* Correspondence: klara@illinois.edu

Abstract: In the rapidly evolving landscape of scientific semiconductor laboratories (commonly known as, cleanrooms), integrated with Internet of Things (IoT) technology and Cyber-Physical Systems (CPSs), several factors including operational changes, sensor aging, software updates and the introduction of new processes or equipment can lead to dynamic and non-stationary data distributions in evolving data streams. This phenomenon, known as concept drift, poses a substantial challenge for traditional data-driven digital twin static machine learning (ML) models for anomaly detection and classification. Subsequently, the drift in normal and anomalous data distributions over time causes the model performance to decay, resulting in high false alarm rates and missed anomalies. To address this issue, we present TWIN-ADAPT, a continuous learning model within a digital twin framework designed to dynamically update and optimize its anomaly classification algorithm in response to changing data conditions. This model is evaluated against state-of-the-art concept drift adaptation models and tested under simulated drift scenarios using diverse noise distributions to mimic real-world distribution shift in anomalies. TWIN-ADAPT is applied to three critical CPS datasets of Smart Manufacturing Labs (also known as “Cleanrooms”): Fumehood, Lithography Unit and Vacuum Pump. The evaluation results demonstrate that TWIN-ADAPT’s continual learning model for optimized and adaptive anomaly classification achieves a high accuracy and F1 score of 96.97% and 0.97, respectively, on the Fumehood CPS dataset, showing an average performance improvement of 0.57% over the offline model. For the Lithography and Vacuum Pump datasets, TWIN-ADAPT achieves an average accuracy of 69.26% and 71.92%, respectively, with performance improvements of 75.60% and 10.42% over the offline model. These significant improvements highlight the efficacy of TWIN-ADAPT’s adaptive capabilities. Additionally, TWIN-ADAPT shows a very competitive performance when compared with other benchmark drift adaptation algorithms. This performance demonstrates TWIN-ADAPT’s robustness across different modalities and datasets, confirming its suitability for any IoT-driven CPS framework managing diverse data distributions in real time streams. Its adaptability and effectiveness make it a versatile tool for dynamic industrial settings.

Keywords: internet of things (IoT); cyber-physical systems (CPSs); concept drift; digital twins; online anomaly classification; scientific laboratories; cleanrooms



Citation: Gupta, R.; Tian, B.; Wang, Y.; Nahrstedt, K. TWIN-ADAPT: Continuous Learning for Digital Twin-Enabled Online Anomaly Classification in IoT-Driven Smart Labs. *Future Internet* **2024**, *16*, 239. <https://doi.org/10.3390/fi16070239>

Academic Editors: Xin Lu, Wei Wang, Dehao Wu and Xiaoxia Li

Received: 31 May 2024

Revised: 25 June 2024

Accepted: 25 June 2024

Published: 4 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Digital twins are emerging as a pivotal technology in the landscape of Industry 4.0, offering dynamic and real-time simulation capabilities that extend across various sectors including manufacturing [1–5], healthcare [6], smart agriculture [7] and scientific research. A digital twin is a real-time virtual model of a physical object or system that mirrors and analyzes its behavior for optimization and decision making [5,8]. Digital twins can be modeled using multiple approaches, each offering distinct advantages depending on the application. One method involves using formal verification techniques that offer theoretical guarantees on system performance by establishing a set of feasible working conditions through sensitivity analysis and uncertainty decomposition [9,10]. These analyses assess

the robustness of system models by identifying conditions where the models are valid and stable. Any deviations from these established norms are leveraged for predictive monitoring, enhancing system reliability through proactive anomaly detection. However, these models require significant domain expertise and knowledge, which limits their application across heterogeneous environments. Additionally, their effectiveness is constrained by the accuracy of the initial assumptions and the feasibility of the working conditions derived from the analyses. Alternatively, data-driven approaches utilize real-world data to create virtual representations of physical systems [5,11]. These models require substantial amounts of data to train effectively. Once deployed as a digital twin, they continuously analyze incoming data to identify and report any deviations, anomalies, or faults in the physical system. Both approaches aim to optimize system performance and reliability, which makes them powerful tools for enabling preventive, predictive and reactive maintenance in varied application domains through real-time monitoring as well as informed decision making.

In the context of scientific manufacturing laboratories (commonly known as “Cleanrooms”), which are strictly controlled environments for industries like semiconductor, pharmaceuticals, microelectronics and nanotechnology, data-driven digital twins play a vital role. These digital twins provide a high level of data monitoring and operational control to prevent contamination and ensure the integrity of sensitive manufacturing processes. For example, a cleanroom’s critical infrastructure components such as Fumehoods and high-end scientific equipment such as Electron Microscopes, Vacuum Pumps, and high-resolution Photolithography Units like the Karl Suss MJB3 Contact Mask Aligner [12], all require continuous monitoring to maintain the laboratory’s operational standards. Digital twins facilitate this by enabling predictive, preventive and reactive maintenance within industrial laboratories as follows: (1) by continuously monitoring system conditions to predict potential failures before they occur, (2) through advanced analytics to anticipate equipment malfunctions and optimize maintenance schedules and (3) by generating real-time updates and alerts to control the physical environment. Figure 1 shows each of the three critical components of cleanrooms that require real-time monitoring: (a) Fumehood (b) Vacuum Pump and (c) the Karl Suss Lithography Unit.

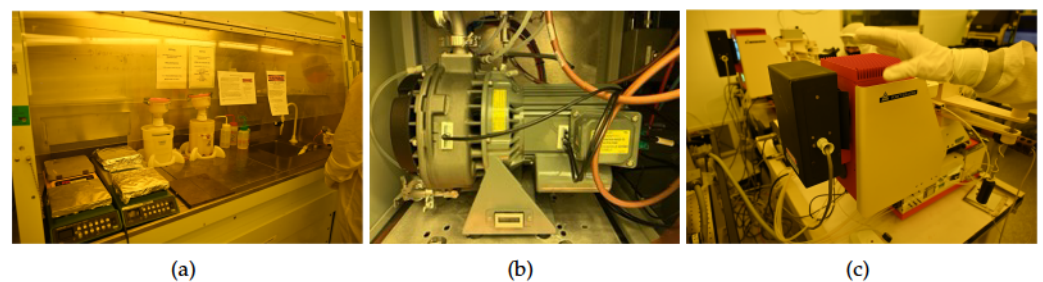


Figure 1. Critical components of Cleanroom: (a) Fumehood, (b) Vacuum Pump and (c) Lithography Unit.

Existing deployments of cyber-physical systems (CPSs) in cleanrooms enhance laboratory functionality by deploying sensors, actuators and edge devices with the physical infrastructure and scientific equipment of the cleanroom labs. For example, Senselet++ [13] uses cost-effective sensors to monitor airflow in Fumehoods and track calibration drifts caused by environmental changes to ensure it remains within safe limits [13]. Similarly, the sensors within the Fumehood are used to monitor air leakage or chemical spillage causing changes in air pressure or hazardous vapor concentrations. Edge devices, which may be microcontrollers or microcomputers such as Raspberry Pis, facilitate the communication of sensor data to the cloud. These edge devices also host the complex data processing algorithms for real-time anomaly detection and identification. Actuators, then, provide real-time alerts to lab managers (in the form of alert messages and alarms), informing them of deviations so they can take immediate action to maintain the required cleanroom

standards. Similarly, to monitor Vacuum Pump malfunctions, temperature and vibration sensors are deployed to detect any changes caused by overheating from nearby thermal furnaces. These sensors generate critical data that are required to be analyzed using various anomaly detection and fault classification models to extract meaningful insights from the data.

Despite the advancements in wireless communications and Internet of Things (IoT) technologies facilitating real-time data acquisition in cyber-physical systems, significant challenges persist in capturing and integrating this information. This is largely due to the complexities involved in transforming vast quantities of data into actionable insights. Since digital twins are the virtual representation of cyber-physical systems, they are enabled with real-time monitoring and analytics. Digital twins, serving as virtual representations of these CPSs, utilize data-driven models equipped with various machine learning and deep learning techniques. These models are trained on extensive historical data to provide robust anomaly detection and classification capabilities, ensuring continuous monitoring and analytics within these dynamic environments. However, despite their advanced monitoring capabilities, the effectiveness of these digital twins largely depends on the underlying anomaly detection models that are used for analysis. The static ML models used for data analysis and anomaly detection face several of the following limitations, which can hinder their effectiveness in such controlled IoT-driven smart laboratories:

- Concept drift:** Traditional ML models are typically trained on historical data representing specific, static conditions. These pre-trained models operate under the assumption that the training data accurately represent the relationships within all data from the target variables (anomaly or normal). This requires data stationarity throughout the model training, testing and deployment phases. However, this assumption is often not met in practical scenarios, particularly in CPS applications that involve streaming data analysis. In numerous industrial contexts, data gathered from manufacturing and operational processes inherently exhibit non-stationary characteristics [1]. For example, over time, the physical components of Vacuum Pumps may wear down, subtly changing their operational efficiency and characteristics. This wear and tear can alter the vibration signatures, heat emissions, or other measurable parameters that were initially used to train the anomaly classification models. Similarly, changes in the operational parameters of the Pump, such as adjustments to speed, pressure settings, or duty cycles to accommodate different scientific experimental tasks in cleanrooms, can lead to concept drift. These operational changes can create new data patterns that the original model was not trained to recognize or handle. This can lead the model performance to deteriorate over time on new, unseen data of which its statistical characteristics have changed significantly from the training dataset. This shift can make the separation curve ambiguous for pre-trained models for correctly distinguishing the normal and anomalous data instance, affecting performance. This phenomenon, known as **concept drift**, can render previously trained models outdated or irrelevant. Figure 2 illustrates the challenge of concept drift in a data-driven digital twin model that has been trained on historical data distributions for anomaly classification. When the real-time data conforms to the identical and independently distributed (iid) model, the performance remains significantly high. However, as operational changes or sensor variations lead to drifts in data distribution (due to the existence of concept drift), the model's performance drastically decreases. This highlights the critical need for adaptive learning mechanisms within digital twins to fit the current concepts of the new data streams.
- Lack of continuous learning:** Traditional ML models require manual retraining and fine-tuning to adapt to new data or changes within the environment, a process that is resource-intensive and impractical for real-time applications. These models typically do not support continuous learning, which is essential for adapting to unpredictable changes within the non-stationary streams of data from CPSs within cleanrooms.

- **Data quality and volume:** The performance of these models heavily depends on the quality and volume of the data that they were trained on, often compromised by missing values or noise due to the complex equipment and multidimensional features involved.

To overcome these challenges, the integration of digital twins as adaptive anomaly identification systems represents a transformative solution. We propose, **TWIN-ADAPT**, a continuous learning framework for digital twin-enabled real-time anomaly classification in IoT-driven smart manufacturing laboratories (cleanrooms). The digital twin framework of TWIN-ADAPT consists of a continual learning-based gradient boosting classifier, LightGBM, known for efficiently handling complex data structures in dynamic environments [14]. These models are deployed on edge devices within the cyber-physical systems directly connected to the sensors. These sensors continuously produce real-time data streams that are fed into the adaptive anomaly classification model of the digital twin operating on the edge devices. The TWIN-ADAPT framework of the digital twin leverages a dual-window strategy combining an adaptive sliding window and a fixed size sliding window for concept drift detection, adaptation and data learning. The sizes of these windows are determined based on the arrival of the data streams. When new data arrive, both the adaptive and fixed windows slide forward, and the system evaluates the classification accuracy of data within the current window against the previous window. If a significant drop in accuracy is observed, surpassing a predefined threshold, it signals the presence of concept drift. This detection triggers an immediate retraining of the classifier using the most recent data captured in the adaptive window.

In addition, the framework also employs an optimization function using Particle Swarm Optimization (PSO) for choosing optimal hyperparameters for the drift adaptation algorithm as well as the anomaly classification model. The idea is based on the work by Yang and Shami [14] that was originally developed for high-dimensional (≈ 80 features) network traffic analysis, which we have adapted for the less complex, yet highly variable data dimensions (features) of cleanroom CPSs. Comparing with [14], we also applied off-the-shelf popular drift adaptation techniques (see details in Section 4.3) to compare their performance against that of TWIN-ADAPT's continual learning framework. Other significant contributions of this work include the following:

- Designing a more comprehensive dataset featuring diverse anomaly distributions to robustly train and evaluate the TWIN-ADAPT model under simulated drift conditions.
- Leveraging the PSO enhanced continual learning models for real-time anomaly classification within a cleanroom's CPSs, like Vacuum Pumps, Fumehoods and Lithography Units. Each CPS component's monitoring model represents its digital twin that accurately reflects the current operational states and anomalies. When concept drift is detected—indicating changes in operational conditions or emerging anomaly patterns—the digital twin's continual learning model dynamically adapts by retraining on new data samples captured through an adaptive sliding window. This ensures that the digital twin remains aligned with the dynamic and non-stationary data distributions, providing robust, up-to-date anomaly monitoring in the dynamic environment of cleanrooms.

It is also worth mentioning that TWIN-ADAPT is primarily deployed within a supervised setting, tailored for anomaly (binary) classification. This approach holds similarity to fault classification processes, where faults are identified through deviations from established normal behaviors—similar to detecting anomalies. The rest of this paper is organized as follows. Section 2 provides a comprehensive background on the concept and importance of digital twins in Industry 4.0. In Section 3, we discuss related works pertaining to the application of digital twins, concept drift detection, adaptation and continual learning strategies for scientific cleanroom laboratories. Section 4 elaborates on this study's methodology, including the dataset description, anomaly injection techniques, model selection and optimization for online drift adaptation. Next, we present a complete system overview in Section 5 that explains the proposed TWIN-ADAPT algorithm and experimental setup. Ex-

perimental results are discussed in Section 6 followed by the Discussion, Future Directions and Conclusion in Section 7, Section 8 and Section 9, respectively.

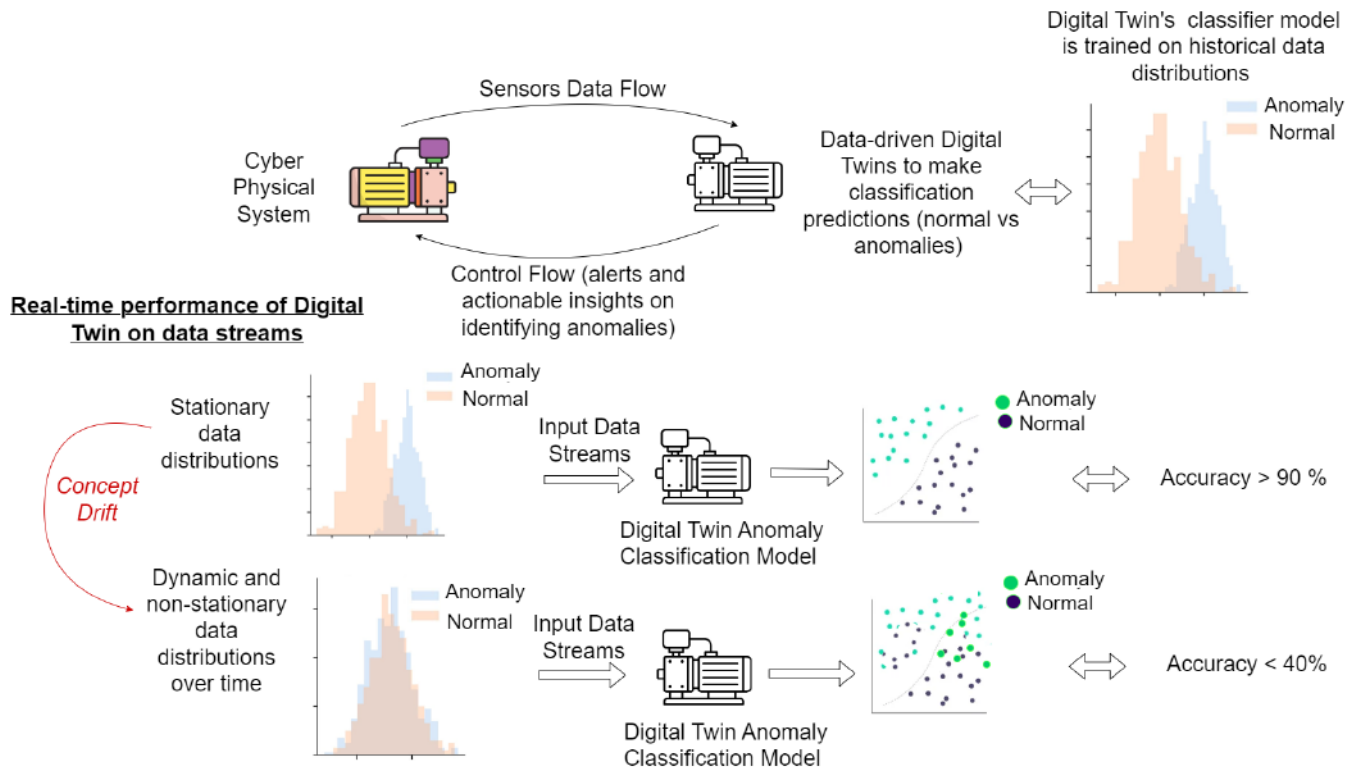


Figure 2. Challenges posed by concept drift on data-driven digital twin anomaly classification.

2. Background

2.1. Concept of Digital Twins

A digital twin is defined as a dynamic digital representation of a physical entity (which could be an object, a system, a process or a workflow) that allows for a two-way automatic data exchange, ensuring that any changes in the physical entity are mirrored in the virtual replica model and vice versa [8]. This integration facilitates real-time updates and interactions, enhancing the accuracy and utility of the digital model in analyzing and predicting the physical counterpart's behavior. Digital twins originated from NASA in the 1960s Apollo program to create a virtual twin of the spacecraft [8], and it was later expanded for various applications in manufacturing processes to create virtual replicas of factories [15]. A digital twin is different from the conventional concepts of digital model and digital shadow. The former involves a mathematical or a theoretical model to study the behavior and/or performance of an actual physical system (such as simulation, 3D models and Computer-Aided Designs), whereas the latter represents an evolving digital representation of an object, where a change in the physical object's state is reflected in the digital model but not vice-versa (such as emulators, anomaly detection and visualizations). Superseding these capabilities, a digital twin represents a "personalized" virtual model of the physical object instead of a generic model that is dynamically updated with real-time data to make informed decisions to fully realize the physical object's value. These personalized models of a digital twin are built using behavioral and contextual (features) data to track the performance variable of the system in real time. The autonomous exchange of data between the physical object and digital twin requires continuous learning to keep the twin in sync with the model under dynamic environments. Figure 3 illustrates digital model, digital shadow and digital twin capabilities and the flow of information between physical and digital counterparts.

