# Trustworthy anomaly detection for smart manufacturing

Li, Z.

# Chapter 1

# Introduction

The utilization of artificial intelligence in manufacturing processes has attracted considerable interest from both academic researchers and industry practitioners [1, 2]. This growing field is commonly known as *smart manufacturing* [2]. Anomaly detection, in particular, is a key area of research with numerous successful and potential applications in smart manufacturing [3], where machine learning algorithms analyze data collected from sensors and other sources to identify abnormal patterns and facilitate predictive maintenance. However, applying anomaly detection techniques to real-world use-cases in smart manufacturing often presents obstacles, motivating the development of this dissertation.

In this chapter, we will first provide an overview of smart manufacturing and related concepts, setting the stage for the broader context of this dissertation and outlining the current research landscape. We will then introduce the concept of trustworthy anomaly detection, which is central to this work. This section includes a review of fundamental concepts and relevant literature on anomaly detection and trustworthy AI. Building on this, we will present trustworthy anomaly detection in the context of smart manufacturing, in which we summarize the main contributions of this dissertation. Furthermore, we will explore the concepts of model-centric AI and data-centric AI, highlighting the necessity of integrating both perspectives for data-driven smart manufacturing. Following this, we will formally present the research questions addressed in this dissertation and the corresponding contributions made to answer them. Finally, we will outline the structure of this dissertation and provide an overview of the remaining chapters.

This chapter is partially based on the following publication:

- Zhong Li, Yuxuan Zhu, and Matthijs van Leeuwen. "A survey on explainable anomaly detection." ACM Transactions on Knowledge Discovery from Data 18, no. 1 (2023): 1-54.

## 1.1 Smart Manufacturing

This section introduces the concepts of Industry 4.0, Smart Manufacturing, Cyber-Physical Systems, Digital Twin, and Predictive Maintenance.

### 1.1.1 Industry 4.0 and Smart Manufacturing

The evolution of mass industries is generally divided into four stages: 1) Industry 1.0, which relied on steam and water-powered machinery; 2) Industry 2.0, characterized by the use of electricity and mass production techniques; 3) Industry 3.0, which introduced digitization and automation; and 4) Industry 4.0, which focuses on cyber-physical systems [4]. Specifically, with the rapid development of information, communication, and other technologies—such as the Internet of Things, Cloud Computing, Big Data, Robotics, and Artificial Intelligence—the manufacturing industry has entered the fourth stage of industrial production, known as Industry 4.0, since the early 2010s [5]. However, the concept of Industry 4.0 does not have a universally accepted definition. A commonly recognized definition is presented below.

**Definition 1.1** (Industry 4.0)**.** Industry 4.0 is commonly viewed as the convergence of various advanced technologies, such as Internet of Things, Cloud Computing, Big Data, Robotics, and Artificial Intelligence, aimed at integrating the physical and virtual worlds through cyber-physical systems (CPS). [4]

Particularly, Smart Manufacturing is a subset within the broader Industry 4.0 framework. It focuses specifically on applying these advanced technologies to improve manufacturing processes. Smart manufacturing has garnered significant attention from industry, government organizations, and academia. Despite this interest, a universally accepted definition of *smart manufacturing* has yet to be established [6, 7]. We present its more commonly used definition in Definition 1.2.

**Definition 1.2** (Smart Manufacturing)**.** According to the National Institute of Standards and Technology, "smart manufacturing is fully integrated, collaborative manufacturing system that respond in real time to meet changing demands and conditions in the factory, in the supply network and in customer needs." [6]

Besides, smart manufacturing is also commonly known as intelligent manufacturing. However, Wang et al. [2] performed detailed comparisons to show the differences between these two concepts. In brief, intelligent manufacturing is defined as the intersection of artificial intelligence and manufacturing [8], and this term has been used since 1980s. Zhong et al. [8] pointed out that the manufacturing has been shifted from knowledge-based intelligent manufacturing to knowledge-enabled and data-driven smart manufacturing, where the term "smart" refers to the creation and usage of data. Therefore, Toben et al. [9] considered smart manufacturing as a new version of intelligent manufacturing that highlights the use of advanced technologies and analytics methods. Specifically, Zheng et al. [5] proposed a conceptual framework of smart manufacturing systems for Industry 4.0, and this framework consists of the following dimensions: smart design, smart machining, smart monitoring, smart control, and smart scheduling.

