Taming Data Quality in AI-Enabled Industrial Internet of Things

Sagar Sen, Erik Johannes Husom, Arda Goknil, Simeon Tverdal, Phu Nguyen
SINTEF, Oslo, Norway

Iker Mancisidor
IDEKO, Elgoibar, Spain

***Abstract*—Artificial intelligence (AI)-enabled Industrial Internet of Things (IIoT) marks the rise of systems at the convergence of tremendous amounts of data from multiple IoT devices for complex machine learning/AI software that supports decision making and predictive maintenance in various industries. However, the omnipresent neglect of data quality leads to the accumulation of dark data and the impregnation of biases in AI systems. We address the problem of taming data quality in AI-enabled IIoT systems by devising *machine learning pipelines* as part of a decentralized edge-to-cloud architecture. These pipelines generate services for (i) erroneous data repair and (ii) unsupervised detection of events and deviations in sensor data. We present the design and deployment of our approach from an AI Engineering perspective using two industrial case studies.**

■ **THE INDUSTRIAL INTERNET OF THINGS (IIOT)** revolutionizes several industries, such as manufacturing, transportation, and energy. It is a major driving force behind Industry 4.0 and employs Artificial Intelligence (AI) techniques, e.g., Machine Learning (ML), to exploit the massive interconnection and large volumes of IIoT data. AI-enabled Industrial IoT systems (IIoTs) improve decision-making [1] and perform predictive maintenance [2] (e.g., tool wear and product defect prediction in the manufacturing domain) in industrial processes. The quality and continuity of IIoT data is a bottleneck and makes these systems rather conservative in what they can achieve. Furthermore, the growing neglect of data quality in AI-enabled IIoTs [3] leads to the accumulation of dark data (unstructured, untagged, and untapped data not analyzed) [4] and the impregnation of biases [5].

IIoT data endures a long journey on the edge-cloud continuum: (i) data obtained by sensors observing industrial processes is consumed by a rugged industrial computer to control actuators, such as a machine tool in manufacturing; (ii) it is transferred to an edge device over wired/wireless communication channels using industrial communication protocols (e.g., OPC-UA, OPC-DA, NMEA, Bluetooth); and (iii) it is aggregated on edge to be transferred to the cloud using API protocols (e.g., REST, RPC, SOAP, GraphQL). Taming data quality in AI-enabled IIoTs aims to detect and manage data quality issues (bias, freezing, precision degradation, data drift in sensors) on this journey and preserve data continuity on the edge-cloud continuum. Sensor bias is an offset shifting sensor output by a constant value. A sensor freezes when its output is constant in successive samples. Precision degradation occurs

This is the author accepted version of an article published in
IEEE Software volume: 39, issue: 6, Nov.-Dec. 2022
https://doi.org/10.1109/MS.2022.3193975