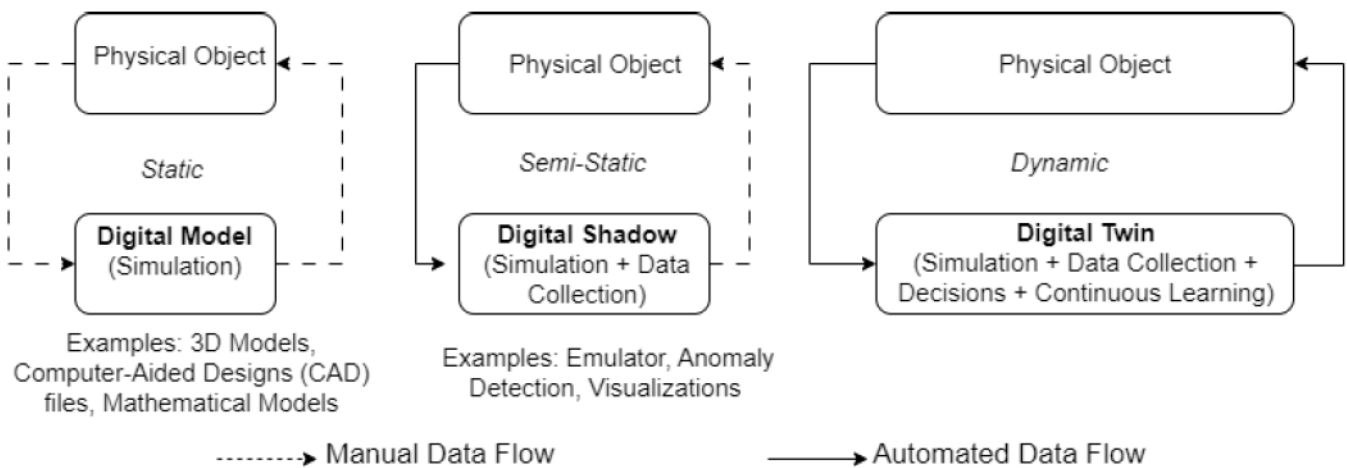


Figure 3. Digital model, digital shadow and digital twin data flow.

2.2. Importance of Digital Twins in Industry 4.0

In a conventional smart manufacturing setup, various sensors embedded in machines collect operational and performance data. Engineers leverage these data to pinpoint causes of machine failures and detect early signs of malfunctioning components to eliminate unplanned downtime. But with the rapid growth in Artificial Intelligence (AI), Internet of Things (IoT), cyber-physical systems (CPSs), edge computing and wireless communication technologies, Industry 4.0 has evolved into bringing together a multitude of industrial devices to create a complete digital ecosystem that can monitor, collect, share and analyze data to provide valuable insights with advanced predictive and prescriptive capabilities. Figure 4 outlines the levels of intelligence in Industry 4.0, progressing from descriptive to prescriptive (reactive) intelligence, illustrating how complexity and value increase at each stage. Descriptive intelligence involves analyzing historical data to understand past events. Diagnostic intelligence focuses on identifying the reasons behind events using techniques like root cause analysis. Predictive intelligence anticipates future events through trend analysis and anomaly prediction. Prescriptive (reactive) intelligence uses real-time data and continuous learning to provide actionable recommendations and adaptive responses, crucial for proactive maintenance in Industry 4.0. Central to achieving these advanced levels of intelligence is the concept of a “Digital Twin” that allows for real-time decision making through accurate analytics. Digital twins provide comprehensive, real-time digital replicas of physical entities, enhancing decision making and efficiency. The diverse capabilities of digital twins, including system modeling, real-time monitoring, optimizations, decision making and predictive and prescriptive maintenance, make digital twins a helpful tool in various applications of Industry 4.0. In aerospace and automotive applications, digital twins optimize performance, predictive maintenance and customization. In smart manufacturing, digital twins provide virtual design verification, process control and real-time fault prediction and diagnosis using machine learning. In healthcare, digital twins enable personalized medicine and optimized hospital operations. The energy sector utilizes digital twins for optimizing wind farms, power grids and nuclear plants. Smart cities and construction industries leverage digital twins for urban planning, resource management and building lifecycle management [16–18]. Thus, these digital twins serve as a powerful tool for transforming traditional industrial manufacturing into intelligent, adaptive systems capable of real-time analysis and decision making.

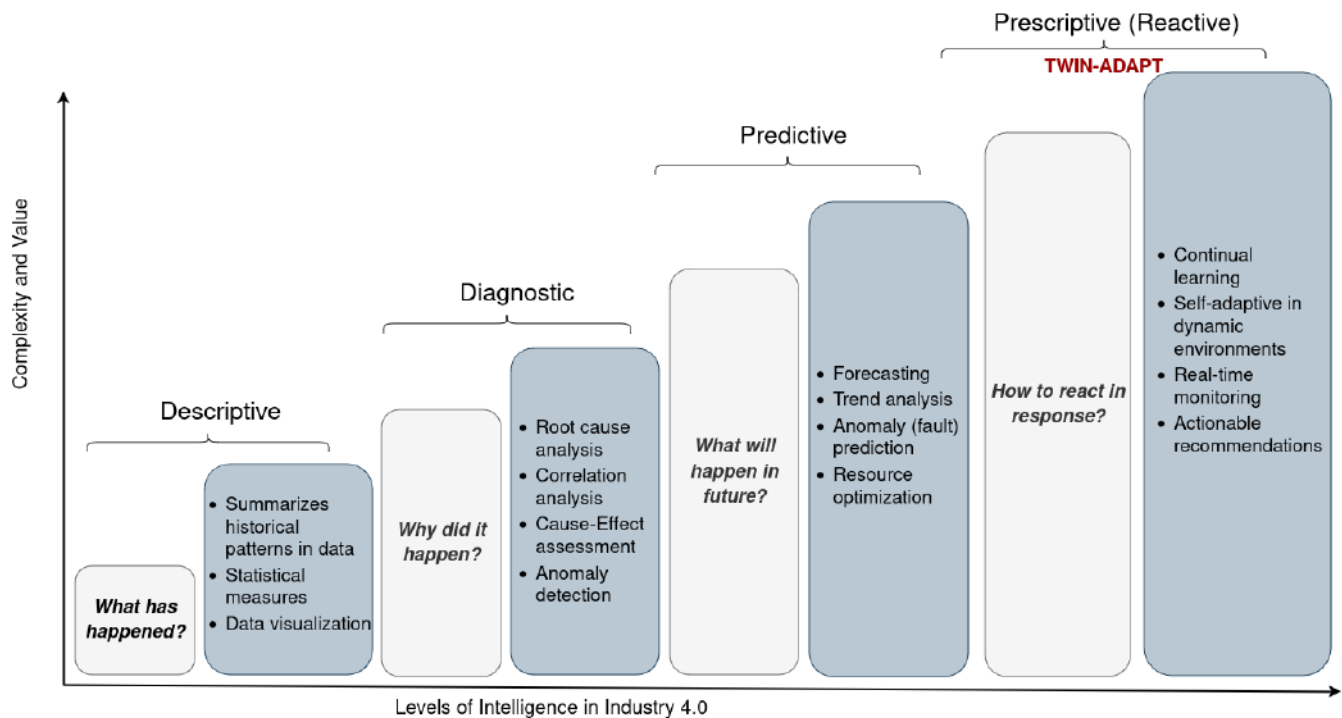


Figure 4. Levels of intelligence in Industry 4.0: from descriptive to prescriptive analytics.

3. Related Works

3.1. Scientific Cleanroom Laboratories

Cleanroom laboratories are specialized workplace environments that are typically used for manufacturing processes or scientific research in various disciplines including semiconductor, nanotechnology, microelectronics, pharmaceuticals, aeronautics, food industry, etc. In scientific industrial research, it is highly important to maintain precise experimental conditions and meticulous control of environmental parameters to ensure accurate experimental results. These environments require minimal contamination from dust, airborne particles, microbes, aerosol particles, chemical vapors and other pollutants to deliver high-quality products and the maximum experimental reproducibility. This is achieved through specialized ventilation systems, the use of high-efficiency particulate air (HEPA) filters and strict procedural protocols. These controls are essential in environments where even microscopic contaminants can affect the integrity of the research or manufacturing process. Governed by stringent guidelines, such as those set by the FDA [19], these facilities provide a meticulously controlled environment. This standardization ensures the optimal performance of sensitive instruments and adherence to environmental conditions like temperature and humidity. For example, in a well-known integrated chip manufacturing laboratory of the Dalian Institute of Semiconductor Technology, the ultra-clean laboratory room is maintained as a 1500 m² hundred-grade room with a temperature of 23 ± 1 °C, relative humidity of $45 \pm 10\%$ and indoor air pressure of +20 Pa, representing the typical conditions of an industrial manufacturing environment [20]. However, beyond these broad regulations, the specific monitoring and analysis of high-end equipment and tools utilized in cleanrooms require a more detailed approach tailored to the needs of individual researchers and lab managers. This fine-grained attention helps in maintaining the integrity and precision necessary for advanced scientific research.

3.2. Anomaly Detection and Classification for Scientific Cleanroom Laboratories

In recent years, there have been several works that have employed machine learning- and deep learning-based anomaly detection techniques for scientific cleanroom laboratory cyber-physical systems. Senselet++ [13] utilizes Singular Spectrum Analysis (SSA)-based real-time anomaly detection, which focuses on searching for known patterns in time series

data. SSA decomposes a time series into a sum of components such as trend, oscillatory components and noise that models time-series data to establish what constitutes a “normal” behavior. In comparing new data against this model, anomalies or deviations from the norm are flagged as anomalies. In another work [21], the authors developed a model-free approach for anomaly detection using the well-known Isolation Forest method. However, none of these methods incorporate concept drift detection or adaption techniques in a non-stationary cyber-physical systems environment where the statistical properties of normal or abnormal behavior may change with time. Table 1 presents a comparative analysis of different anomaly detection models used in CPSs within cleanroom laboratories. It highlights the capabilities of each method, such as online operation, adaptability to concept drift and their effectiveness across various anomaly distributions. TWIN-ADAPT stands out for its adaptability and comprehensive model evaluation in heterogeneous anomaly distributions, showcasing advanced features not present in the other methods listed.

Table 1. Comparative analysis of related works on application of anomaly detection and identification in smart cleanroom laboratories.

Characteristic	SSA Model [13]	Isolation Forest [21]	TWIN-ADAPT
Online	✓	X	✓
Concept drift adaptation	X	X	✓
Model evaluation on different anomaly distributions	X	X	✓
Type of learning	Unsupervised	Unsupervised	Supervised ¹

¹ While TWIN-ADAPT is implemented in a supervised setting, it is extendable to unsupervised settings such as using clustering or density estimation methods to determine drifts instead of classification accuracy in continuous streams of data. However, this is not within the scope of this study.

3.3. Artificial Intelligence-Based Anomaly Detection Models for Digital Twins

Digital twin technology is increasingly recognized as a vital tool in real-time anomaly detection and classification within CPSs, smart manufacturing, industrial plants, aircraft systems [3,22] and other complex control environments [3–5,22–26]. The integration of digital twin allows for the seamless transition from static to dynamic, real-time data processing, enhancing the detection and classification of anomalies before they escalate into more severe issues. This shift is crucial due to the growing complexity and integration of modern systems, which exposes them to new vulnerabilities and attacks. In building and civil infrastructure management, the authors of [27] developed digital twins for anomaly detection, using Bayesian online change-point detection for asset management. A method to detect anomalies by comparing digital twin fingerprints generated from side-channel emissions (acoustic, vibration data) with physical system features was developed [2], but this method depends on the presence of detectable side-channel emissions. Huang et al. [28] introduced a digital twin-driven anomaly detection framework utilizing edge intelligence for early failure detection and maintenance management in industrial systems. In [29], the authors developed a two-phase fault diagnosis method assisted by digital twins, employing deep transfer learning to enhance fault diagnosis in both the development and maintenance phases of manufacturing. Booyse et al. [30] explored a deep digital twin (DDT) for predictive health monitoring and automating maintenance scheduling using operational data directly. In another work [31], the authors present a case study on the application of a machine learning-based digital twin for anomaly detection in small-scale systems, focusing on the omni wheels of soccer robots. They demonstrate that even with limited data, ML-based digital twins can effectively identify anomalies, showcasing their potential beyond large-scale industrial applications. State-of-the-art deep neural network models such as Autoencoders, Multi-Layer Perceptrons, Transforms and CNNs have been heavily explored for anomaly detection and classification tasks in time-series sensor data [6,32,33]. These models can capture complex patterns and improve the accuracy of anomaly detection

in digital twin frameworks. Although these works demonstrate sophisticated anomaly detection and identification frameworks for digital twins by employing advanced machine learning or deep learning techniques, they do not explicitly discuss strategies for handling concept drift wherein new types of anomalies may emerge over time or new ‘normality’ patterns are introduced in the equipment’s operational behavior. This opens avenues for further exploration within the context of digital twin technology, especially to maintain the robustness of anomaly detection systems in dynamic environments where system behavior and conditions evolve over time.

3.4. Concept Drift Detection

Variability in new data from cyber-physical systems can be caused by various factors such as aging sensors, system updates or external events affecting the data distribution, leading to the problem of concept drift. Addressing this requires robust mechanisms for both detection and adaptation [14]. Several methods exist for drift detection. Window-based methods like Adaptive Windowing (ADWIN) [34] use sliding windows to track the mean values of data over time. ADWIN adjusts window sizes dynamically, comparing the means of two sub-windows. If their difference exceeds a set threshold, ADWIN detects a change and resets the window to adapt to the new data conditions. These methods are quick and simple to implement. Performance-based methods, such as the drift detection method (DDM) [35], monitor the error rate of a classifier, adjusting it based on how well the classifier predicts new data [36]. If the classifier correctly predicts an instance, the DDM lowers the error rate; if incorrect, it increases the rate. The DDM signals a warning or identifies a drift when the error rate crosses a predefined threshold. However, these approaches have their drawbacks. Window-based methods might discard useful historical data, which can be critical for maintaining context in data analysis. On the other hand, performance-based methods, while effective in identifying sudden drifts do not perform well for slow and gradual drifts [14,36,37]. Sakurai et al. [38] provide a benchmark evaluation for existing drift detection algorithms, where their experimental results show that the DDM algorithm’s most pronounced limitation is its incapability to detect incremental (or gradual drifts).

3.5. Concept Drift Adaptation

Once drift is detected, it is crucial to effectively manage the observed changes so that the learning model can seamlessly adapt to the new data patterns. There are two broad categories for drift adaptation: incremental learning and online ensemble methods [39]. In the incremental learning strategy, data samples are processed sequentially in order to update the learning model as new data arrive. The Hoeffding Tree (HT) [40] is a variant of the decision tree that offers theoretical guarantees to make incremental drift adjustments for data streams. Unlike traditional decision trees that wait to evaluate the best possible split, HTs calculate the required sample size for choosing a splitting node, allowing for timely updates with incoming data. However, HTs have limitations due to their static internal nodes, which, once set, cannot be altered. Thus, their adaptability to concept drift is confined to the creation of new nodes, limiting their flexibility and responsiveness. To address these shortcomings, the Extremely Fast Decision Tree (EFDT) [41] was developed as an advanced version of the HT. The EFDT improves adaptability by allowing for dynamic adjustments to its decision nodes post creation. This means that an EFDT not only adds new nodes but can also revise existing splits if subsequent data suggest a more effective alternative. This approach enables the EFDT to respond more effectively to concept drifts than its predecessor, though it may still lag in rapid adaptation. Some prominent models for ensemble-based drift adaptation include Adaptive Random Forest (ARF) [42], Streaming Random Patches (SRP) [43] and Leverage Bagging (LB) [44]. ARF and SRP, both utilizing Hoeffding Trees as base learners and incorporating ADWIN for drift detection, aim to dynamically adjust the model ensemble by replacing underperforming trees. While ARF focuses on local subspace randomization, SRP employs a global strategy to enhance learner diversity, often at the cost of increased computational time [37]. On the other hand, the

online ensemble bagging method, LB, primarily utilizes a resampling technique, often using a Poisson distribution to assign random weights to instances in the training data. This approach enhances the diversity of the training samples that each model in the ensemble sees, potentially increasing the robustness and accuracy of the ensemble. However, it falls short in scenarios where data evolve rapidly and unpredictably, as simple resampling may not be sufficient to capture and adapt to new patterns effectively.

3.6. Continuous Learning for Digital Twins

The importance of continual learning in digital twins (DTs) is highlighted in both the lumber industry [45] and dynamic digital twin frameworks over wireless networks [46]. In the lumber industry, Active Learning (AL) methods are leveraged to address the challenges of dynamic environments, ensuring that DTs actively select and learn from new data samples to remain adaptive and synchronized with real-world changes. Similarly, in dynamic digital twins over wireless networks, a novel edge continual learning framework has been proposed to accurately model the evolving affinity between physical and cyber twins, emphasizing synchronization and minimizing de-synchronization, which is crucial for robust and reliable real-world communication. Lombardo et al., in [47], developed a digital twin-based architecture for Intelligent Location-Based Services (I-LBSs) with continual learning (CL), integrating MLOps to manage dynamic updates and adapt to evolving data and task conditions. In summary, a variety of continual learning strategies have been explored in the digital twin space that are essential in maintaining the accuracy and relevance of DTs across space and time. This emphasizes the need for efficient continual learning strategies for digital twins, particularly in the context of anomaly detection and identification in real-time streams, which is critical for industrial cyber-physical systems that are susceptible to data distribution changes.