### 1.1.2 Cyber-Physical Systems and Digital Twin

Tao et al. [10] indicated that cyber-physical integration is a crucial prerequisite for and the core of smart manufacturing. Particularly, Cyber-Physical Systems (CPS) and Digital Twins (DTs) are two preferred means to achieve such an integration. The formal definition for CPS is given in Definition 1.3, while the formal definition for DT is given in Definition 1.4.

**Definition 1.3** (Cyber-Physical Systems)**.** "Cyber-Physical Systems are multidimensional and complex system that integrate the cyber world and the dynamic physical world. Through the integration and collaboration of computing, communication, and control, CPS provide real-time sensing, information feedback, dynamic control, and other services." [10, 11, 12]

**Definition 1.4** (Digital Twin)**.** A Digital Twin is a high-fidelity digital replica of a physical asset, process, or system in virtual space. It uses real-time data and simulations to mirror the physical counterpart's state, performance, and behavior. [10, 13]

Tao et al. systematically analyzed and compared Cyber-Physical Systems (CPS) and Digital Twins (DTs) across several aspects: origin, cyber-physical mapping, hierarchical modeling, and core elements [10]. The relationship between CPS and DTs can be summarized as follows: CPS are foundational systems that integrate physical processes with digital controls and computations [14], while DTs are specific digital

replicas of physical entities or processes [15]. DTs rely on CPS for real-time data acquisition and interaction, as CPS emphasize the powerful computing and communication capabilities of the cyber world more strongly when compared to DTs [16]. Conversely, DTs focus on creating high-fidelity virtual models that replicate the physical system's geometry, structure, behavior, rules, functionality, and other dynamics [10]. Consequently, DTs can provide detailed insights into operations, enabling predictive maintenance, optimization of manufacturing processes, and what-if analysis through simulations [10].

### 1.1.3 Predictive Maintenance

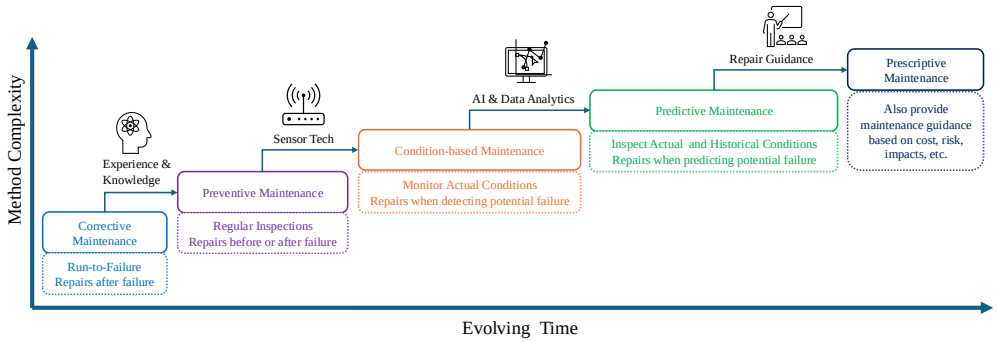


**Figure 1.1:** Maintenance strategies can be divided into five main stages based on their evolution over time. (Inspired by Figure 1 in [17])

In this dissertation, we focus on the *predictive maintenance* aspect, which is one of the major goals of Industry 4.0 [18]. Specifically, predictive maintenance is the application that can leverage data and simulations from DTs to anticipate and plan maintenance activities before failures occur. On a broader level, maintenance plays a pivotal role in industrial applications, as effective strategies prevent unexpected downtimes, lower operational costs, and potentially extend the remaining useful lifetime of machinery [17]. Consequently, maintenance practices have attracted substantial attention from both industry and academia [19, 20, 17], showing continuous evolution over recent years. According to [17], these practices can broadly be categorized into the following five stages, as shown in Figure 1.1:

1) **Corrective Maintenance**: Also known as reactive maintenance, this run-to-

failure strategy involves performing repairs or replacements only after equipment has failed. It focuses on addressing issues after they occur, rather than preventing them in advance.