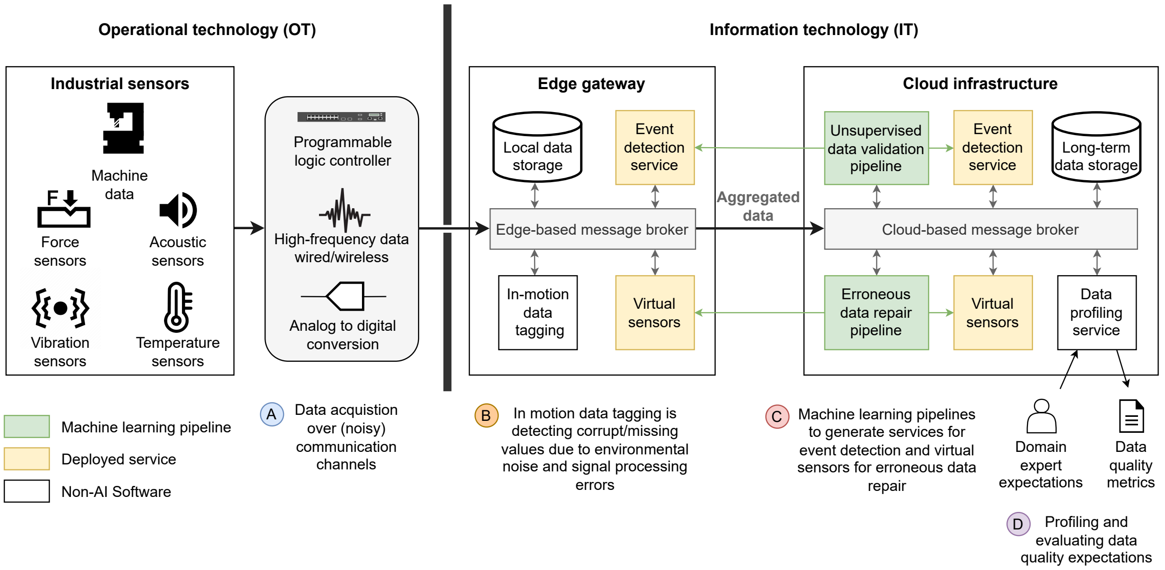

Operational technology (OT)
Information technology (IT)
Industrial sensors
Machine data
Force sensors
Acoustic sensors
Vibration sensors
Temperature sensors
Programmable logic controller
High-frequency data wired/wireless
Analog to digital conversion
Edge gateway
Local data storage
Event detection service
Edge-based message broker
In-motion data tagging
Virtual sensors
Aggregated data
Cloud infrastructure
Unsupervised data validation pipeline
Event detection service
Long-term data storage
Cloud-based message broker
Erroneous data repair pipeline
Virtual sensors
Data profiling service
Machine learning pipeline
Deployed service
Non-AI Software
A Data acquistion over (noisy) communication channels
B In motion data tagging is detecting corrupt/missing values due to environmental noise and signal processing errors
C Machine learning pipelines to generate services for event detection and virtual sensors for erroneous data repair
Domain expert expectations
Data quality metrics
D Profiling and evaluating data quality expectations

Figure 1: Edge-Cloud AI pipeline architecture to tame data quality.

when sensor reading variance increases. Factors such as material corrosion and damage affect data drift (i.e., change in the input data that affect the model's ability to make accurate predictions). We present a decentralized edge-cloud AI pipeline architecture for taming data quality to address the aforementioned issues. The architecture supports our two ML-based data quality pipelines (erroneous data repair and unsupervised data validation pipelines in Figure 1) accompanied by our in-motion data tagging solution. Having an AI engineering perspective (AI engineering dimensions [6]), we address the engineering challenges of realizing this architecture in industrial production environments.

Sensor measurements are often corrupted or have missing values due to electromagnetic interference, packet loss, or signal processing faults. The first data quality task is to detect and tag erroneous and missing values on the edge gateway (*in-motion data tagging*). Our ML pipelines leverage recurrent patterns in IIoT data to (i) generate virtual sensors repairing missing/corrupt data in one sensor based on high-quality data from other sensors (*erroneous data repair pipeline*[1]) and (ii) determine events and deviations in an unsupervised manner (*unsupervised data validation pipeline*[2]). We containerize and deploy their services (ML models) on edge for real-time inference and the cloud for inference on historical data. Domain experts specify heuristics to profile data and generate quality metrics (*data profiling service*). These metrics help maintain high data quality standards for AI-based applications and auditing.

In-motion Data Tagging

Tagging in-motion high-frequency data is the first step in detecting data quality issues at the data source in real-time. In-motion tagging is crucial to prevent erroneous data in posterior statistical analysis and closed-loop machine control algorithms. For instance, active control for vibration damping could apply erroneous responses due to unexpected signal peaks. We deploy our in-motion data tagging solution on edge to label parts of time-varying sensor data as corrupt/missing with error codes (see Figure 1). Tagging erroneous values instead of removing them helps identify data for posterior data repair. For instance, a low signal-to-noise (SNR) value could be identified as erroneous. We compare the peaks in the data before and after data repair as some may need to match.

Sensor failure due to hardware faults. Sensors fail due to unpredictable environmental conditions (e.g., too high temperature) or cable disconnection. Failures in Integrated Electronic

[1] https://github.com/sintef-9012/erdre
[2] https://github.com/ejhusom/udava

Piezo-Electric (IEPE) sensors can be detected by filtering bias voltage from the sensor output signal. If the bias voltage is not a constant value or is null, the data is marked with an error code.

Sensor failure due to electromagnetic interference (EMI). High-frequency noise cannot be easily filtered when EMI is in the measurement range of the sensors. We find the SNR ratio (measure the sensor signal level and noise floor) to discard EMI. If it is below 2:1, the measurement is noisy and marked with an error code. The SNR of 2:1 gives a vibration sensor the capacity to measure problematic vibrations despite background noise.