4. Methodology

The end-to-end workflow for TWIN-ADAPT comprises dataset preparation to simulate different types of drift behavior, followed by the choice of different off-the-shelf models for online drift detection and adaption to compare their performances with that of our proposed TWIN-ADAPT's continuous learning model. Model optimization is an integral part of TWIN-ADAPT's continuous learning framework to select optimal values for different thresholds and window sizes.

4.1. Dataset Description

In the Senselet++ platform [13], an IoT-driven end-to-end sensing platform was designed for the University of Illinois Urbana-Champaign's (UIUC) scientific cleanroom laboratories (Figure 5 shows a cleanroom semiconductor lab at UIUC). This IoT sensing platform consists of a CPS for the critical high-end components, including lab instruments and infrastructure, within the cleanroom laboratory. It monitors various physical factors such as temperature (in Celsius), humidity (in %), pressure (in Bar) and rate of airflow (in meters/second to measure quantity of air being moved) using low-cost sensors. Each cleanroom component is equipped with its own CPS, comprising a data acquisition system tailored to the specific requirements of the component. For more details on the data acquisition unit, please refer to our previous work on Senselet++ [13]. Additionally, it is important to note that our current work is not focused on building the IoT system itself. Instead, our approach was designed to work with any existing IoT-driven CPS platform that exhibits the existence of concept drift, which poses considerable challenges in developing ML models, as their learning performance may progressively degrade due to data distribution changes. Subsequently, our aim is to demonstrate a framework for an advanced, online adaptive learning model that can detect and react to concept drift that occurs in IoT-driven CPS data streams. We narrow down the scope of the extensive CPS subsystems to the three most important components of cleanrooms: the Fumehood, Vacuum Pump, and Karl Suss

Lithography Unit. The CPS connected to each of these cleanroom components collects time-series data from various sensors, as discussed below:

- **Fumehood:** Data from temperature, humidity, and airflow are recorded, providing a comprehensive view of the environmental conditions within the Fumehood.
- **Vacuum Pump:** Temperature sensor data (in Celsius) are aggregated to monitor the operational health and detect any thermal anomalies that may indicate malfunctions or efficiency issues.
- **Karl Suss Lithography Unit:** Both temperature (in Celsius) and humidity sensor data (in %) are gathered to ensure that the system operates within optimal conditions for precise manufacturing processes.

These collected data form the basis for developing digital twins for each component's CPS, enabling real-time drift detection and continuous adaptive learning for highly precise anomaly classification. Figure 6a–c illustrate the distribution of data for the Fumehood's temperature sensor and humidity sensor and the Vacuum Pump's temperature data.

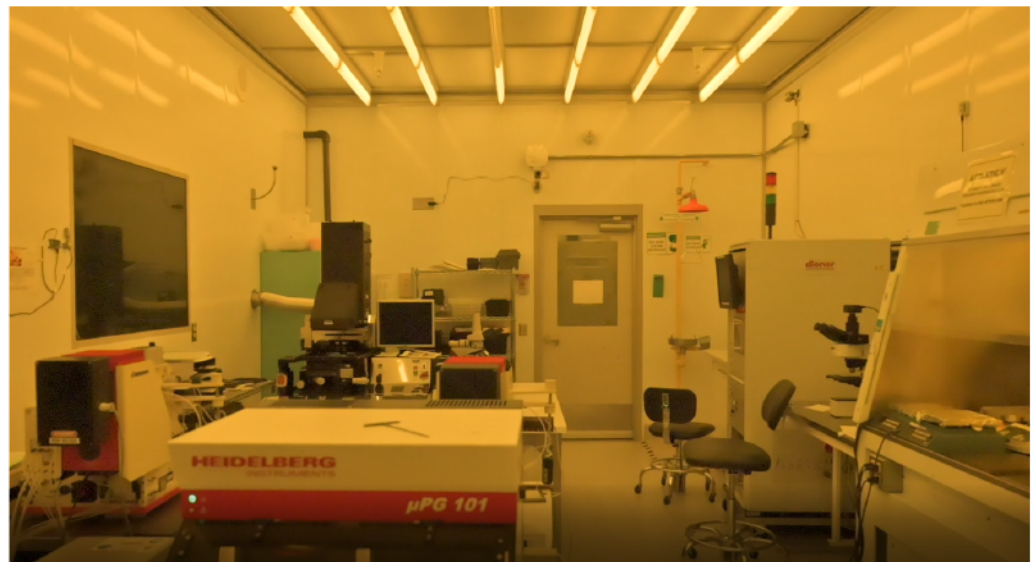


Figure 5. An overview of a scientific semiconductor laboratory (cleanroom) at UIUC.

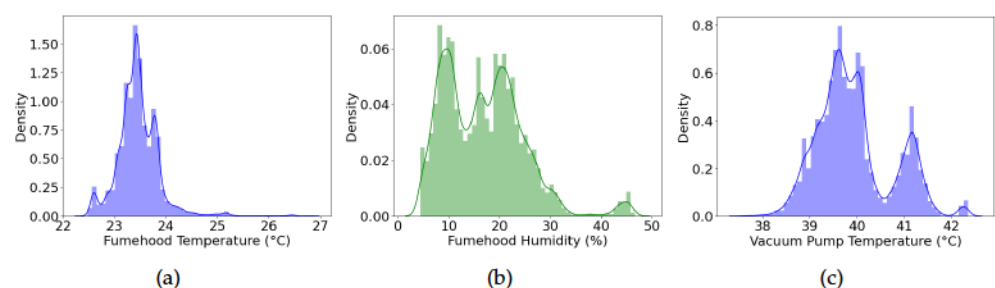


Figure 6. Sensor data distribution over one month for (a) Fumehood temperature, (b) Fumehood humidity and (c) Vacuum Pump temperature.

4.2. Dataset Preparation with Anomaly Injection

To simulate drift in the equipment's CPS behavior, we inject synthetic noise into the sensor data values collected for each component of the cleanroom's CPSs. Evaluating the performances of adaptive learning models requires distinct differences between training and testing data, which demonstrate changes in drift distributions. Consequently, we introduce one type of noise within the training data and a different noise distribution for the testing data. In the training dataset, Gaussian white noise is injected with a standard deviation (σ_N) of 0.1 to emulate anomalous operational variations. For the testing dataset,

a combination of Gaussian and Uniform distribution anomalies is introduced, each with a standard deviation (σ_N) of 0.5. This mixed approach in the testing set is designed to challenge the classifiers' ability to distinguish between normal operations and anomalies, where the latter follows a different distribution in the training and testing dataset.

Using a mixed-noise model (Gaussian and Uniform) in the testing data, we ensure a variety of anomaly profiles: some will be obvious due to the higher deviation of the Uniform distribution, while others will closely resemble the training dataset's anomalies (Gaussian), albeit with a different standard deviation. This setup provides a comprehensive assessment of the classifier's performance, evaluating its ability to detect anomalies across various thresholds. It tests the classifier's robustness against clear anomalies and its precision in distinguishing between normal and anomalous conditions in a more ambiguous and realistic operational environment. Figure 7 illustrates the probability density functions of the added noise distributions. For each type of distribution, the mean (expected) value of the noise is set to zero, and the standard deviation varies from 0.1 to 0.5 to simulate different noise levels.

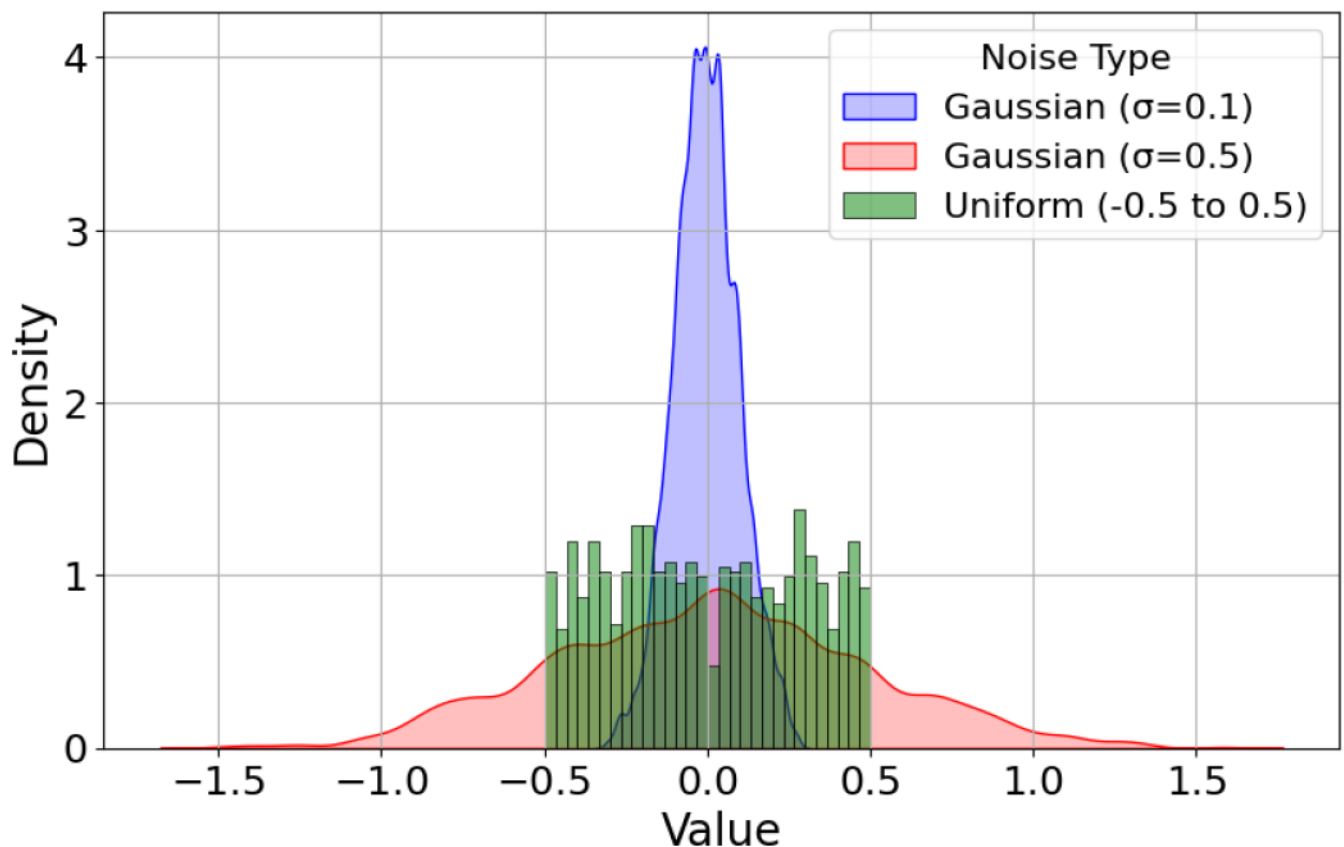


Figure 7. Probability density functions of the noise injected to simulate anomalies.

Two hyperparameters control the extent of anomalies in both the training and testing sets: the noise ratio and the noise level. Specifically, these are defined as follows:

- **Noise Ratio:** Determines the proportion of data points in the dataset that are affected by noise.
- **Noise Level:** Determines the intensity of the noise injected into the data. A higher noise level means that the anomalies will deviate more significantly from the original data points. For example, if $\text{noise_level} = 0.1$, the anomalies will be relatively close to the original data, while $\text{noise_level} = 0.5$ will introduce larger deviations. This parameter is used directly as the standard deviation for the Gaussian and Uniform distributions of injected noise.

For instance, in the Vacuum Pump dataset, which primarily involves temperature readings (a single-feature dataset), even a slight increase in noise level significantly degrades the model performance, due to the model's sensitivity to small deviations in a less complex feature space. Conversely, the Fumehood, equipped with multiple sensors for temperature, humidity and airflow, requires a more substantial injection of anomaly distribution to induce concept drift. This is because the presence of multiple features dilutes the impact of noise on a single sensor, necessitating stronger noise ratios and levels in the multiple-feature (multiple sensors) dataset in order to challenge the model's performance during the test phase. Then, Table 2 shows different values set for the noise level and noise ratio for each type of cyber-physical system's training and testing dataset.

Table 2. Type of noise distribution, noise level and noise ratio for training and testing datasets.

CPS Datasets	Training Data			Testing Data		
	Noise Distribution	Noise Level (σ_N)	Noise Ratio	Noise Distribution	Noise Level (σ_N)	Noise Ratio
Vacuum Pump	Gaussian	0.1	10%	Gaussian and Uniform	0.5	35%
Lithography Unit	Gaussian	0.1	10%	Gaussian and Uniform	0.5	65%
Fumehood	Gaussian	0.1	10%	Gaussian and Uniform	0.5	80%

4.3. Model Selection for Online Drift Detection and Adaptation

In our work, we present TWIN-ADAPT, a continuous learning framework for online anomaly classification using digital twins. This framework incorporates advanced machine learning models, optimization and adaptive algorithms to maintain the precision and effectiveness of digital twins in classifying real-time anomalies. It is worth mentioning that each digital twin is a specialized model designed for anomaly classification specific to its respective CPS component of the cleanroom. For instance, the digital twin of the Fumehood classifies anomalies in multivariate data, including temperature, humidity and airflow, which are vital for maintaining stringent cleanroom standards. The digital twin for the Vacuum Pump focuses solely on the temperature data for anomaly classification to capture essential thermal dynamics of the Pump, while the digital twin for the Karl Suss Lithography Unit integrates both temperature and humidity data, crucial for accurate microfabrication process control.

Specifically, the TWIN-ADAPT framework integrates the LightGBM model into the digital twin of each CPS component in order to enhance the system's responsiveness and adaptability to dynamic environments. This robust machine learning model, an off-the-shelf gradient boosting decision tree (GBDT) system, is distinguished by its fast training speed, low memory requirements, superior accuracy and parallel processing capabilities. These attributes make LightGBM especially well suited for handling the vast and complex data streams inherent in cleanroom operations, where small fluctuations in equipment activity can significantly influence operational efficiency [14,48].

LightGBM's unique capabilities stem from its innovative Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). GOSS enhances model training efficiency by focusing on data samples with large gradients that are more informative for creating predictive models, thereby reducing the computational burden without compromising model accuracy [49]. Concurrently, EFB optimizes feature handling by bundling mutually exclusive features, significantly reducing the dimensionality of the data, which is crucial for processing the high-dimensional data typically found in cleanroom's cyber-physical system sensors. These features of LightGBM are crucial for maintaining the digital twin's accuracy and reliability over time, enabling it to effectively detect and adapt to concept drift within the data streams. In coupling LightGBM with a dual windowing

strategy consisting of a sliding window and an adaptive window, the model is equipped to dynamically adjust its parameters in response to changing data distributions. The sliding window monitors the model's performance on recent data, while the adaptive window accumulates new data to update the model when drift is detected. The algorithm operates by comparing the accuracy of the current observation window with that of the previous window. If the accuracy drops below a certain threshold, it indicates potential drift, prompting the system to adapt (see details for Algorithm 1 in Section 5.1). This ensures that the digital twin continues to learn and evolve, maintaining a high anomaly classification accuracy even as new patterns emerge within the cleanroom's cyber-physical systems' operational data. Thus, compared to other online learning models, an optimized and adaptive LightGBM model within its digital twin framework offers an efficient and scalable solution for the continuous and real-time demands of cleanroom monitoring. This empowers the continuous learning model as an indispensable tool in the domain of the preventive and predictive maintenance of cyber-physical systems.

In addition to TWIN-ADAPT's optimized and adaptive LightGBM model, we consider several existing online drift detection and learning models to serve as baselines for comparison. These online learning models include the following:

- **ARF-ADWIN:** Adaptive Random Forest combined with ADWIN drift detection.
- **ARF-DDM:** Adaptive Random Forest combined with DDM drift detection.
- **SRP-ADWIN:** Streaming Random Patches model combined with ADWIN drift detection.
- **SRP-DDM:** Streaming Random Patches model combined with DDM drift detection.
- **HTs:** Hoeffding Trees that incrementally adapt to data streams using the Hoeffding bound to determine the necessary samples for node splitting.
- **EFDT:** Extremely Fast Decision Trees quickly adapt to concept drifts by splitting nodes upon reaching confidence levels.
- **LB:** Leverage Bagging uses bootstrap samples and the Poisson distribution to construct diverse online ensembles.

These models are selected for comparison due to their strong adaptability to concept drift and their robust data stream analysis capabilities. Both ARF and SRP are state-of-the-art drift adaptation methods that demonstrated superior performances in handling concept drift, as demonstrated in various experimental studies. Both methods are online ensemble models constructed with multiple HTs, which provide robust incremental learning capabilities. They do not require the tuning of data chunk sizes, which can lead to delays in drift detection and increased execution times in block-based ensembles. The choices of drift detection methods—ADWIN and DDM—are strategic: ADWIN excels at detecting gradual drifts, while DDM is more effective for sudden drifts. By combining these drift detection methods with ARF and SRP, we ensure that our base learners can handle both types of drifts efficiently. Additionally, three well-known individual drift adaptation models—EFDT, HT and LB provide a diverse range of mechanisms to adjust to changing data dynamics, making them ideal benchmarks for assessing their relative performances with our custom combination of online learning models and TWIN-ADAPT. Specifically, EFDT's rapid adaptation through immediate node splitting offers a contrast to more gradual drift adjustments, while HT's incremental learning approach and LB's unique leveraging of bootstrap samples offers unique perspectives on managing data variability and drift. By comparing these established models with TWIN-ADAPT's optimized and continuous learning-driven LightGBM model, we aim to demonstrate the effectiveness of our approach in classifying anomalies under varying conditions and drift scenarios in evolving data streams.

4.4. Model Optimization

To optimize the anomaly classifier model's (LightGBM) hyperparameters, drift detection and adaptation thresholds and sliding window sizes in TWIN-ADAPT, we employ Particle Swarm Optimization (PSO). PSO, well known for its efficient global search capability and speed, is an ideal choice for this task due to its ability to operate without the

need for gradient information, making it well suited for complex, non-linear optimization problems often found in real-world applications [49,50].