2) **Preventive Maintenance**: Also known as planned maintenance, this approach involves scheduling regular inspections and maintenance activities to prevent equipment failures. It is carried out proactively, even when the equipment is operating normally.

3) **Condition-based Maintenance**: It involves monitoring the actual condition of equipment based on real-time data, aiming to determine when maintenance is needed. Maintenance actions are performed only when specific indicators show that performance is deteriorating or a failure is likely to occur.

4) **Predictive Maintenance**: It builds upon Condition-based Maintenance by using advanced analytics (such as machine learning and data mining techniques) to predict when equipment will fail based on its current condition and historical data. It not only monitors equipment conditions but also forecasts future failures with a higher degree of accuracy, enabling more efficient scheduling of maintenance activities.

5) **Prescriptive Maintenance**: It refers to more advanced strategies that not only predict but also recommend optimal maintenance actions to prevent or mitigate equipment failures. It provides guidance on the best interventions, considering factors like cost, risk, and operational impact.

While the more recent prescriptive maintenance strategy offers several advantages over predictive maintenance, its implementation in real-world applications remains challenging without a robust foundation in predictive maintenance. Moreover, predictive maintenance itself faces numerous unresolved issues [17], such as: 1) industrial data is susceptible to errors due to harsh environmental conditions, sensor faults, or transmission errors; 2) the volume of data is large and growing exponentially; 3) data types can vary in actual industrial applications, resulting in different modalities of data such as videos, audios, texts, images, time series, graphs, logs, and tabular data; and 4) industrial environments and production systems can differ widely between manufacturers. Additionally, Digital Twins (DTs), which can be modeled in the virtual world for each critical component in the physical world, provide a promising approach to enhancing predictive maintenance [5]. For these reasons, our focus is on predictive maintenance. Formally, *predictive maintenance* can be defined as follows.

**Definition 1.5** (Predictive Maintenance)**.** "Predictive maintenance is a set of activities that detect changes in the physical condition of equipment (signs of failure) in order to carry out the appropriate maintenance work for maximizing the service life of equipment without increasing the risk of failure." [21]

However, with the rapid advancement of technology and increasing demands in industry, the traditional definition of predictive maintenance (namely Definition 1.5) has become somewhat overly generalized and even outdated. To address evolving needs, Predictive Maintenance 4.0 (PdM 4.0) has emerged, aligned with the principles of Industry 4.0, providing a blueprint for more intelligent and efficient predictive maintenance systems [22, 23, 24, 21]. PdM 4.0 leverages advanced technologies such as the Internet of Things (IoT) for data collection, Big Data techniques for data preprocessing, and Data Mining and Machine Learning techniques for in-depth data analysis. These technologies collectively enable decision support systems that accurately predict when maintenance should be performed, thereby preventing unexpected breakdowns and minimizing downtime. As shown in Figure 1.2, the system architecture for PdM 4.0 generally consists of the following components [22, 23, 24, 21]:

- **Data Acquisition:** Sensors are employed to collect data such as temperature, humidity, and vibration from physical assets. This data is then transmitted through networks using Internet of Things (IoT) technologies.
- **Data Pre-processing:** The collected data is stored in data warehouses, which can be in the Cloud, where Big Data techniques are used for data cleaning, integration, feature extraction, and transformation.
- **Data Analysis:** Data mining and machine learning techniques are leveraged to perform two main tasks:
  - *Diagnosis:* Involves anomaly detection (unsupervised or semi-supervised) and anomaly classification (supervised) in the data.
  - *Prognosis:* Focuses on predicting future anomalies and potential failures.
- **Decision Support:** Utilizes the results from data analysis to conduct fault detection, fault isolation, fault prediction, and degradation assessment, thereby supporting maintenance decision-making.
- **Maintenance Implementation:** Implements maintenance activities in the physical world according to the maintenance decisions generated in the cyber world.