Sensor failure due to signal processing errors. Errors such as aliasing, ski-slope, spectral leakage, or jitter may arise after post-processing sensor data (e.g., Fast Fourier Transform). Aliasing and leakage can be avoided by low-pass filters and Hanning windows, respectively. We detect Ski-slope in the frequency spectrum of a signal where frequency appears high at low speeds and very low at high ones. We compute deviations in periodicity with a reference clock to detect jitter.

Erroneous Data Repair

We provide an ML pipeline (Figure 2(a)) that automatically repairs erroneous data identified by our in-motion data tagging solution. The pipeline generates a virtual sensor that estimates the signal of interest based on data from other sensors in the same IIoT system.

The pipeline preprocesses raw data from input sensors and trains an ML model (Step 1 in Figure 2(a)). The data preprocessing entails the selection of window size and features from the input to predict one or more values of the target sensor. The model is evaluated based on a metric expressing the prediction error concerning ground truth for unforeseen target sensor data, mean squared error or R^2-score.

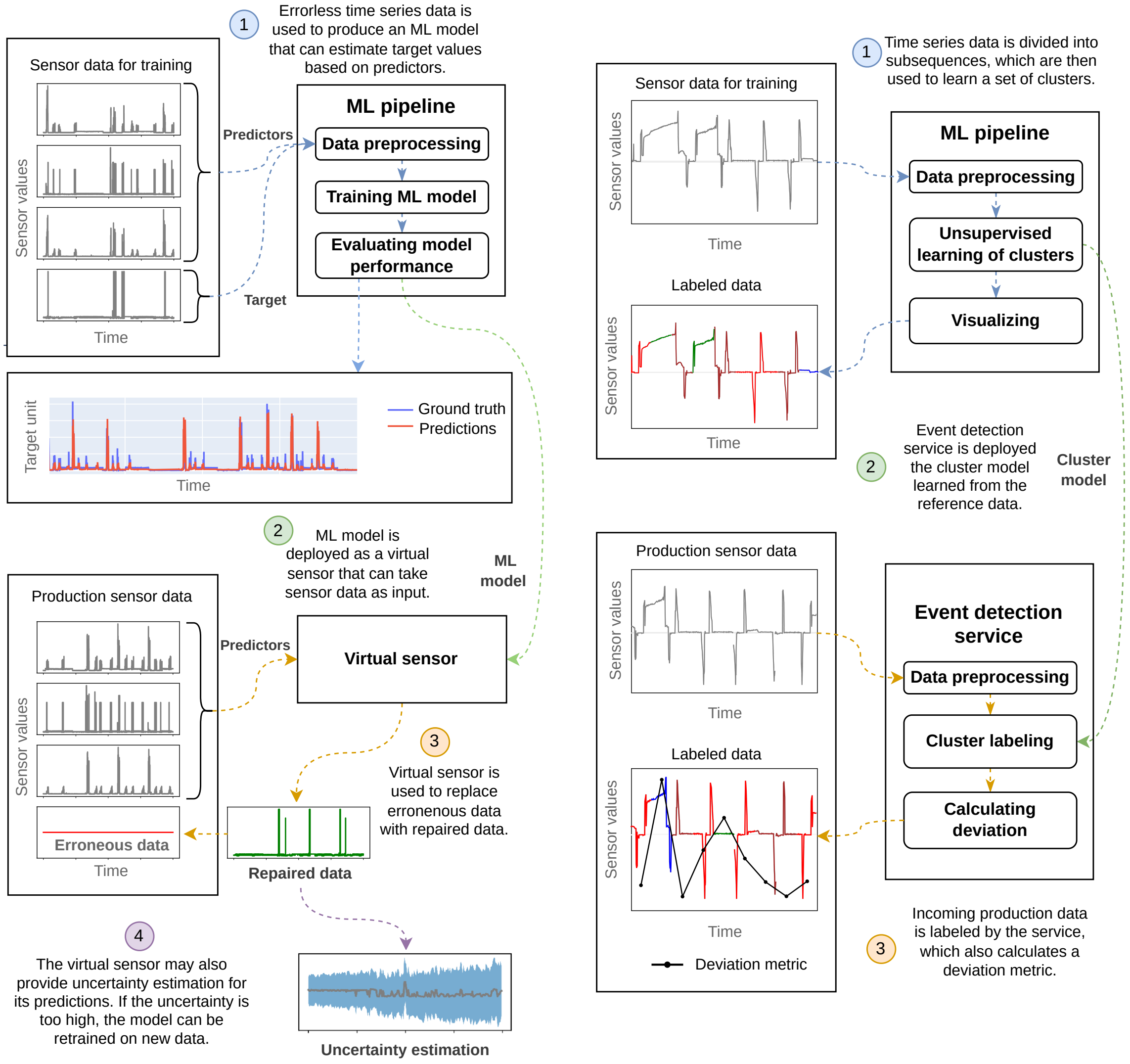
The trained model is containerized with an API and deployed as a virtual sensor (Step 2). The virtual sensor takes production sensor data as input and repairs its erroneous part (Step 3). It can be deployed on edge for real-time repair, while another virtual sensor with a larger input window repairs historical data on the cloud. Virtual sensors for erroneous data repair are prone to *concept drift* due to process changes (e.g., different parts produced) and the environment (e.g., vibrations from other machines, high temperature). They cannot generate accurate values for the faulty sensor when input data is *out of distribution*. Uncertainty estimation [8] in the performance of virtual sensors can autonomously guide the generation and deployment of new virtual sensors using recent data (Step 4). Bayesian Neural Networks (BNNs) can determine the uncertainty of a virtual sensor's performance in the form of confidence intervals for the prediction [9]. When confidence intervals are above a threshold, it is necessary to store new data and train a new virtual sensor based on new and some old data to minimize *catastrophic forgetting*. During this adaptation, the virtual sensor may not be usable. Hence, we encounter a trade-off between adaptation time and virtual sensor performance. Moreover, it is hard to train BNNs due to additional parameters to train probability distributions.