The integration of PSO in our system focuses on optimizing several critical hyperparameters of the LightGBM model, such as the number of tree leaves, maximum tree depth, minimum number of data samples in one leaf, learning rate and number of base learners. By adjusting these parameters, the model can be finely tuned to accurately model the data streams from various cleanroom CPS components, each represented by a dedicated digital twin. Our implementation uses PSO to iteratively explore the hyperparameter space. Each particle in the swarm represents a potential solution, i.e., a set of hyperparameters, and moves through the solution space by updating its velocity and position based on its own experience and that of its neighbors. This collective intelligence approach helps quickly converge to the best solution, which is the set of hyperparameters that result in the highest predictive accuracy for our LightGBM model. This approach not only improves the performance of the LightGBM classifier model in detecting and adapting to concept drift in the data streams of cleanroom's CPS but also enhances the model's efficiency by reducing computational overhead.

In addition to optimizing the LightGBM model parameters, PSO is also utilized to optimize the hyperparameters of TWIN-ADAPT's adaptive and sliding window algorithm. This includes finding the optimal values for the thresholds used to determine drift and trigger adaptation by comparing the model accuracy across consecutive windows of samples in a data stream (refer to Algorithm 2 in Section 5.1).

5. System Overview

The proposed TWIN-ADAPT system comprises an adaptive continuous learning-driven *digital twin* specifically designed for anomaly classification on real-time streaming data from a cleanroom's cyber-physical system components such as the Vacuum Pump, Fumehood and Photolithography Unit. Each component's monitoring model (i.e., the anomaly classifier model or the learner) is represented by its digital twin, comprising both offline and online operational stages. Figure 8 presents an overview of the TWIN-ADAPT framework.

During the offline stage, data from the CPS sensors of that component are aggregated to create a historical dataset, which is utilized to train an (offline) initial model using LightGBM. This model undergoes refinement through hyperparameter optimization using Particle Swarm Optimization (PSO), ensuring that it is finely tuned to the unique characteristics of each CPS component's operational data. In the online stage, the digital twin model processes live data streams that are continuously generated by the cyber-physical system (sensors) of the cleanroom component. Initially, the LightGBM model trained during the offline phase is employed to monitor these data streams for anomalies. Upon the detection of concept drift in the streaming data—signifying operational changes or new anomaly patterns—by the optimized adaptive and sliding windowing method, the proposed continuous learning model collects new data samples through the adaptive window. These samples are reflective of the latest operational states of the CPS. The model is then retrained on these new data to accurately align with the latest patterns, ensuring that each component's digital twin dynamically adapts to maintain effective anomaly classification and continuous operational monitoring in the non-stationary environment of cleanrooms.

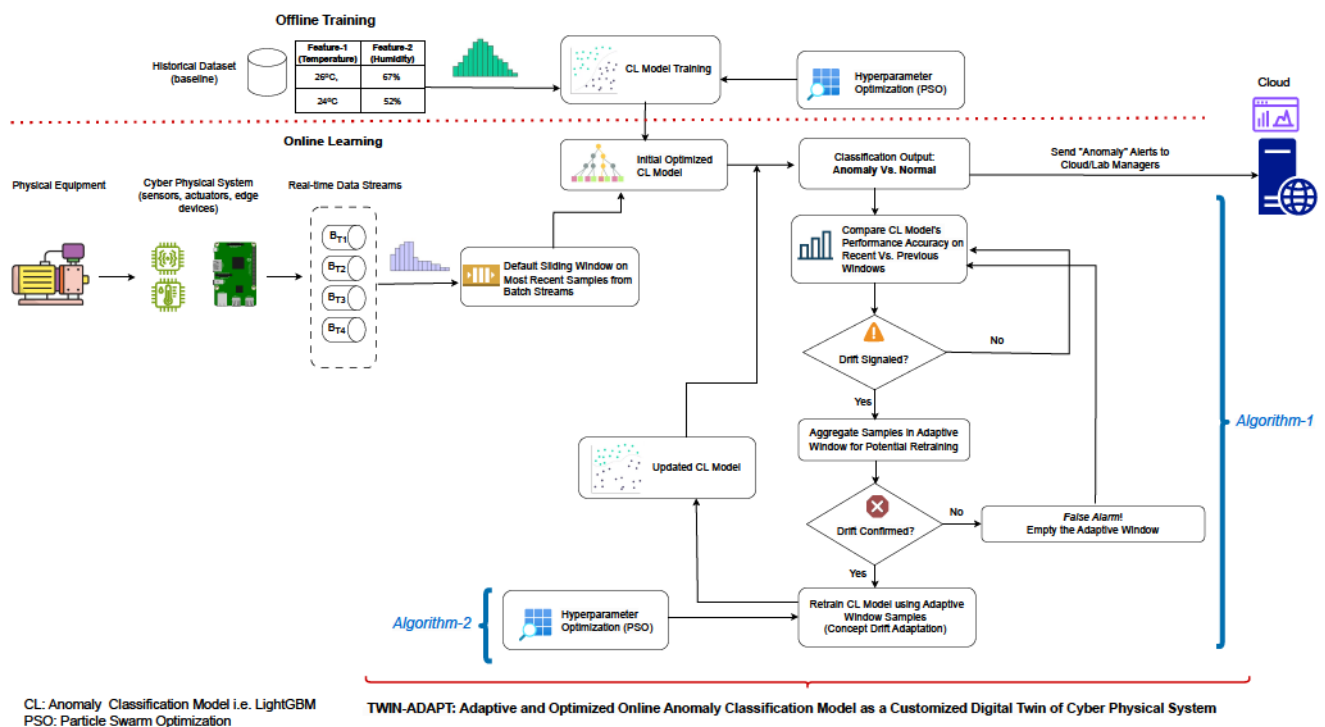


Figure 8. System overview of TWIN-ADAPT.

5.1. TWIN-ADAPT: Continuous Learning for Digital Twin-Enabled Online Anomaly Classification

TWIN-ADAPT integrates two pivotal algorithms: *Continuous Learning Model using Adaptive and Sliding Window (Algorithm 1)* and *Hyperparameter Optimization for Continuous Learning Model (Algorithm 2)*. The former dynamically adjusts to changing data from cleanroom equipment sensors, while the latter fine-tunes the model settings using Particle Swarm Optimization (PSO) for optimal performance. The continuous learning strategy for real-time, adaptive anomaly detection in TWIN-ADAPT assigns a dedicated digital twin to each critical CPS component of the cleanroom laboratory. This empowers precise preventive monitoring and analytics under heterogeneous data distribution settings. Similar to many sequential learning algorithms, TWIN-ADAPT's adaptive and sliding window framework operates in two distinct phases: the offline learning phase and the online continuous learning phase. During the initial offline learning phase, a static classifier model (*LightGBM*) is trained on historical sensor data. In the subsequent online continuous learning phase, the pre-trained model's performance is continuously monitored. The model's performance metrics are compared against previous observations, and the model is updated using new concept data as they become available.

The continuous learning algorithm (*Algorithm 1*) operates by monitoring the performance of a pre-trained model on incoming data and making adjustments as necessary to maintain the prediction accuracy. The algorithm utilizes a dual-window strategy that employs both adaptive and sliding windows. The sliding window is used for immediate change detection, while the adaptive window accumulates data necessary for retraining the models. The process begins by initializing an adaptive window *AdaptWin* and setting the system *CurrentState* to a normal state. As new data points arrive in a streaming batch of b samples, the last few samples from the batch are collected into an observation window *ObsWin* consisting of a fixed size *winSize* such that *winSize* is a subset of b , specifically designed to capture the most recent data points within the batch. The accuracy of the current observation window *ObsWin* is compared with the accuracy of the previous window *PrevAcc* consisting of $b - \text{winSize}$ samples to detect any significant drops in performance between the consecutive windows. In the normal state, if the accuracy drops below a

defined threshold (γ), it indicates a potential drift, and the system transitions to a warning state (lines: 5 to 11). In this warning state, new data samples are collected into the adaptive window *AdaptWin* (this process is illustrated in Figure 9). If the accuracy continues to drop further, confirming the drift (δ), the model is updated using the samples in the adaptive window, and the system returns to the normal state after clearing the adaptive window (lines: 13 to 20). If the accuracy stabilizes before confirming the drift, indicating a false alarm, the adaptive window is cleared without updating the model, and the system returns to the normal state (lines: 21 to 24).

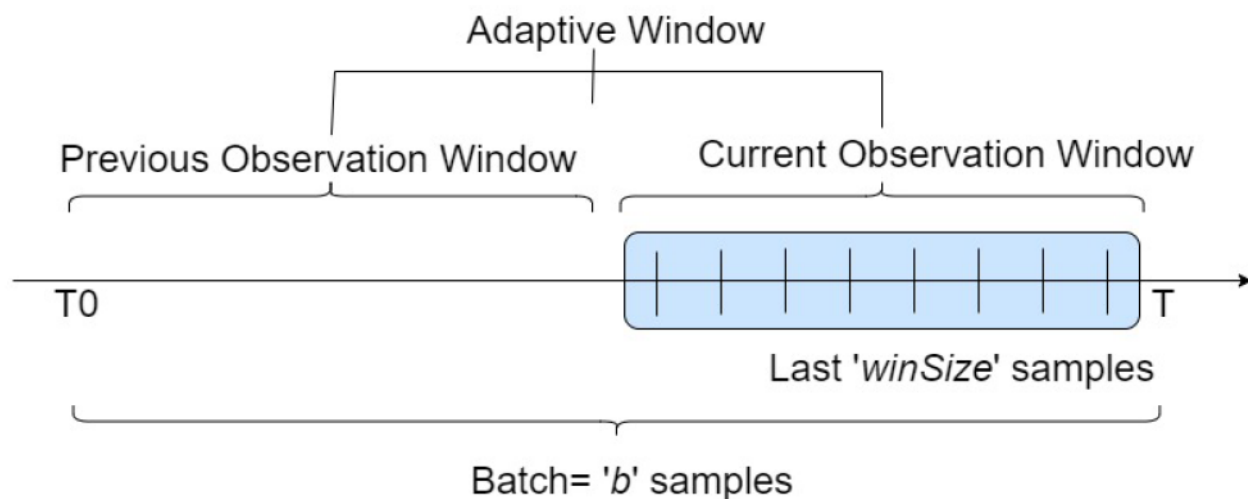


Figure 9. Continuous learning process for Algorithm 1. The “current” sliding observation window (*ObsWin*) captures the performance metric for short concepts, and the previous observation window (*PrevObsWin*) captures the feature–target relationship over the last complete window. In comparing accuracy between two windows, the adaptive window (*AdaptWin*) starts to collect data samples as new concept samples for potential drift adaptation.

The rationale behind aggregating samples in the adaptive window is to identify abrupt changes in the data while being able to ignore spurious drifts. This approach ensures that only significant and persistent changes trigger model updates, reducing the overhead of adaptations to minor or transient data variations. The algorithm also includes a state for monitoring the model performance in the drift state (State 2) to check if further adaptation is required. If the model reaches State 2, it continues to collect new samples to compute the accuracy of the current observation window *CurrentAcc* and compare it with the accuracy from the drift starting point (new concept accuracy, *NewConceptAcc*). If this accuracy is below the warning level of the new concept, it indicates that the model is outdated and needs retraining with the new concept samples. The learner will be updated again on the samples in the adaptive window, and the system will transition to the normal state, emptying the adaptive window (lines: 35 to 41 of Algorithm 1). Similarly, the model will also continue collecting samples for updating if the size of the adaptive window has reached its maximum limit to ensure that the real-time constraints of memory and processing speed are met.

Algorithm 1 Continuous Learning Model using Adaptive and Sliding Windows

Input: *DataFlow*: stream of sensor data from CPS component
 γ : threshold for initiating detection, δ : threshold for adaptation
 $winSize$: size of the observation window (for detection), $winSizeMax$: maximum size of the data collection window (for adaptation)
 $Model$: pre-trained model on historical data
Output: New average accuracy of $Model$

```

1: function ADAPTMODEL(DataFlow,  $Model$ ,  $\gamma$ ,  $\delta$ ,  $winSize$ ,  $winSizeMax$ )
2:    $AdaptWin \leftarrow \emptyset$ 
3:    $CurrentState \leftarrow 0$  ▷ 0: Normal, 1: Warning, 2: Adaptation
4:   for  $batch = b$  samples  $\in DataFlow$  do
5:      $ObsWin \leftarrow$  last  $winSize$  samples from batch  $b$ 
6:      $PrevObsWin \leftarrow b - winSize$  samples
7:      $CurrAcc \leftarrow$  calculate\_accuracy( $ObsWin$ )
8:      $PrevAcc \leftarrow$  calculate\_accuracy( $PrevObsWin$ )
9:     if  $CurrentState == 0$  and  $CurrAcc < \gamma \times PrevAcc$  then
10:       $AdaptWin \leftarrow AdaptWin \cup \{batch\}$ 
11:       $CurrentState \leftarrow 1$  ▷ Enter warning state
12:    end if
13:    if  $CurrentState == 1$  then
14:       $AdaptWinLen \leftarrow |AdaptWin|$ 
15:      if  $CurrAcc < \delta \times PrevAcc$  then
16:        ▷ Accuracy of current observation window drops to drift level
17:         $CurrentState \leftarrow 2$ 
18:         $f \leftarrow CurrAcc$  ▷ Record accuracy at drift starting point
19:         $Model \leftarrow$  UpdateModel( $Model$ ,  $AdaptWin$ )
20:        ▷ Update the model by retraining it on new concept samples
21:      else if  $CurrAcc > \gamma \times PrevAcc$  or  $AdaptWinLen == winSizeMax$  then
22:        ▷ False alarm, accuracy increased back or max window size reached
23:         $AdaptWin \leftarrow \emptyset$ 
24:         $CurrentState \leftarrow 0$ 
25:      else
26:         $AdaptWin \leftarrow AdaptWin \cup \{batch\}$ 
27:      end if
28:    end if
29:    if  $CurrentState == 2$  then ▷ Within drift state
30:       $AdaptWinLen \leftarrow |AdaptWin|$ 
31:      if  $AdaptWinLen \geq winSizeMax$  then
32:        ▷ If sufficient new concept samples are collected
33:         $NewConceptAcc \leftarrow$  calculate\_accuracy( $ObsWin[f : f + winSize]$ )
34:        ▷ Calculate accuracy on new concept window starting from index  $f$ 
35:        if  $CurrAcc < \delta \times NewConceptAcc$  OR  $AdaptWinLen == winSizeMax$ 
36:          ▷ When new concept accuracy drops to the warning level or sufficient
37:          new concept samples are collected
38:           $Model \leftarrow$  UpdateModel( $Model$ ,  $AdaptWin$ )
39:          ▷ Retrain the classifier on all the newly collected samples
40:           $AdaptWin \leftarrow \emptyset$  ▷ Empty new concept window
41:           $CurrentState \leftarrow 0$  ▷ Change to a normal state
42:        else
43:           $AdaptWin \leftarrow AdaptWin \cup \{batch\}$  ▷ Keep collecting new samples
44:        end if
45:      end if
46:    end if
47:  end for
48:  return New Average Accuracy of  $Model$ 
49: end function

```

Algorithm 2 Hyperparameter Optimization for Continuous Learning Model

Input: *DataFlow*: stream of sensor data, *ConfigSpace*: space for Hyperparameter tuning, *TimeLimit*: maximum duration for optimization
Output: *OptimalParams*: detected optimal hyperparameter values, *BestAccuracy*: best accuracy achieved

```

1: function HYPERPARAMETEROPTIMIZATION(DataFlow, ConfigSpace, TimeLimit)
2:   HighestAcc  $\leftarrow$  0
3:   for i  $\leftarrow$  1 to TimeLimit do
4:      $\gamma, \delta, \text{winSize}, \text{winSizeMax} \leftarrow \text{OptimizeConfig}(\text{ConfigSpace})$ 
5:     Acc  $\leftarrow \text{AdaptModel}(\text{DataFlow}, \text{Model}, \gamma, \delta, \text{winSize}, \text{winSizeMax})$ 
6:     if Acc > HighestAcc then
7:       HighestAcc  $\leftarrow$  Acc
8:       OptimalParams  $\leftarrow$  ( $\gamma, \delta, \text{winSize}, \text{winSizeMax}$ )
9:     end if
10:  end for
11:  return HighestAcc, OptimalParams
12: end function

```

To ensure the model's effectiveness, a hyperparameter optimization function is implemented. Subsequently, the Hyperparameter Optimization function in Algorithm 2 iteratively tunes the thresholds and window sizes within a specified time limit. Specifically, the digital twin model integrates Particle Swarm Optimization (PSO) within its Hyperparameter Tuning function. This allows for the fine-tuning of key model parameters like window sizes and sensitivity thresholds. In customizing hyper parameters such as *windowSize*, *maxWindowSize* and the detection thresholds (γ and δ), TWIN-ADAPT can tailor the model's responsiveness by finding the optimal values of hyperparameters that can construct a highly accurate model. For each hyperparameter configuration space, the optimization function runs the continuous learning model function and evaluates the model's accuracy (lines: 3 to 10 in Algorithm 2). The highest accuracy achieved and the corresponding optimal parameters are stored and returned at the end of the optimization process to generate an optimization anomaly classification model for the CPS. Specifically, the parameters for optimization include the following:

- **Drift detection threshold (γ):** This threshold is used to compare the accuracy of the current observation window (with the most recent *winSize* samples) with the previous window (containing $b - \text{winSize}$ samples from a batch of b samples). If the accuracy drops below γ times the previous accuracy, it indicates potential drift. Optimizing γ helps in accurately detecting when the data distribution changes.
- **Adaptation threshold (δ):** Once drift is detected, this threshold determines whether the model should be adapted. If the accuracy of the current window of the most recent samples drops below δ times the previous accuracy, adaptation is triggered. Optimizing δ ensures that the model adapts appropriately to maintain performance. A small adaptation threshold can lead to frequent retraining, harming the overall prediction performance because it causes the model to react too sensitively to minor fluctuations, resulting in a high false alarm rate.
- **Window sizes (*winSize* and *winSizeMax*):** The size of the observation window (*winSize*) and the maximum size of the adaptive window (*winSizeMax*) are crucial for the algorithm's performance. The observation window size affects how quickly changes in data distribution are detected, while the adaptive window size determines how much historical data are considered for retraining the model. A larger adaptive window size implies a greater capacity for the model to retain past information, while a smaller size facilitates quicker adaptation to recent changes. The *winSizeMax* value sets the sensitivity to the forgetting rule, dictating when the adaptive window should be emptied, effectively resetting the model's memory.

This strategic use of adaptive and sliding windows coupled with hyperparameter optimization makes TWIN-ADAPT an advanced system capable of maintaining the high standards required in smart laboratory operations. This approach allows the model to dynamically adapt to changes in data patterns, maintaining high prediction accuracy even in the presence of concept drift. By continuously monitoring performance and adjusting the model as needed, the algorithm ensures robust and reliable analytics for dynamic distributions in cleanroom environments.

5.2. Experimental Setup

In the experimental setup, we use the *River* Python module to implement and evaluate various tree-based models and algorithms on evolving data streams. Python has been widely adopted in scientific development in recent years due to its versatility, extensive support for libraries in machine learning and active community support. Therefore, it is used in this study. RiverML provides a robust framework for continuous learning and concept drift detection, which are essential for handling real-time data streams. Three CPS datasets are considered for the anomaly injection and adaptive online classification models: Fumehood data, Vacuum Pump data and Lithography data. The data for each CPS component consists of time-series data collected over a period of one month from different sensors attached to each CPS, as discussed in Section 4.1. Each dataset is split into 70% for training and 30% for testing. The raw data collected from the CPS sensors are considered normal, and to simulate drifts with anomalies, we inject different distributions of anomalies into the training and testing data as discussed in Section 4.2. The datasets are treated as binary datasets with two labels: “normal” for the unaltered data and “abnormal” for the samples where noise was injected in either the training or testing set.

To evaluate the performance of the proposed TWIN-ADAPT’s adaptive anomaly classification framework, we use hold-out validation. The model is trained on the training set and then evaluated on the test set. Given that the datasets are unbalanced, we utilize four different metrics to assess the model’s performance: accuracy, precision, recall and F1 score. These metrics provide a comprehensive evaluation of the model’s ability to distinguish between normal and abnormal states, reflecting its effectiveness in identifying CPS anomalies and attacks in real time.

6. Experimental Results

In this section, we present the results of various experiments, including anomaly injection in the training and testing datasets, the performances of static (non-adaptive) models on noise-induced datasets, the performances of baseline drift adaptation models and the performance of TWIN-ADAPT compared to the offline model.

6.1. Anomaly Injection in Datasets

Figure 10 shows the distribution of anomalies and normal data for temperature, humidity and airflow features of the Fumehood subsystem in both the training and testing datasets. As expected, the anomalies (in red) are more frequent in the testing set compared to the training set for each feature. Additionally, the anomalies in the testing dataset exhibit a broader distribution, indicating greater variance in anomaly values compared to the training dataset for each feature.

Figures 11 and 12 illustrate the distribution of temperature and humidity values in the training and testing datasets for the Vacuum Pump and Lithography Unit CPSs, respectively, distinguishing between normal readings and anomalies. For both CPS components of the cleanroom, the training datasets show anomalies concentrated around specific values, while the testing datasets exhibit a broader, more dispersed anomaly distribution. This shift indicates changes in operational conditions or anomaly patterns between the training and testing phases. The normal data distribution in the training datasets has pronounced peaks, whereas the testing datasets show a flatter distribution.

Table 3 represents the Kolmogorov–Smirnov (KS) test’s p -values used to quantify the difference in anomaly distributions between the training and testing datasets for each feature of the three CPS components (Fumehood, Vacuum Pump and Lithography Unit). The KS statistic is a non-parametric statistical method for measuring the equality between two probability distributions with different shapes [51]. A p -value from the KS test indicates the probability that the observed differences between the distributions could have occurred by chance. In the literature [52–55], a p -value less than a conventional threshold (commonly 0.05 or 0.01) suggests a significant difference between the two distributions, implying that the samples do not come from the same distribution. In Table 3, the results show significant differences in the distributions of anomalies, particularly in features like temperature and airflow, where very low p -values indicate substantial discrepancies, indicating the existence of concept drift between anomalies of the training and testing datasets. These findings highlight the impact of different noise injection methods on the anomaly distribution, which is crucial for evaluating the performance and sensitivity of adaptive drift adaptation models for accurate anomaly classification models under concept drift conditions.

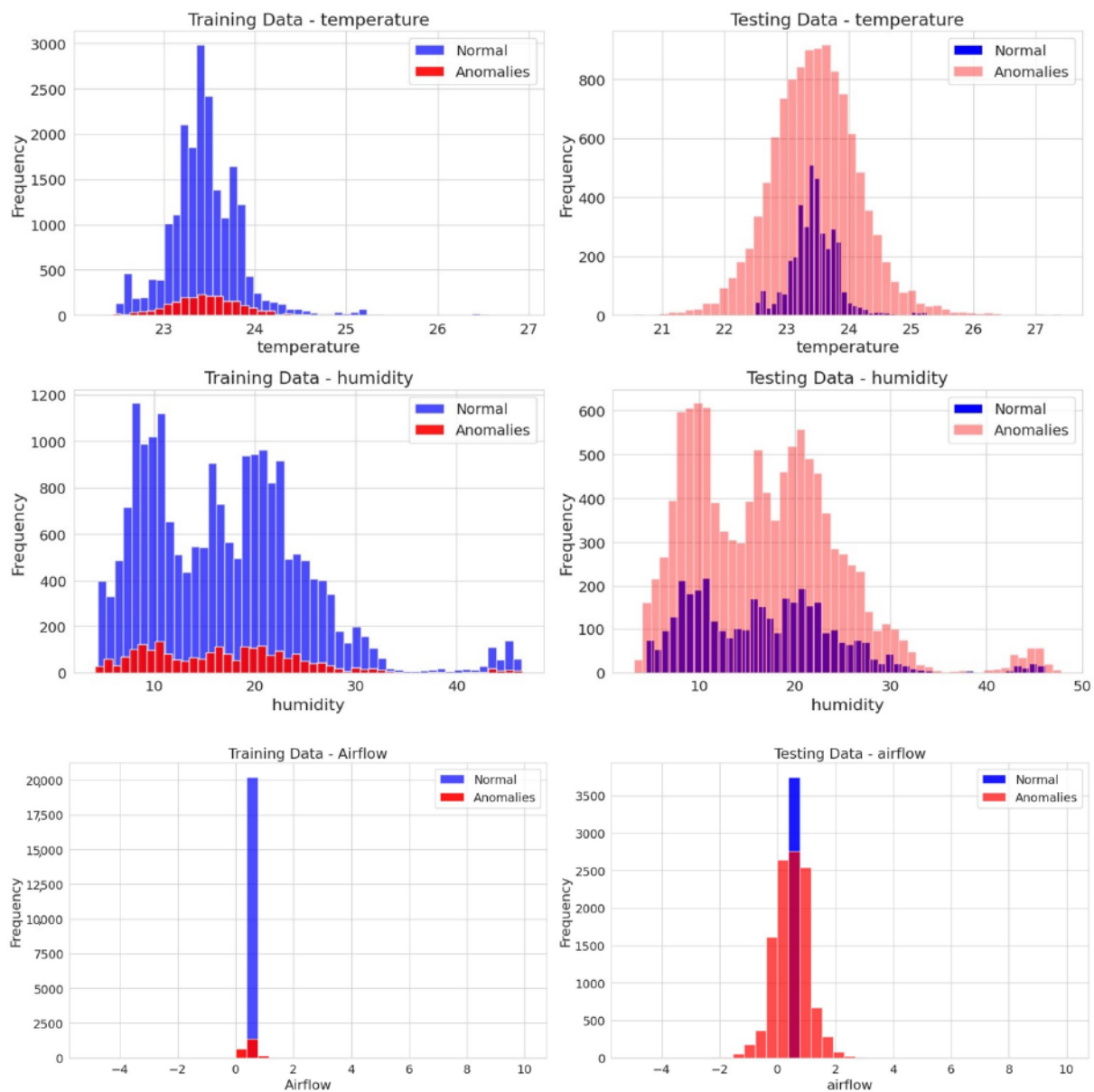


Figure 10. Distribution of anomalies and normal data for different Fumehood features (temperature, humidity and airflow) in both training and testing datasets.

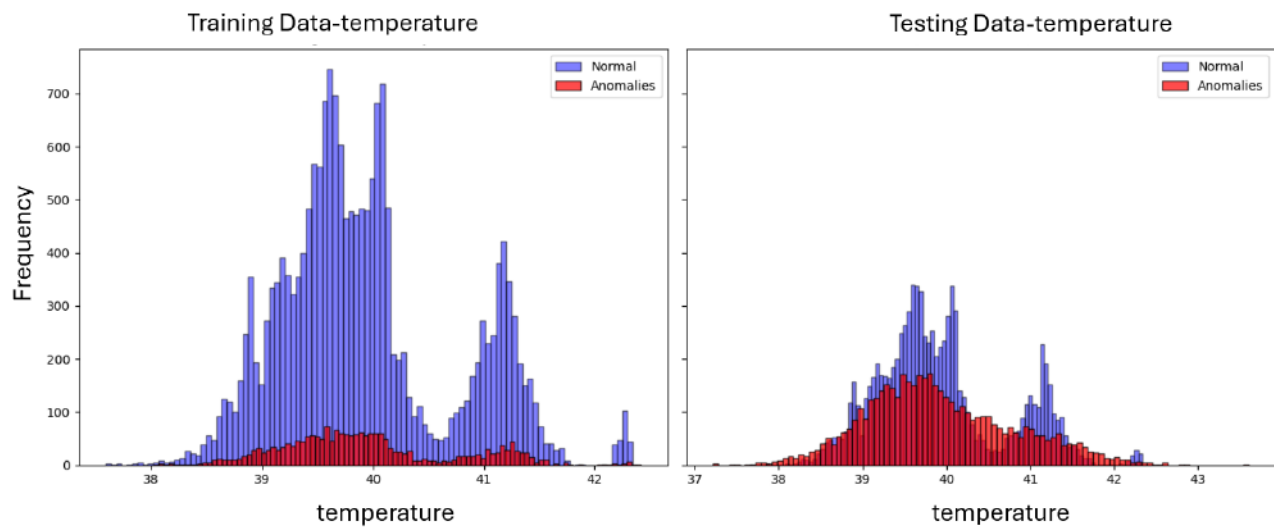


Figure 11. Distribution of anomalies and normal data for Vacuum Pump's feature (temperature) in both training and testing datasets.

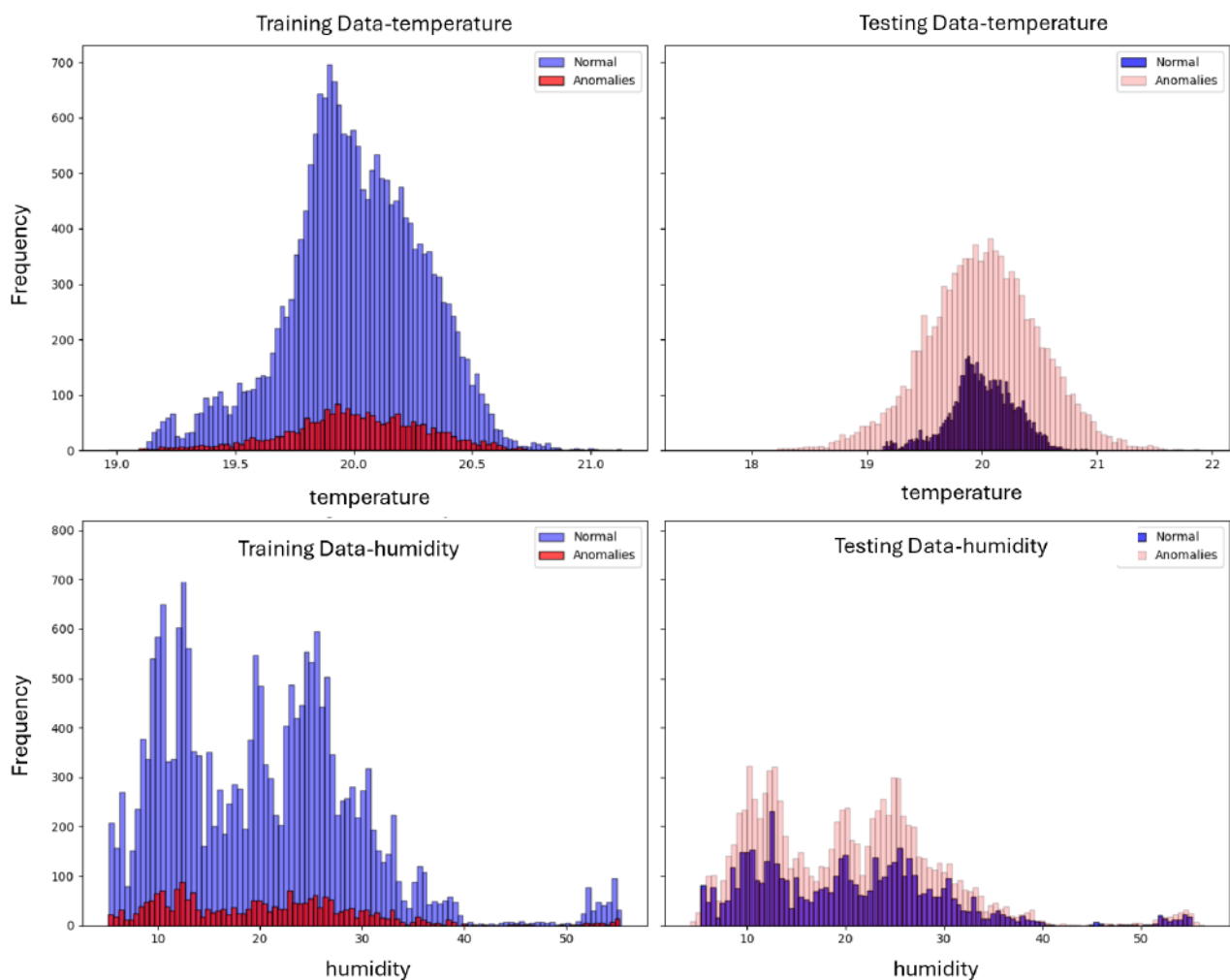


Figure 12. Distribution of anomalies and normal data for different Lithography Unit's features (temperature and humidity) in both training and testing datasets.

Table 3. *p*-values for KS-Test to measure difference in anomaly distributions between training and testing data for each feature of CPS dataset.

CPS Dataset	Feature	<i>p</i> -Value
Fumehood	Temperature	0.00066
	Humidity	0.016
	Airflow	0.001
Vacuum Pump	Temperature	0.0028
Lithography Unit	Temperature	8.789×10^{-26}
	Humidity	0.079

6.2. Performances of Static ML Models

Three well-known tree-based machine learning models are utilized to benchmark the performances of static ML models on the training and testing data. Specifically, we choose the Decision Tree, Random Forest and LightGBM models. Each model is configured with a maximum depth of 10 and a minimum samples split of 10, with a random state set to 42. The performances of these different static machine learning models were explored to identify the best-performing algorithm based on accuracy for both the training and testing datasets. These models are static, meaning that they do not adapt to changes in data distributions. To assess whether a model's performance deteriorates when subjected with new test data that have a different distribution from the training data, we calculated the cumulative accuracy over time. This approach helps in identifying how the accuracy of predictions evolves as the model encounters more data points.

The cumulative accuracy at any given point is determined using the ratio of the number of correct predictions up to that point to the total number of predictions made. Mathematically, cumulative accuracy over time for the combined (training + testing) dataset is defined as

$$\text{Cumulative Accuracy} = \frac{\sum_{i=1}^n I(y_i = \hat{y}_i)}{n}$$

where the variables are defined as follows:

- y_k is the true label for the k -th data point.
- $\hat{y}_k$ is the predicted label for the k -th data point.
- $I(y_i = \hat{y}_i)$ is the correctness indicator that outputs "True" if the prediction is correct (i.e., the predicted label matches the true label), and "False" otherwise.
- i is the current data point index, ranging from 1 to the total number of samples in both the training and testing datasets combined.

Figure 13 shows the performances of the static ML classification models on Fumehood dataset. As observed in Figure 13, the continuous learning model of TWIN-ADAPT consistently outperforms Random Forest and Decision Tree in both the training and testing phases, identifying it as the best static model. A notable drop in accuracy is observed for all classifiers when transitioning from the training to the testing data. This decline occurs because the samples in the training set have a similar statistical distribution, but the test set introduces a different statistical distribution. Consequently, the models trained on the training data cannot accurately detect anomalies or attacks in the test data. This phenomenon, known as concept drift, results in a significant decrease in accuracy when the test set begins. The performance is expected to degrade further if newer, more varied distributions are introduced into the testing data, demonstrating the limitations of static models in dynamic environment.

Figures 14 and 15 show the performances of the static models for anomaly classification on the Vacuum Pump and Lithography Unit datasets. In both cases, the optimized and continuous learning-driven LightGBM model of TWIN-ADAPT maintains an accuracy comparable to those of Random Forest and Decision Tree during the training phase. However, a significant decline in accuracy is observed for all classifiers at the start of the

testing phase, marked by the black dot. This drop indicates concept drift, as the test data introduce a different statistical distribution in anomalies, causing all models to struggle with maintaining their accuracy levels. Nonetheless, LightGBM still shows a marginally better performance across all three datasets during the testing phase, suggesting that it may generalize better than RF and DT in dynamic environments. Therefore, we select LightGBM for optimization, drift detection and adaptation for online anomaly detection in the TWIN-ADAPT.