Unsupervised Data Validation

IIoT data acquired during industrial processes can reveal transitions in process behavior reflecting normal operation or process shifts and drifts leading to product defects. Process shifts and drifts are unexplained or unexpected trends of a measured process parameter(s) away from its intended target value in time-ordered analysis. Our unsupervised data validation pipeline [10] automatically discovers reference patterns representing modes of process behavior in training data from a reference production cycle. Its event detection service tracks deviations (process shifts and drifts) in production data by checking the recurrence of these patterns (see Figure 2(b)).

The pipeline splits the training time-series data into subsequences and extracts a low-dimensional summary vector of statistical features (Step 1 in Figure 2(b)). It uses k-means (when the number of clusters is known) or mean-shift clustering (when unknown) to assign feature vectors into clusters.

The cluster model (set of cluster centers) is deployed in an event detection service (Step 2). The service computes the feature vectors of the production data and assigns labels to its subsequences (Step 3). The labels are employed to detect events, such as anomalies where data points are too far from cluster centers in optimal


1
Errorless time series data is used to produce an ML model that can estimate target values based on predictors.
Sensor data for training
Sensor values
Time
Predictors
Target
ML pipeline
Data preprocessing
Training ML model
Evaluating model performance
Target unit
Time
Ground truth
Predictions
2
ML model is deployed as a virtual sensor that can take sensor data as input.
ML model
Production sensor data
Sensor values
Predictors
Virtual sensor
Erroneous data
Time
Repaired data
3
Virtual sensor is used to replace erroneous data with repaired data.
4
The virtual sensor may also provide uncertainty estimation for its predictions. If the uncertainty is too high, the model can be retrained on new data.
Uncertainty estimation
1
Time series data is divided into subsequences, which are then used to learn a set of clusters.
Sensor data for training
Sensor values
Time
Labeled data
Sensor values
Time
ML pipeline
Data preprocessing
Unsupervised learning of clusters
Visualizing
2
Event detection service is deployed the cluster model learned from the reference data.
Cluster model
Production sensor data
Sensor values
Time
Labeled data
Sensor values
Time
Deviation metric
Event detection service
Data preprocessing
Cluster labeling
Calculating deviation
3
Incoming production data is labeled by the service, which also calculates a deviation metric.