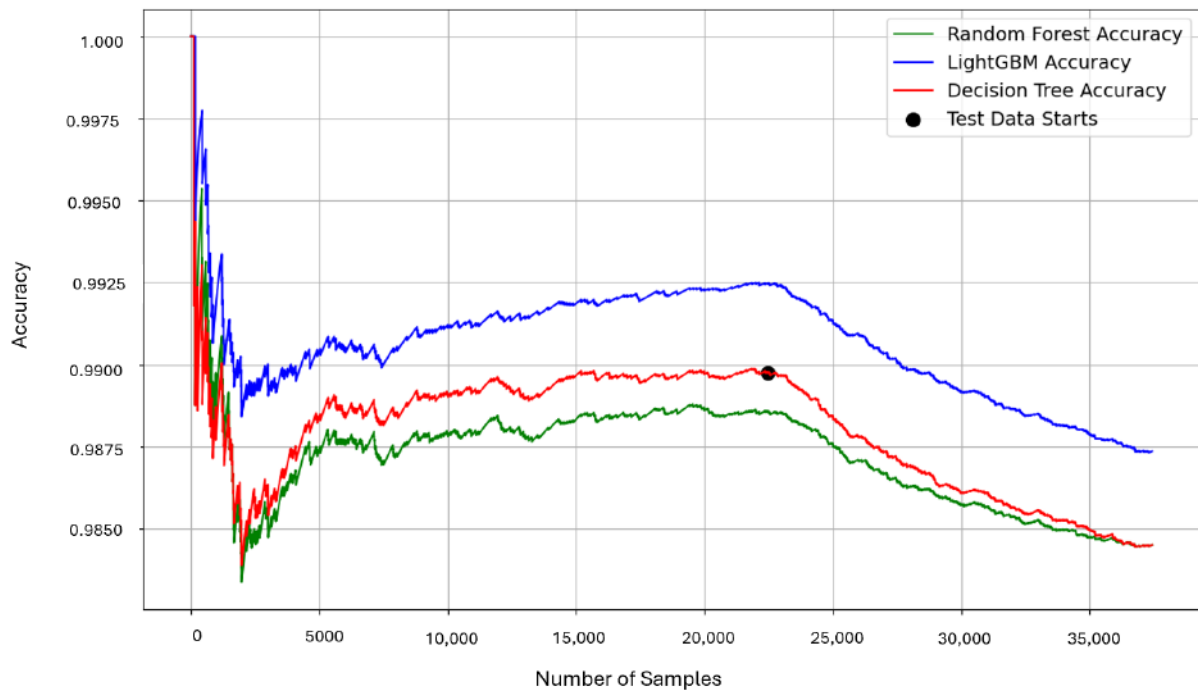


Figure 13. Static ML models' performances on training and testing datasets of Fumehood.

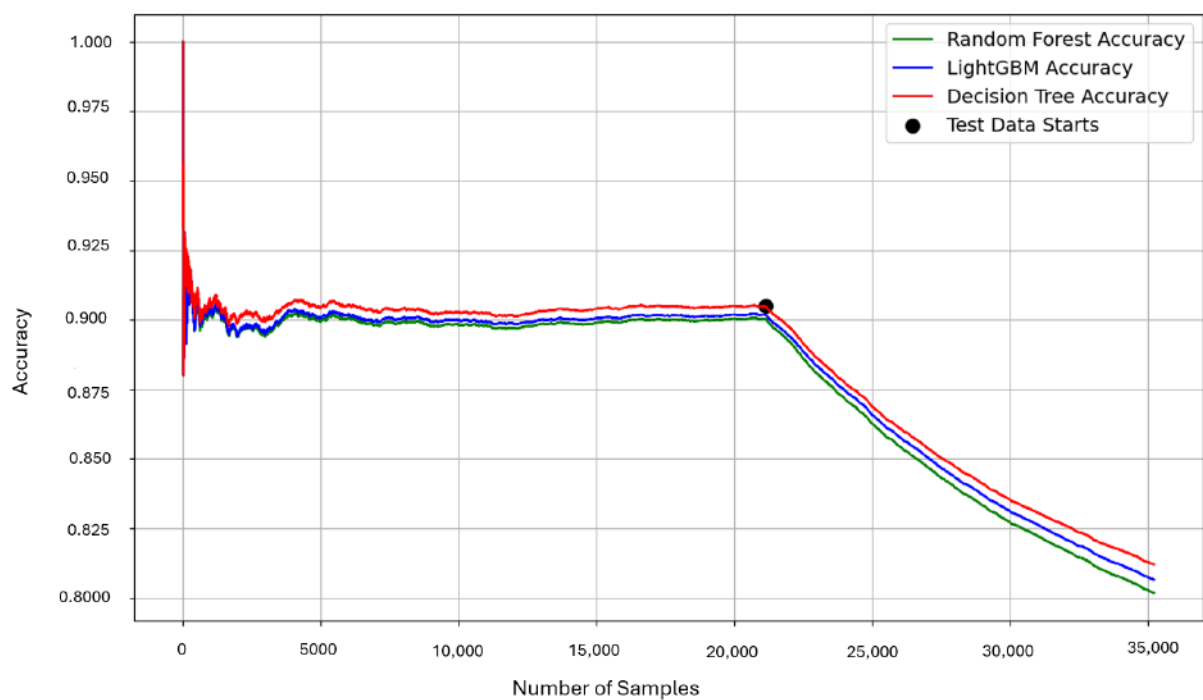


Figure 14. Static ML models' performances on training and testing datasets for Vacuum Pump.

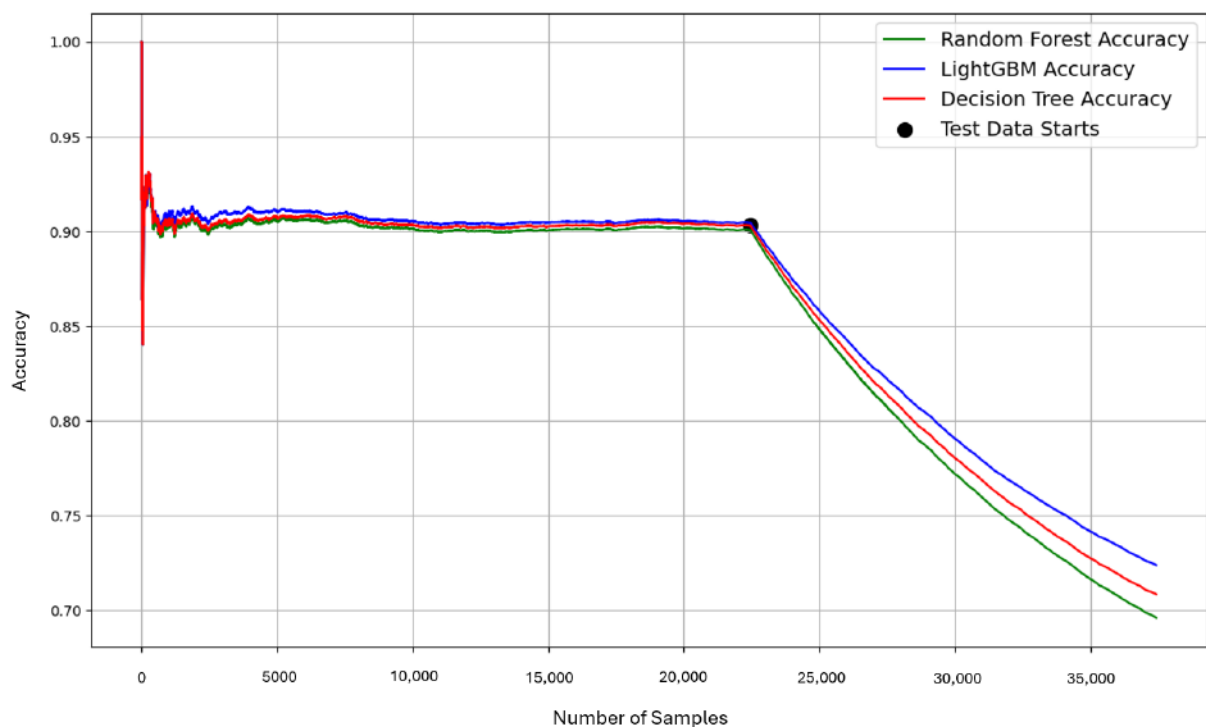


Figure 15. Static ML models' performances on training and testing datasets for Lithography Unit.

6.3. Performance of Baseline Drift Adaptation Models for Anomaly Classification

As discussed in Section 4.3, we analyze the effectiveness of our adaptive and optimized anomaly classification model of TWIN-ADAPT by comparing it with seven baseline models for drift adaptation. Four of these models—including ARF-ADWIN, ARF-DDM, SRP-ADWIN and SRP-DDM—are customized by combining drift detectors (ADWIN and DDM) with drift adaptors (ARF and SRP) to create robust models capable of handling concept drift. The other three individual models for online drift adaptation are EFDT, LB and HB. The performances of the various baseline drift adaptation models on the testing datasets, following their training on the respective training sets, are depicted in Figures 16–18. For the Fumehood dataset, as shown in Figure 16, the SRP-DDM model achieves the highest average accuracy of 97.01%, followed by the LB model at 96.45% and the SRP-ADWIN model at 95.53%. The ARF-ADWIN and ARF-DDM models also maintain high accuracy levels, whereas the EFDT and HT models perform significantly worse, with average accuracies of 86.11% and 91.62%, respectively. On the Vacuum Pump dataset (Figure 17), the SRP-DDM model again shows a superior performance with an average accuracy of 67.33%, closely followed by the LB model at 67.93%. The SRP-ADWIN and ARF-DDM models also maintain competitive accuracy levels. However, the EFDT and HT models lag behind, with average accuracies of 62.53% and 65.41%, respectively. For the Lithography dataset (Figure 18), the results indicate that the SRP-DDM model leads with an average accuracy of 64.82%, closely matched by the ARF-DDM and SRP-ADWIN models. The ARF-ADWIN model also performs relatively well, while the EFDT model shows a lower accuracy at 55.27%. The HT model achieves an average accuracy of 45.56%, indicating that it struggles the most with this dataset.

To complement these results, Table 4 also presents the F1 scores, precision, recall and accuracy for various models across the Fumehood, Vacuum Pump, and Lithography Machine datasets. In observing the F1 scores and precision values, the SRP-DDM model consistently achieves a high performance across all three datasets. For instance, the SRP-DDM model has an F1 score of 0.97 for the Fumehood dataset, 0.705 for the Vacuum Pump dataset and 0.678 for the Lithography Machine dataset, indicating its superior performance in comparison to the other baseline drift adaptation models. It is worth noting that across

all datasets, the models initially struggle to adapt but achieve stable performances after processing a few samples, demonstrating their ability to handle concept drifts effectively over time. This trend is evident in the initial fluctuations followed by stabilization in the accuracy metrics. In summary, while the SRP-DDM and LB models generally exhibit the highest performances across the datasets, the other baseline drift adaptation learners such as ARF-ADWIN, ARF-DDM and SRP-ADWIN also maintain competitive accuracies. Conversely, the EFDT and HT models consistently show low performances as they fail to capture the intricate relationships within the data due to the models' reliance on simpler, rule-based structures for adaptation.

Table 4. Comparison of performance measures for different drift adaptation models.

Title 1	Metrics	CPS Datasets		
		Fumehood	Vacuum Pump	Lithography Unit
ARF-ADWIN	Accuracy (%)	95.72	64.89	65.73
	F1 Score	0.957	0.646	0.649
	Precision	0.957	0.643	0.645
	Recall	0.957	0.649	0.6572
ARF-DDM	Accuracy (%)	95.47	66.52	67.01
	F1 Score	0.955	0.663	0.664
	Precision	0.955	0.662	0.661
	Recall	0.954	0.665	0.67
SRP-ADWIN	Accuracy (%)	95.98	70.21	67.30
	F1 Score	0.959	0.691	0.668
	Precision	0.958	0.690	0.666
	Recall	0.959	0.702	0.673
SRP-DDM	Accuracy (%)	97.34	70.53	67.84
	F1 Score	0.973	0.692	0.671
	Precision	0.974	0.693	0.668
	Recall	0.973	0.705	0.678
EFDT	Accuracy (%)	94.52	61.33	58.75
	F1 Score	0.948	0.544	0.569
	Precision	0.955	0.548	0.56
	Recall	0.945	0.613	0.588
HT	Accuracy (%)	94.83	66.36	52.75
	F1 Score	0.951	0.559	0.516
	Precision	0.96	0.672	0.674
	Recall	0.948	0.663	0.527
LB	Accuracy (%)	97.19	70.0	65.28
	F1 Score	0.972	0.638	0.629
	Precision	0.973	0.719	0.628
	Recall	0.972	0.70	0.653
TWIN-ADAPT	Accuracy (%)	96.97	71.92	69.26
	F1 Score	0.97	0.695	0.693
	Precision	0.973	0.710	0.695
	Recall	0.97	0.719	0.692

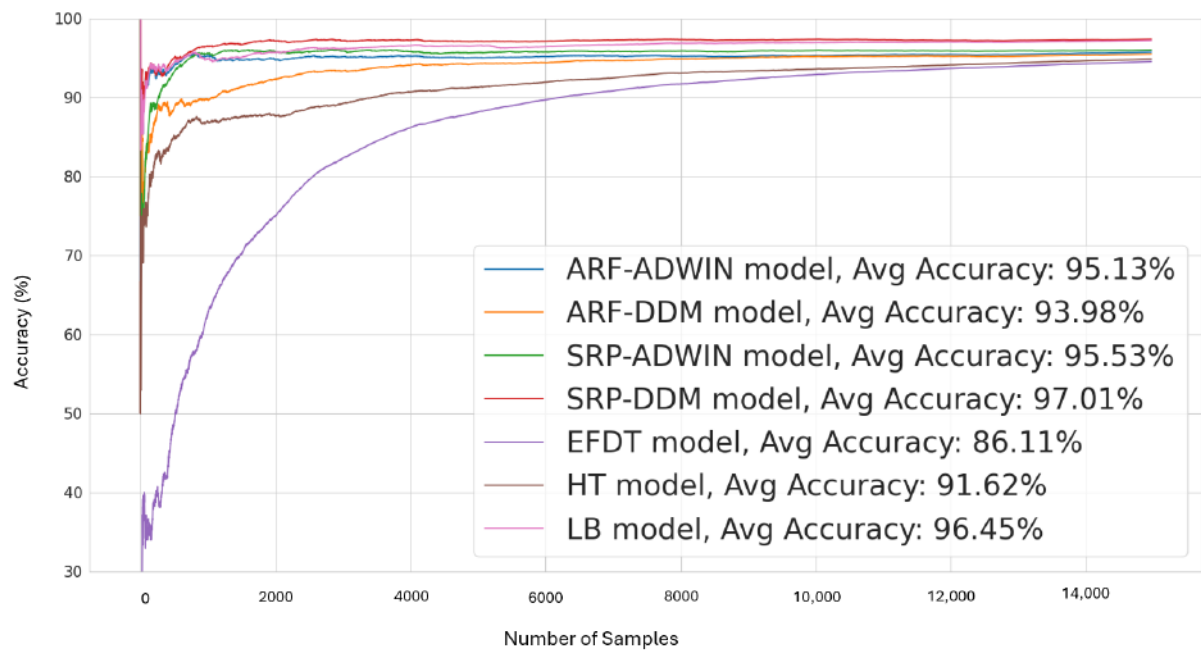


Figure 16. Average accuracy comparison for baseline drift adaptation models on Fumehood dataset.

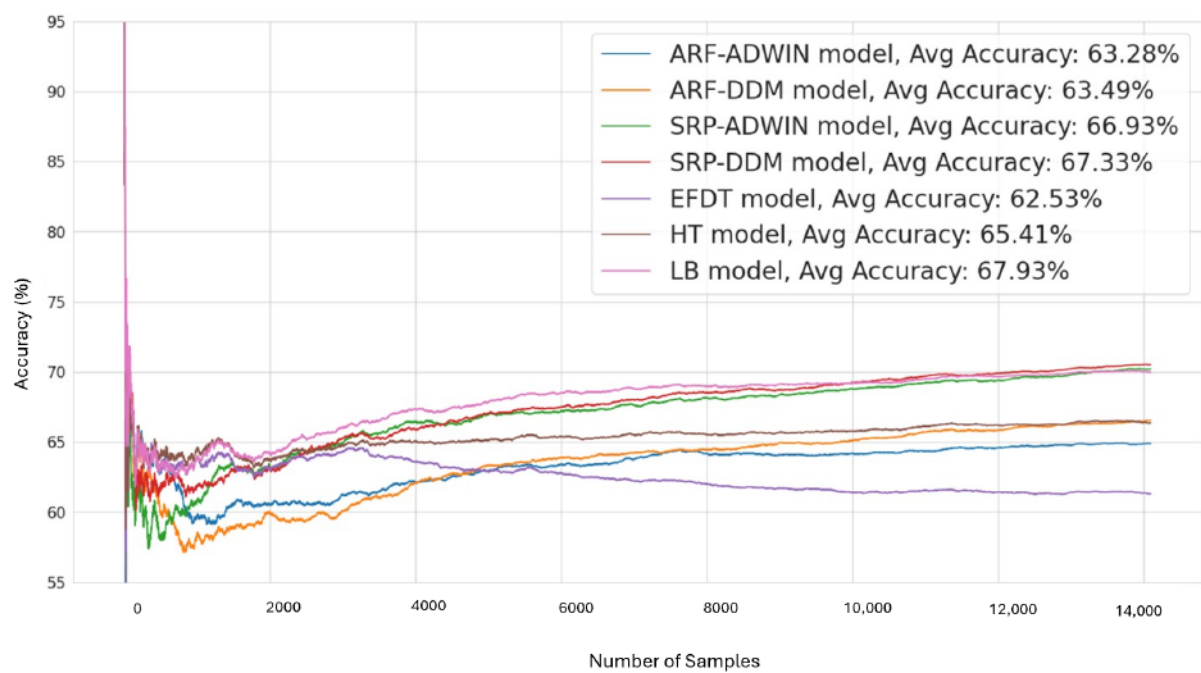


Figure 17. Average accuracy comparison for baseline drift adaptation models on Vacuum Pump dataset.