(a) Erroneous data repair

(b) Unsupervised data validation

Figure 2: ML pipelines for (a) Erroneous data repair and (b) Unsupervised data validation. We can use any supervised ML algorithm to obtain the model in (a), but deep learning techniques have especially proved efficient for time series analysis [7]. The pipeline in (a) can be used in any domain dealing with time-series data. We use unsupervised ML algorithms in (b). We designed the pipeline in (b) for the manufacturing industry due to the repetitiveness of production cycles that enables learning meaningful clusters. We evaluated the performance of our data repair and validation approaches in our case studies. Virtual sensors in (a) effectively repair erroneous data in our case studies when redundant/correlated sensors are available. The best ML architecture in our case studies based on R^2 and MSE scores is Long Short-Term Memory Network (LSTM). LSTM models have good accuracy in most cases but need more training time. Convolutional Neural Network (CNN) models are a good compromise between performance and training time. The event detection service in (b) assigns different behavior patterns to different clusters in our case studies, allowing for unsupervised detection of various forms of behavior. The deviation metric helps validate data over multiple production cycles and reason about the root cause of deviation by comparison with other parameters (e.g., tool wear).

production cycles. The service also calculates a deviation metric by checking the sum of distances between feature vectors and cluster centers. The metric increases when there are drifts in process behavior or changes in process parameters.

Data Profiling

Domain-specific knowledge about data quality can help detect quality issues that the pipelines cannot. For instance, temperature readings exceeding 120 degrees and more than five percent of data points are invalid since several occurrences of such temperatures could damage machine components. Machine tool temperature and force measurements often have a linear correlation. It is crucial to quantify the limits of this correlation and ensure that nonlinear behavior is detected before catastrophic consequences. Our architecture adopts Great Expectations (GE)[3], an open-source data quality framework that enables engineers to write *expectations* (assertions) on data for domain-specific data quality checks. GE generates code from expectations to analyze data and obtain interactive hypertext data documents. These documents profile data and assess data quality based on satisfaction of expectations. They are beneficial for auditing data acquisition and persistence from industrial processes. Furthermore, they help create a data quality culture by making engineers confident in their data-driven decisions and corresponding uncertainty estimates.

Case studies: Engineering the pipelines for industrial settings

Numerous challenges arise in engineering our data quality pipelines in Figure 1 for industrial production environments. Therefore, we analyzed the pipelines and the proposed architecture by using AI engineering dimensions [6] and two different industrial settings in the manufacturing domain: (a) broaching fir tree slots for jet engine turbine discs performed at CFAA - Advanced Manufacturing Centre for Aeronautics in Spain and (b) high-speed CNC milling of car engine cylinder heads at Renault factory in Spain.

Deployment infrastructure. We can deploy our pipelines on a standalone machine, edge device, or the cloud as a docker container with access to a time-series database (e.g., InfluxDB) or an API provided by a data acquisition system. The pipelines acquired the training data in the aerospace case from the SAVVY data systems edge device[4]. They obtained the automotive data through the KASEM cloud API[5] developed to maintain manufacturing machines. The high-frequency automotive data were stored in the cloud infrastructure KASEM E-maintenance. We experienced several deployment infrastructure-related challenges in the pipeline design for the *edge-cloud continuum* while adhering to data privacy and security constraints (e.g., sensor data is the intellectual property of data producers). Therefore, many scientific and engineering contributions lie in designing distributed ML architectures [11] for a fleet of distributed sensors, edge devices, and the cloud. When our pipelines are deployed on a standalone machine or edge device (often local “on-premises”), data producers (e.g., manufacturers) handle data security based on security standards such as ISO/IEC 27000 and ISA/IEC-62443 series for industrial cybersecurity. In cloud-based deployment, sensitive data are sent from data producers to the Cloud infrastructure using standard protocols (e.g., TLS[6]) to ensure the transmitted data security.

Real-time decision-making. One goal is to create real-time inference services (event detection service and virtual sensors on edge gateway in Figure 1). The pipelines are invoked to create an inference service based on the availability of training data start and end timestamps (specified by a domain expert or automatically derived based on product quality information). The services are containerized and deployed on edge to achieve real-time decision-making.

Model building and versioning. Our pipelines build ML models as often as necessary, but mostly when new reference (training) data is available. Data freshness, a quality dimension referring to how up-to-date the data is, affects model creation. When it is low, the pipelines employ new reference data to create a new model (automatic, e.g., when product quality and tool wear information in our case studies are available

[3] https://greatexpectations.io/

[4] https://www.savvydatasystems.com/

[5] https://www.predict.fr/produits-services/logiciels/

[6] https://datatracker.ietf.org/doc/html/rfc8446

for unsupervised data validation). In addition, uncertainty estimation autonomously guides the generation and deployment of new virtual sensors (models) using recent data. Models are stored and versioned as binary files using a chronological system in Git (with DVC[7]).