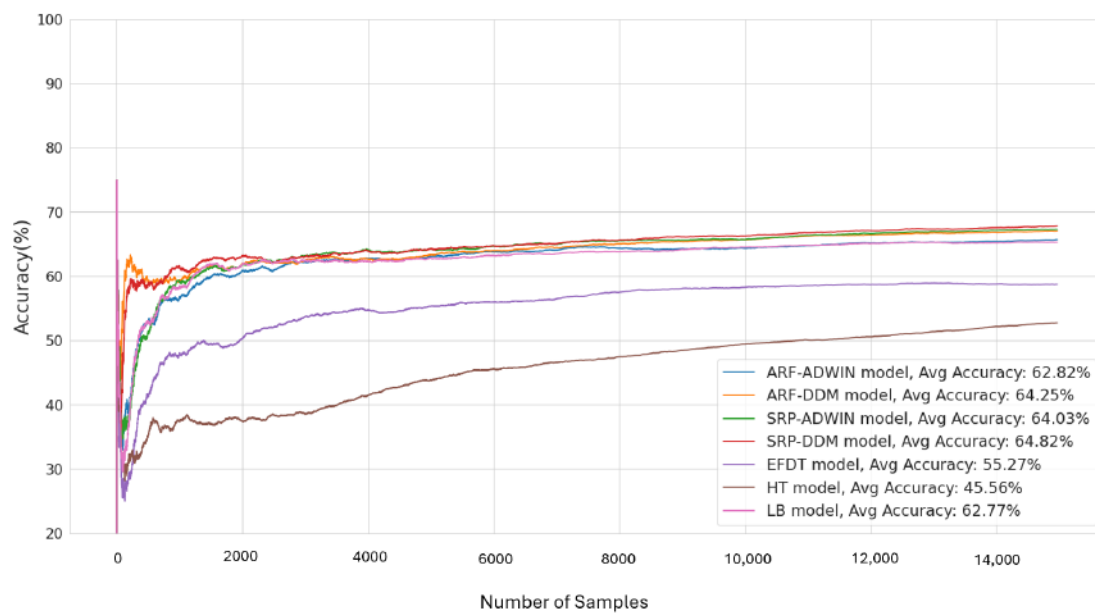


Figure 18. Average accuracy comparison for baseline drift adaptation models on Lithography Unit dataset.

6.4. Performance of TWIN-ADAPT

In order to obtain optimized TWIN-ADAPT models, their hyperparameters are automatically tuned using a Particle Swarm Optimization (PSO) technique as already discussed in Section 5.1 (Algorithm 2). The initial hyperparameter search range and the computed optimal hyperparameter values for TWIN-ADAPT's Algorithm 1 on the three experimental datasets are shown in Table 5. After using PSO, optimal hyperparameter values are assigned to the LightGBM classifier to construct an optimized model for adaptive anomaly classification under concept drift conditions.

The performance comparison of TWIN-ADAPT's optimized and online LightGBM model against the offline LightGBM model during the testing phase is shown in Figures 19–21. For the Fumehood dataset, as illustrated in Figure 19, the online adaptive model achieves a relatively higher average accuracy of 96.97% compared to the offline LightGBM model's accuracy of 96.41% (with an average performance improvement of 0.57%). For the Vacuum Pump dataset, as illustrated in Figure 20, the adaptive model again shows a significantly superior performance with an average accuracy of 71.92%, while the offline LightGBM model achieves only 65.13% (with an average performance improvement of 10.42%). For the Lithography dataset, Figure 21 reveals that the online adaptive LightGBM model outperforms the offline model with an average accuracy of 69.26% compared to 39.57% accuracy from the static LightGBM model (with an average performance improvement of 75.60%).

Table 5. TWIN-ADAPT's hyperparameter configuration.

Hyperparameters	Search Space	Fumehood	Vacuum Pump	Lithography Unit
γ	[0.90, 1]	0.977	0.984	0.9711
δ	[0.90, 1]	0.965	0.912	0.917
<i>winSize</i>	[100, 1000]	247	541	360
<i>winSizeMax</i>	[500, 5000]	2892	4111	2443

From these results, it is evident that the presented TWIN-ADAPT's optimized and continuous learning-driven LightGBM model consistently outperforms the offline LightGBM model across all datasets, exemplifying the effectiveness of the adaptive learning approach in dynamic industrial CPS environments. This is further supported by the performance

metrics in Table 4, which show that the TWIN-ADAPT model also excels in precision, recall and F1 score, indicating its robustness and reliability for online anomaly classification for CPS datasets. It is also worth mentioning that the performance improvement for the Fume-hood dataset is marginal because even the offline model performs well, indicating it has already converged. In contrast, the significant improvements observed in the Lithography and Vacuum Pump datasets demonstrate that adaptation in dynamic environments can significantly enhance performance under concept drift conditions. The results emphasize the necessity and advantage of incorporating adaptive learning mechanisms in CPS utilizing anomaly and fault classification for online data streams.

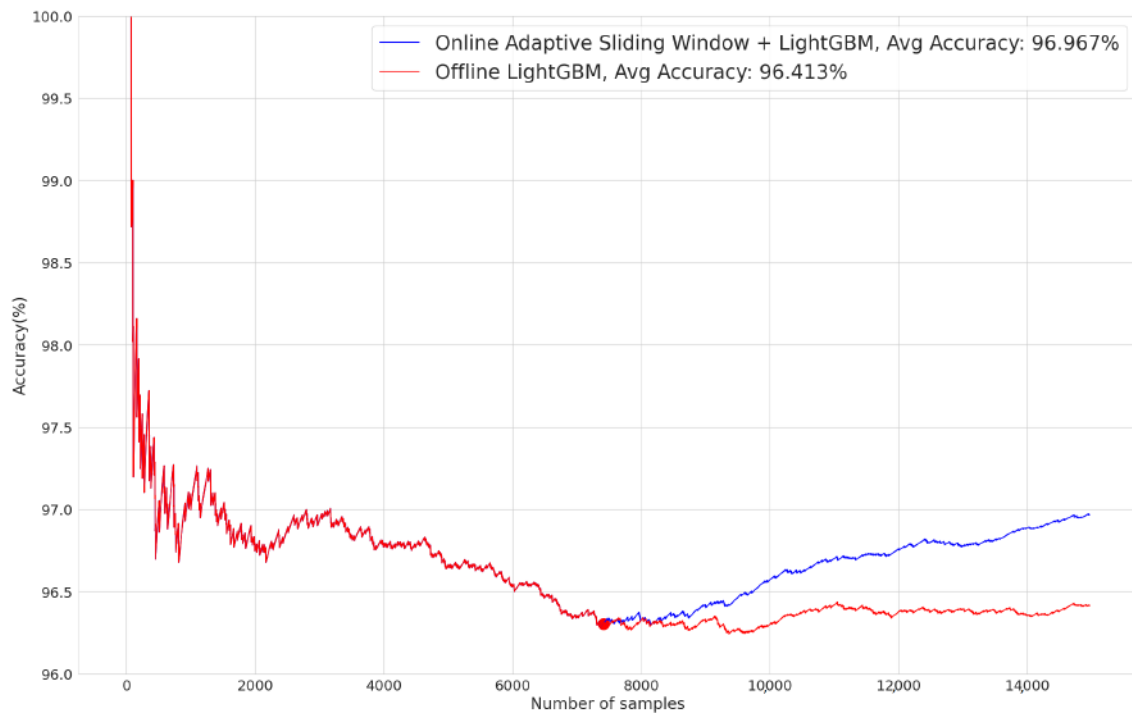


Figure 19. Average accuracy comparison of TWIN-ADAPT's online adaptive LightGBM model vs. offline LightGBM model for Fumehood test dataset.

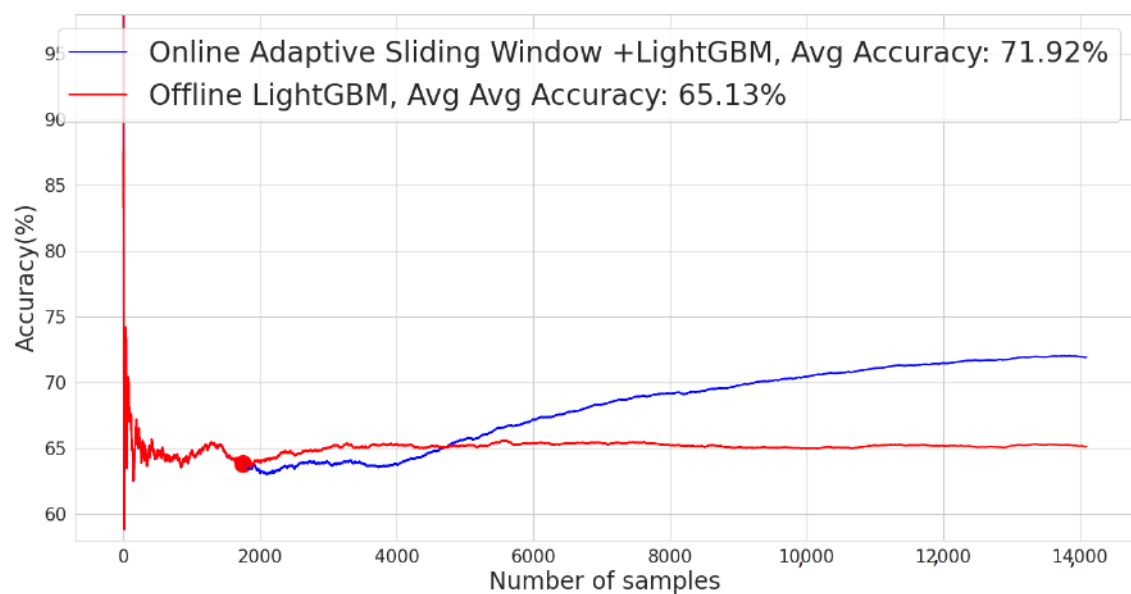


Figure 20. Average accuracy comparison of TWIN-ADAPT's online adaptive LightGBM model vs. offline LightGBM model for Vacuum Pump's test dataset.

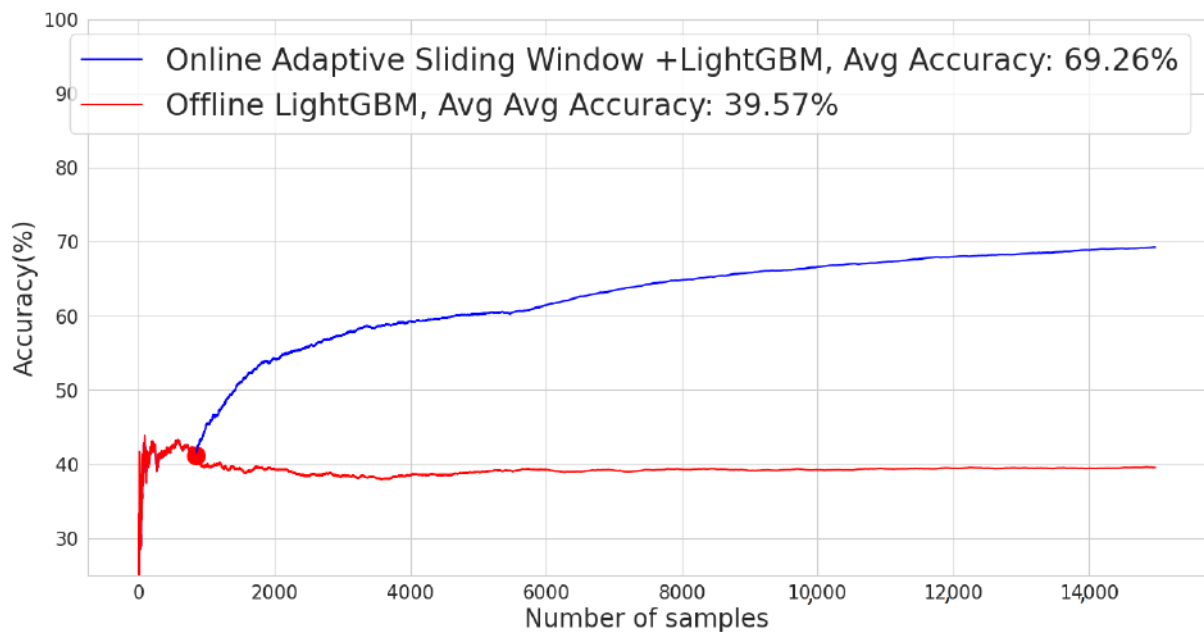


Figure 21. Average accuracy comparison of TWIN-ADAPT's online adaptive LightGBM model vs. offline LightGBM model for Lithography Unit's test dataset.

7. Discussion

- Performance of TWIN-ADAPT across different CPS datasets:** TWIN-ADAPT works well with the Fumehood dataset, which contains multiple features that offer sufficient information to train the model accurately (average accuracy of 96.97% and F1 score of 0.97). In contrast, the significant performance improvements observed on the Lithography and Vacuum Pump datasets demonstrate that TWIN-ADAPT significantly enhances the performance under concept drift conditions compared to the older offline model. These results could be further improved by exploring other advanced deep learning-based classification models within the TWIN-ADAPT framework, such as CNNs, MLPs or Transformers to enhance the performance on datasets with limited features for learning or high-anomaly distributions.
- Comparative analysis between TWIN-ADAPT and baseline drift adaptation models:** As observed in Table 4, TWIN-ADAPT generally outperforms traditional drift adaptation methods due to its more flexible adaptation strategy. While methods like ADWIN rely on fixed-size sliding windows to manage data and detect drift and DDM monitors degradation in model performance—which are not dynamically adaptive to new or evolving data patterns—TWIN-ADAPT's approach allows for more fine-grained adjustments to both sudden and gradual concept changes, integrating drift detection and adaptation into a singular process. This eliminates the need to pair separate drift detectors with adaptation models, which typically increases computational and time overheads. TWIN-ADAPT seamlessly combines sliding and adaptive windows, enhancing both the detection and adaptation to drift within a streamlined framework. Additionally, instead of comparing the statistical properties of the data for drift detection and adaptation, TWIN-ADAPT focuses on performance-based adaptation. This ensures that model updates for retraining occur only when the performance degrades significantly, rather than responding to every minor data fluctuation that might not impact the model's effectiveness.
- Computational complexity of TWIN-ADAPT:** The computational complexity of TWIN-ADAPT's continual learning model is determined by the operations within its sliding and adaptive windows, and the hyperparameter optimization using Particle Swarm Optimization. Specifically, the sliding window operation has a complexity of $O(NM)$, where N is the number of data samples in the every batch, and M represents

the number of times that the hyperparameters are tuned within PSO. PSO itself is relatively efficient, with a complexity of $O(N \log N)$, supporting parallel execution to enhance performance. This setup ensures that the TWIN-ADAPT model remains computationally feasible for real-time applications in IoT-driven cyber-physical systems, balancing the need for accurate drift adaptation with the real-time constraints of computational resources.

- **Cross-domain generalization:** The integration of automated hyperparameter optimization using Particle Swarm Optimization (PSO) allows TWIN-ADAPT to be effectively generalized across various domain datasets. This feature enables adaptive anomaly classification without the need for manual tuning or intervention, which reduces the risk of bias and enhances model applicability in diverse settings.
- **Impact of data quality check to prevent overfitting:** The integrity of TWIN-ADAPT's performance is heavily reliant on high data quality. Spurious drifts and noisy data can significantly distort the model performance due to frequent retraining and resets. Implementing stringent data quality checks ensures that only relevant and precise data influence the model's training process, thereby avoiding overfitting and underfitting.

8. Future Directions

The following future enhancements to the TWIN-ADAPT system can focus on several key areas to improve its robustness and adaptability:

- **Incorporating old concepts for retraining:** A potential enhancement could involve retaining the historical concepts within the adaptation model, rather than discarding them from the adaptive window. This approach would enable the model to quickly revert to previous configurations (of model hyperparameters) without the need for retraining in cases where recurrent drifts occur, causing the old anomalous patterns to re-emerge. Such a strategy would reduce the computational overhead of duplicated retraining, improving the responsiveness in dynamic environments.
- **Alternate model learners for different drift scenarios:** Incorporating a model zoo, or an alternating learner framework [10], represents a significant potential enhancement for TWIN-ADAPT. By maintaining a collection of models, each tailored to different types of previously encountered concept drifts, the system can efficiently manage recurrent patterns without needing to retrain from scratch each time an old pattern resurfaces. This approach not only saves computational resources but also reduces latency in response to drifts, ensuring that the digital twin can quickly switch between models to match the current data scenario.
- **Testing in varied drift scenarios and noise distributions:** Future developments could involve the extensive testing of TWIN-ADAPT across a variety of concept drift types—such as gradual, sudden, recurrent and incremental drifts—to enhance drift detection and adaptation methodologies. This testing would help develop domain-specific adaptation policies, potentially avoiding unnecessary adaptations in cases of spurious drifts associated with specific tasks or processes. Additionally, exploring the impact of different noise distributions like Power Law and Laplacian, beyond the commonly used Gaussian and Uniform models, could provide deeper insights into the robustness of TWIN-ADAPT against various types of data disturbances.
- **Adaptive feature selection:** An adaptive feature selection mechanism can be developed to dynamically adjust the features used for model training based on their correlations with the target variables, ensuring that the model remains effective despite fluctuations in data streams.
- **Generative AI and LLMs for drift adaptation within digital twins:** Generative models such as GANs and VAEs can create synthetic data samples that mimic possible future normal operational states, anomalies (or faults) of the monitored CPS not yet seen in the real data. This capability is useful for training the digital twin model to handle potential future scenarios, effectively preparing the system for “unseen” drifts and anomalies. Moreover, LLMs can be used to generate explanatory narratives or

decisions based on the data trends observed by the digital twin [56]. For instance, if a drift is detected, the model can suggest potential causes (such as drift localization or features causing the drift) and recommend corrective actions based on similar past scenarios it has learned during training.

- **Collaborative learning in digital twins:** Collaborative learning in digital twins enables CPS components to collectively analyze and respond to concept drift and anomalies, ensuring robust decision making. In sharing insights, even if one CPS component shows drift, others can validate its reliability, reducing false positives and enhancing overall system accuracy. This approach fosters a resilient, interconnected CPS in industrial environments.
- **Inducing different types of drifts using anomaly injection:** Currently, our noise injection method introduces random points of anomalies to induce drifts of different distributions in the testing data, which closely resemble abrupt, sudden and incremental drifts. However, systematic studies can be conducted to strategically simulate other drift types (like re-occurring drifts) more explicitly to observe the model's robustness.
- **Contrastive learning:** Contrastive learning is an innovative machine learning approach that helps the model adapt to concept drift using pairs of similar and dissimilar data points to continuously update the model, ensuring that it recognizes relevant changes in the data over time. As a future prospect, it can also be used for digital twins to enhance their adaptability and accuracy in reflecting data distribution changes in real-world systems.
- **Advanced AI models:** TWIN-ADAPT can be further extended using deep neural network methods such as CNNs, Multi-Layer Perceptrons (MLPs) and Autoencoders for anomaly detection and classification. These models can not only handle large volumes of data but also better generalize across different types of data, thus improving accuracy in digital twin frameworks. Additionally, Physics-Informed Neural Networks (PINNs) [57] can be used to incorporate domain-specific physical properties into neural network architecture, enhancing model robustness in dynamic environments. However, these models require large training datasets and continual learning to stay updated, which often leads to the problem of catastrophic forgetting [58]. Thus, continuous learning approaches are essential to ensure models retain past knowledge while adapting to new data.

9. Conclusions

In dynamic environments, such as scientific laboratories integrated with IoT and CPSs, operational changes, sensor aging, machine degradation, software updates or the addition of new processes and equipment can lead to varied data distributions over time causing model degradation decay for anomaly (or fault) detection and classification. These changes necessitate the use of adaptive models capable of detecting anomalies even as data distributions evolve, ensuring a high model performance. Continuous anomaly classification monitoring in the rapidly changing conditions of scientific laboratories requires models that can dynamically learn and adapt to evolving data streams. In this context, we develop a continual learning algorithm for digital twin-enabled online anomaly classification in TWIN-ADAPT to adjust to concept drift while maintaining an accurate anomaly classification. Our experimental results with the TWIN-ADAPT model highlight the effectiveness and robustness of its adaptive and optimized classification approach in dynamic CPS settings. TWIN-ADAPT employs a dual-window strategy for drift adaptation, integrated with Particle Swarm Optimization, to continually adjust to evolving data streams. Our method is evaluated under simulated drift scenarios using different noise distributions to emulate anomalous drifts in the training and testing data distributions. The TWIN-ADAPT framework significantly outperforms its offline counterparts and demonstrates a competitive performance against benchmark drift adaptation algorithms for three industrial CPS components. This continual learning strategy of TWIN-ADAPT ensures a better accuracy

and reliability of machine learning models used for digital twins to better suit the changing needs of industrial settings.

Author Contributions: Conceptualization, R.G. and K.N.; methodology, R.G. and K.N.; software, R.G.; validation, R.G., B.T. and Y.W.; resources, B.T., Y.W. and K.N.; data curation, R.G., B.T. and Y.W.; writing—original draft preparation, R.G. and K.N.; writing—review and editing, R.G., B.T. and K.N.; supervision, K.N.; project administration, K.N.; funding acquisition, K.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Science Foundation, NSF OAC 21-26246, “CC* Integration-Large” MAINTLET: Advanced Sensory Network Cyber-Infrastructure for Smart Maintenance in Campus Scientific Laboratories.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy restrictions.

Acknowledgments: Any results and opinions are authors own and do not represent views of the National Science Foundation.

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

References

1. Yao, Y.; Qian, Q. Dynamic Industrial Optimization: A Framework Integrates Online Machine Learning for Processing Parameters Design. *Future Internet* **2024**, *16*, 94. [\[CrossRef\]](#)
2. Chhetri, S.R.; Faezi, S.; Canedo, A.; Faruque, M.A.A. QUILT: Quality inference from living digital twins in IoT-enabled manufacturing systems. In Proceedings of the International Conference on Internet of Things Design and Implementation: IoTDI’19, Montreal, QC, Canada, 15–18 April 2019; pp. 237–248. [\[CrossRef\]](#)
3. Zaccaria, V.; Stenfelt, M.; Aslanidou, I.; Kyprianidis, K.G. Fleet Monitoring and Diagnostics Framework Based on Digital Twin of Aero-Engines, Volume 6: Ceramics; Controls, Diagnostics, and Instrumentation; Education; Manufacturing Materials and Metallurgy. In Proceedings of the Turbo Expo: Power for Land, Sea, and Air, Oslo, Norway, 11–15 June 2018.
4. Losi, E.; Venturini, M.; Manservigi, L.; Ceschini, G.F.; Bechini, G. Anomaly Detection in Gas Turbine Time Series by Means of Bayesian Hierarchical Models. *J. Eng. Gas Turbines Power* **2019**, *141*, 111019. [\[CrossRef\]](#)
5. Gao, C.; Park, H.; Easwaran, A. An anomaly detection framework for digital twin driven cyber-physical systems. In Proceedings of the ACM/IEEE 12th International Conference on Cyber-Physical Systems: ICCPS ’21, Nashville, TN, USA, 19–21 May 2021; pp. 44–54. [\[CrossRef\]](#)
6. Das, T.; Wang, Z.; Sun, J. TWIN: Personalized Clinical Trial Digital Twin Generation. In Proceedings of the KDD 2023—29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Long Beach, CA, USA, 6–10 August 2023; pp. 402–413. [\[CrossRef\]](#)
7. Kalyani, Y.; Vorster, L.; Whetton, R.; Collier, R. Application Scenarios of Digital Twins for Smart Crop Farming through Cloud–Fog–Edge Infrastructure. *Future Internet* **2024**, *16*, 100. [\[CrossRef\]](#)
8. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**, *8*, 108952–108971. [\[CrossRef\]](#)
9. Tao, F.; Xiao, B.; Qi, Q.; Cheng, J.; Ji, P. Digital twin modeling. *J. Manuf. Syst.* **2022**, *64*, 372–389. [\[CrossRef\]](#)
10. Xu, Y.; Xu, R.; Yan, W.; Ardis, P. Concept drift learning with alternating learners. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 2104–2111. [\[CrossRef\]](#)
11. Al-Ali, A.R.; Gupta, R.; Batool, T.; Landolsi, T.; Aloul, F.; Al Nabulsi, A. Digital Twin Conceptual Model within the Context of Internet of Things. *Future Internet* **2020**, *12*, 163. [\[CrossRef\]](#)
12. Hines, D.R.; Siwak, N.P.; Mosher, L.A.; Ghodssi, R. MEMS lithography and micromachining techniques. In *MEMS Materials and Processes Handbook*; Springer: Boston, MA, USA, 2011; pp. 667–753.
13. Tian, B.; Yang, Z.; Moeini, H.; Gupta, R.; Su, P.; Kaufman, R.; McCollum, M.; Dallesasse, J.; Nahrstedt, K. Senselet++: A low-cost internet of things sensing platform for academic cleanrooms. In Proceedings of the 2021 IEEE 18th International Conference on Mobile Ad Hoc and Smart Systems (MASS), Denver, CO, USA, 4–7 October 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 90–98.
14. Yang, L.; Shami, A. A Lightweight Concept Drift Detection and Adaptation Framework for IoT Data Streams. *IEEE Internet Things Mag.* **2021**, *4*, 96–101. [\[CrossRef\]](#)
15. Cimino, C.; Negri, E.; Fumagalli, L. Review of digital twin applications in manufacturing. *Comput. Ind.* **2019**, *113*, 103130. [\[CrossRef\]](#)
16. Singh, M.; Srivastava, R.; Fuenmayor, E.; Kuts, V.; Qiao, Y.; Murray, N.; Devine, D. Applications of Digital Twin across Industries: A Review. *Appl. Sci.* **2022**, *12*, 5727. [\[CrossRef\]](#)
17. Folgado, F.J.; Calderón, D.; González, I.; Calderón, A.J. Review of Industry 4.0 from the Perspective of Automation and Supervision Systems: Definitions, Architectures and Recent Trends. *Electronics* **2024**, *13*, 782. [\[CrossRef\]](#)

18. Zayed, S.; Attiya, G.; El-Sayed, A.; Mahmoud, A.; Hemdan, E.E.D. An Efficient Fault Diagnosis Framework for Digital Twins Using Optimized Machine Learning Models in Smart Industrial Control Systems. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 69. [CrossRef]
19. Transcontinental Advanced Coatings. What Is a Cleanroom Laboratory. Available online: <https://www.transcontinentaladvancedcoatings.com/what-is-a-cleanroom-laboratory/> (accessed on 30 May 2024).
20. Ding, X.; Tian, Y.; Yu, Y. A Real-Time Big Data Gathering Algorithm Based on Indoor Wireless Sensor Networks for Risk Analysis of Industrial Operations. *IEEE Trans. Ind. Inform.* **2016**, *12*, 1232–1242. [CrossRef]
21. Jahan, I.; Alam, M.M.; Ahmed, M.F.; Jang, Y.M. Anomaly Detection in Semiconductor Cleanroom Using Isolation Forest. In Proceedings of the 2021 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Republic of Korea, 20–22 October 2021; pp. 795–797. [CrossRef]
22. Balta, E.C.; Tilbury, D.M.; Barton, K. A Digital Twin Framework for Performance Monitoring and Anomaly Detection in Fused Deposition Modeling. In Proceedings of the 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), Vancouver, BC, Canada, 22–26 August 2019; pp. 823–829. [CrossRef]
23. Castellani, A.; Schmitt, S.; Squartini, S. Real-World Anomaly Detection by Using Digital Twin Systems and Weakly Supervised Learning. *IEEE Trans. Ind. Inform.* **2021**, *17*, 4733–4742. [CrossRef]
24. Wang, Y.; Cao, Y.; Wang, F.Y. Anomaly Detection in Digital Twin Model. In Proceedings of the 2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI), Beijing, China, 15 July–15 August 2021; pp. 208–211. [CrossRef]
25. Álvaro G.; Bregon, A.; Martínez-Prieto, M.A. Digital Twin Learning Ecosystem: A cyber-physical framework to integrate human-machine knowledge in traditional manufacturing. *Internet Things* **2024**, *25*, 101094. [CrossRef]
26. Xu, Q.; Ali, S.; Yue, T. Digital Twin-based Anomaly Detection with Curriculum Learning in Cyber-physical Systems. *ACM Trans. Softw. Eng. Methodol.* **2023**, *32*, 1–32. [CrossRef]
27. Lu, Q.; Xie, X.; Parlikad, A.K.; Schooling, J. Digital Twin-Enabled Anomaly Detection for Built Asset Monitoring in Operation and Maintenance. *Autom. Constr.* **2020**, *118*, 103277. [CrossRef]
28. Huang, H.; Yang, L.; Wang, Y.; Xu, X.; Lu, Y. Digital Twin-driven online anomaly detection for an automation system based on edge intelligence. *J. Manuf. Syst.* **2021**, *59*, 138–150. [CrossRef]
29. Xu, Y.; Sun, Y.; Liu, X.; Zheng, Y. A Digital-Twin-Assisted Fault Diagnosis Using Deep Transfer Learning. *IEEE Access* **2019**, *7*, 19990–19999. [CrossRef]
30. Booyse, W.; Wilke, D.N.; Heyns, S. Deep digital twins for detection, diagnostics and prognostics. *Mech. Syst. Signal Process.* **2020**, *140*, 106612. [CrossRef]
31. Liu, X.; Brand, M.V.D.; Muctadir, H.; van de Molengraft, R. ML-based Digital Twin for anomaly detection: A case-study on Turtle soccer robots. In Proceedings of the 2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Durres, Albania, 6–8 September 2023; pp. 179–182. [CrossRef]
32. Sachdeva, U.; Vamsi, P.R. A Study on Anomaly Detection with Deep Learning Models for IoT Time Series Sensor Data. In Proceedings of the 2022 8th International Conference on Signal Processing and Communication (ICSC), Noida, India, 1–3 December 2022; pp. 11–14. [CrossRef]
33. Sonata, I.; Heryadi, Y. Transformer and CNN Comparison for Time Series Classification Model. In Proceedings of the 2023 15th International Congress on Advanced Applied Informatics Winter (IIAI-AAI-Winter), Bali, Indonesia, 11–13 December 2023; pp. 160–164. [CrossRef]
34. Bifet, A.; Gavaldà, R. Learning from Time-Changing Data with Adaptive Windowing. In Proceedings of the 2007 SIAM International Conference on Data Mining (SDM), Minneapolis, MN, USA, 26–28 April 2007; pp. 443–448. [CrossRef]
35. Bayram, F.; Ahmed, B.S.; Kassler, A. From concept drift to model degradation: An overview on performance-aware drift detectors. *Knowl. Based Syst.* **2022**, *245*, 108632. [CrossRef]
36. Palli, A.S.; Jaafar, J.; Gomes, H.M.; Hashmani, M.A.; Gilal, A.R. An Experimental Analysis of Drift Detection Methods on Multi-Class Imbalanced Data Streams. *Appl. Sci.* **2022**, *12*, 11688. [CrossRef]
37. Yang, L.; Manias, D.M.; Shami, A. PWPAAE: An Ensemble Framework for Concept Drift Adaptation in IoT Data Streams. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; pp. 1–6. [CrossRef]
38. Sakurai, G.Y.; Lopes, J.F.; Zarpelão, B.B.; Barbon Junior, S. Benchmarking Change Detector Algorithms from Different Concept Drift Perspectives. *Future Internet* **2023**, *15*, 169. [CrossRef]
39. Sun, Y.; Wang, Z.; Liu, H.; Du, C.; Yuan, J. Online Ensemble Using Adaptive Windowing for Data Streams with Concept Drift. *Int. J. Distrib. Sens. Netw.* **2016**, *12*, 4218973. [CrossRef]
40. Wares, S.; Isaacs, J.P.; Elyan, E. Data stream mining: Methods and challenges for handling concept drift. *SN Appl. Sci.* **2019**, *1*, 1412. [CrossRef]
41. Manapragada, C.; Webb, G.I.; Salehi, M. Extremely Fast Decision Tree. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining: KDD '18, London, UK, 19–23 August 2018; pp. 1953–1962. [CrossRef]
42. Gomes, H.M.; Bifet, A.; Read, J.; Barddal, J.P.; Enembreck, F.; Pfahringer, B.; Holmes, G.; Abdesslem, T. Adaptive random forests for evolving data stream classification. *Mach. Learn.* **2017**, *106*, 1469–1495. [CrossRef]
43. Gomes, H.M.; Read, J.; Bifet, A. Streaming Random Patches for Evolving Data Stream Classification. In Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), Beijing, China, 8–11 November 2019; pp. 240–249. [CrossRef]

44. Lv, Y.; Peng, S.; Yuan, Y.; Wang, C.; Yin, P.; Liu, J.; Wang, C. A classifier using online bagging ensemble method for big data stream learning. *Tsinghua Sci. Technol.* **2019**, *24*, 379–388. [\[CrossRef\]](#)
45. Chabanet, S.; El-Haouzi, H.B.; Thomas, P. Toward a self-adaptive digital twin based Active learning method: An application to the lumber industry. *IFAC-PapersOnLine* **2022**, *55*, 378–383. [\[CrossRef\]](#)
46. Hashash, O.; Chaccour, C.; Saad, W. Edge Continual Learning for Dynamic Digital Twins over Wireless Networks. *arXiv* **2022**, arXiv:2204.04795.
47. Lombardo, G.; Picone, M.; Mamei, M.; Mordonini, M.; Poggi, A. Digital Twin for Continual Learning in Location Based Services. *Eng. Appl. Artif. Intell.* **2024**, *127*, 107203. [\[CrossRef\]](#)
48. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A highly efficient gradient boosting decision tree. In Proceedings of the 31st International Conference on Neural Information Processing Systems: NIPS'17, Long Beach, CA, USA, 4–9 December 2017; pp. 3149–3157.
49. Liu, J.; Yang, D.; Lian, M.; Li, M. Research on Intrusion Detection Based on Particle Swarm Optimization in IoT. *IEEE Access* **2021**, *9*, 38254–38268. [\[CrossRef\]](#)
50. Yang, L.; Shami, A. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing* **2020**, *415*, 295–316. [\[CrossRef\]](#)
51. Massey, F.J., Jr. The Kolmogorov-Smirnov test for goodness of fit. *J. Am. Stat. Assoc.* **1951**, *46*, 68–78. [\[CrossRef\]](#)
52. Shang, D.; Zhang, G.; Lu, J. Fast concept drift detection using singular vector decomposition. In Proceedings of the 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Nanjing, China, 24–26 November 2017; pp. 1–6. [\[CrossRef\]](#)
53. Luo, J.; Wang, S.; Zhang, E.; Luo, J. Non-cooperative signal detection in alpha stable noise via Kolmogorov-Smirnov test. In Proceedings of the 2015 8th International Congress on Image and Signal Processing (CISP), Shenyang, China, 14–16 October 2015; pp. 1464–1468. [\[CrossRef\]](#)
54. Okamura, H.; Dohi, T. On Kolmogorov-Smirnov Test for Software Reliability Models with Grouped Data. In Proceedings of the 2019 IEEE 19th International Conference on Software Quality, Reliability and Security (QRS), Sofia, Bulgaria, 22–26 July 2019; pp. 77–82. [\[CrossRef\]](#)
55. Iqbal, J. Noise Pattern Recognition in Oceanic Environment Using Statistical Characterization of Oceanic Noise in Deep Sea: A Computational Design Approach. In Proceedings of the 2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 14–18 January 2020; pp. 596–600. [\[CrossRef\]](#)
56. Sun, Y.; Zhang, Q.; Bao, J.; Lu, Y.; Liu, S. Empowering digital twins with large language models for global temporal feature learning. *J. Manuf. Syst.* **2024**, *74*, 83–99. [\[CrossRef\]](#)
57. Zideh, M.; Chatterjee, P.; Srivastava, A. Physics-Informed Machine Learning for Data Anomaly Detection, Classification, Localization, and Mitigation: A Review, Challenges, and Path Forward. *IEEE Access* **2023**, *12*, 4597–4617. [\[CrossRef\]](#)
58. Dash, S.; Yin, J.; Shankar, M.; Wang, F.; Feng, W.C. Mitigating Catastrophic Forgetting in Deep Learning in a Streaming Setting Using Historical Summary. In Proceedings of the 2021 7th International Workshop on Data Analysis and Reduction for Big Scientific Data (DRBSD-7), St. Louis, MO, USA, 14 November 2021; pp. 11–18. [\[CrossRef\]](#)

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