Integration of models and components. Data owners/producers invoke ML models as a web server or docker container. These models are deployed on-premises on an edge device or the cloud of data owners (see Figure 1). They provide a simple and open API (without security authentication) for receiving input (training and production data) and sending output (clusters and deviations or repaired data). The realization of our edge-cloud architecture for our case studies was highly impacted by where the predictive maintenance is done. The validation of fir tree slots in the aerospace case was at the edge device of the SAVVY data systems. Therefore, we deployed the ML models as a simple web service in this resource-constrained device. The data acquired from a fleet of machines in the automotive case was transferred to the cloud having virtually unlimited resources. There, we containerized the ML models as a service for validating and repairing data in parallel on more than one CNC machine in the automotive factory.

The challenges in integrating models and components are principally related to the evolution of the industrial processes and, consequently, the data and model evolution. Although we consider manufacturing processes for producing the same products or parts, there might be minor modifications to product specifications and process parameters (e.g., the need to ramp up production) that can render ML models obsolete (e.g., concept drift). Obsolete models can be addressed by uncertainty estimation and continual learning. Our architecture can be a basis for implementing continual learning mechanisms, such as online training, replay, and knowledge distillation. Therefore, there is a need to investigate continual learning [12] and domain adaptation [13] in conjunction with the continuous deployment of ML models [14].

DataOps. We developed and deployed our pipelines on the cloud following the DataOps discipline [15]. The operational IIoT data and up-to-date domain knowledge from field experts were the means to improve the pipelines. For example, we received new data and requirements for the data repair pipeline, which led to generating and deploying new versions of virtual sensors. Uncertainty estimation, in turn, is a significant part of DataOps and provided feedback to update virtual sensors in our case studies.

Implications and way forward

Data quality in IIoT should be documented and traceable along the entire production line/industrial processes over time. We introduce the concept of *data quality hallmarks* (certificates of data quality-related information in IIoT) in our EU InterQ[8]. Data quality metrics, events, and deviations should be part of a hallmark for a well-defined asset for a given time. Data quality hallmarks can be generated at regular intervals and traceable using immutable blockchain technology. We have been developing a solution (InterQ-TrustedFramework) for end-to-end industrial data traceability, trust, and security.

Data quality hallmarks should be connected to process and product quality hallmarks. Process quality is the degree to which the manufacture of a product meets its process requirements. Product quality entails how well a product/artifact satisfies customer needs, serves its purpose, and meets industry standards. Data, process, and product quality hallmarks should be traceable using blockchain. Traceability will enable isolating root causes for faults in complex IIoT ecosystems. Smart contracts between the stakeholders of an IIoT ecosystem should be used to accept/reject *quality transactions* from different stakeholders to adhere to stakeholder agreements on quality and build trust more autonomously.

The data quality solution standardization for AI-enabled IIoT is an active area for standardization bodies such as ISO8000. Data quality validation and repair should be standard for each IIoT level (sensor, edge, and cloud). Then, the traceability of data quality hallmarks can positively impact traditional industries trying AI solutions to optimize their processes.

[7] Open-source Version Control System for Machine Learning Projects, https://dvc.org/

[8] https://interq-project.eu/

Manufacturers implementing our architecture will gain more control over the data and the process, supporting more efficient production. We expect more control over the data adjusting an ongoing machining process will outweigh the cost of setting up the infrastructure needed.

■ REFERENCES

1. M. Andronie, G. Lăzăroiu, M. Iatagan, C. Uță, R. Ștefănescu, and M. Cocoșatu, “Artificial intelligence-based decision-making algorithms, internet of things sensing networks, and deep learning-assisted smart process management in cyber-physical production systems,” *Electronics*, vol. 10, no. 20, p. 2497, 2021.
2. W. Yu, T. Dillon, F. Mostafa, W. Rahayu, and Y. Liu, “A global manufacturing big data ecosystem for fault detection in predictive maintenance,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 183–192, 2019.
3. A. A. Alwan, M. A. Ciupala, A. J. Brimicombe, S. A. Ghorashi, A. Baravalle, and P. Falcarin, “Data quality challenges in large-scale cyber-physical systems: A systematic review,” *Information Systems*, vol. 105, p. 101951, 2022.
4. A. Corallo, A. M. Crespino, V. Del Vecchio, M. Lazoi, and M. Marra, “Understanding and defining dark data for the manufacturing industry,” *IEEE Transactions on Engineering Management*, 2021.
5. S. Akter, G. McCarthy, S. Sajib, K. Michael, Y. K. Dwivedi, J. D’Ambra, and K. Shen, “Algorithmic bias in data-driven innovation in the age of ai,” p. 102387, 2021.
6. J. Bosch, H. H. Olsson, and I. Crnkovic, “Engineering AI systems: A research agenda,” in *Artificial Intelligence Paradigms for Smart Cyber-Physical Systems*, 2021, pp. 1–19.
7. J. C. B. Gamboa, “Deep learning for time-series analysis,” *arXiv preprint arXiv:1701.01887*, 2017.
8. N. Jourdan, S. Sen, E. J. Husom, E. Garcia-Ceja, T. Biegel, and J. Metternich, “On the reliability of machine learning applications in manufacturing environments,” in *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.
9. J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher *et al.*, “A survey of uncertainty in deep neural networks,” *arXiv preprint arXiv:2107.03342*, 2021.
10. E. Husom, S. Tverdal, A. Goknil, and S. Sen, “Udava: An unsupervised learning pipeline for sensor data validation in manufacturing,” in *1st International Conference on AI Engineering - Software Engineering for AI (CAIN’22)*, 2022, pp. 159–169.
11. J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, “A survey on distributed machine learning,” *ACM Computing Surveys*, vol. 53, no. 2, pp. 1–33, 2020.
12. B. Maschler, H. Vietz, N. Jazdi, and M. Weyrich, “Continual learning of fault prediction for turbofan engines using deep learning with elastic weight consolidation,” in *ETFA’20*, 2020, pp. 959–966.
13. M. Azamfar, X. Li, and J. Lee, “Deep learning-based domain adaptation method for fault diagnosis in semiconductor manufacturing,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 33, no. 3, pp. 445–453, 2020.
14. I. Prapas, B. Derakhshan, A. R. Mahdiraji, and V. Markl, “Continuous training and deployment of deep learning models,” *Datenbank-Spektrum*, vol. 21, no. 3, pp. 203–212, 2021.
15. A. R. Munappy, D. I. Mattos, J. Bosch, H. H. Olsson, and A. Dakkak, “From ad-hoc data analytics to dataops,” ser. ICSSP ’20, 2020, p. 165–174.

Sagar Sen is a Senior Research Scientist at SINTEF Digital. He received his Ph.D. from the University of Rennes 1/INRIA Rennes, France and M.Sc. from McGill University, Canada both in computer science. His research interests are in engineering and testing of lifelong AI systems for application domains such as manufacturing and health. Contact him at sagar.sen@sintef.no.

Erik Johannes Husom is a Master of Science at Trustworthy Green IoT Software research group at SINTEF Digital. He received his MSc in Computational Physics from the University of Oslo (Norway) in 2021. His research interests include applied machine learning, AI engineering, and responsible AI. Contact him at erik.husom@sintef.no.

Arda Goknil is a senior research scientist at SINTEF Digital in Oslo, Norway. His research interests include model-driven software engineering, software testing, AI engineering, and intermittent computing. Dr. Goknil received his doctorate in software engineering from University of Twente in the Netherlands. Contact him at arda.goknil@sintef.no.

Simeon Tverdal is Master of Science at Trustworthy Green IoT Software research group at SINTEF Digital. He received his M.Sc. in computer science from the University of Oslo, Norway, in 2021. His

research interests include cybersecurity, applied machine learning, and trustworthy AI. Contact him at simeon.tverdal@sintef.no.

Phu Nguyen is a senior research scientist at SINTEF Digital in Oslo, Norway. He has an international education and research background, from Vietnam (BSc) to the Netherlands (MSc), Luxembourg (Ph.D.), and Norway. His current research interests include the IoT, Microservices, Edge Computing, Cloud Native, DevOps, Security By Design. Contact him at phu.nguyen@sintef.no.

Iker Mancisidor is an engineer in the Dynamics and Control research group at Ideko technology center, since 2009. He obtained a Ph.D. degree in Mechanical Engineering from the University of the Basque Country in 2014, and he spent 6 months as a postdoctoral fellow at the Precision Controls Laboratory of the University of Waterloo (Canada) in 2016. His research interests include machine-tool chatter, active damping, electromagnetic actuators, mechatronic systems, monitoring, machine-tool control and automation, and modal analysis. Contact him at imancisidor@ideko.es.