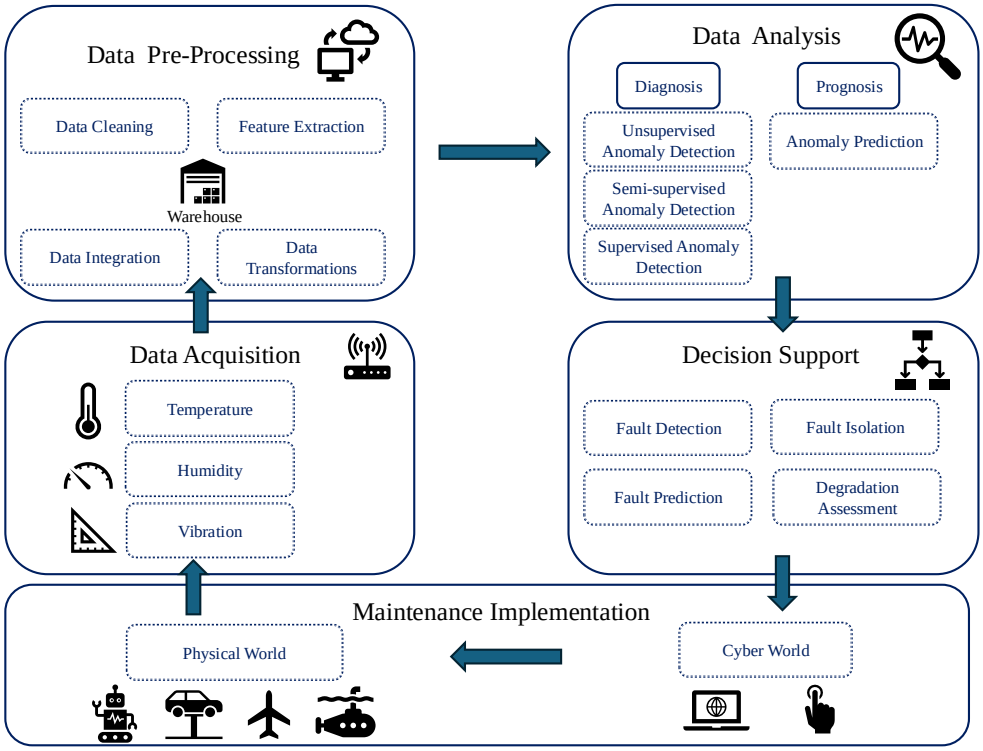


**Figure 1.2:** The system architecture for PdM 4.0 generally consists of five components. (Inspired by Figure 6 in [24])

As emphasized in [18], anomaly detection is central to Predictive Maintenance, with a primary focus on identifying anomalies in equipment at early stages and alerting the manufacturing technicians to initiate maintenance activities. Moreover, the abnormal behavior of data can stem from various causes. As Nunes et al. [17] highlighted, the detected anomalies can be explored for decision support in two main ways based on their causes of anomalies: 1) if anomalies are due to noise from sensor malfunctions, low battery, or other external disturbances, they are considered irrelevant and should be removed to prevent misinterpretation; 2) if anomalies result from relevant events, such as potential equipment failures or process issues, they should be automatically detected from sensor data and utilized by models to predict the remaining useful lifetime, prompting further analysis and potentially leading to maintenance actions. The main contributions of this dissertation center around anomaly detection, which will be introduced in the next section. However, rather than limiting our scope to

anomaly detection in the specific context of predictive maintenance and sensor data, we emphasize that many of our developed approaches and findings are applicable to more generic scenarios.

## 1.2 Trustworthy Anomaly Detection

In this section, we will begin by reviewing the fundamental concepts, terminologies, and relevant literature in anomaly detection. Next, we will provide a brief overview of trustworthy artificial intelligence. Building on this foundation, we will highlight the intersection of these two fields, where we will position the contributions of the various research papers included in this dissertation.

### 1.2.1 Anomaly Detection

We first introduce the concepts of *anomaly* and *anomaly detection*. Then, we present a categorization of anomaly detection techniques based on the availability of labels.

**Definition 1.6** (Anomaly)**.** An anomaly is an object that is notably different from the majority of the remaining objects.

Depending on the specific application domain, an anomaly can also be called an outlier or a novelty. Moreover, it may also be known as an unusual, irregular, atypical, inconsistent, unexpected, rare, erroneous, faulty, fraudulent, malicious, unnatural, or strange object [25]. Except for a few works such as [25], the term *outlier* is used as a synonym for *anomaly* in most research. For consistency, we will use the term *anomaly* in this dissertation.

**Definition 1.7** (Anomaly Detection)**.** The process of identifying anomalies is called *anomaly detection*.

Since the seminal work in [26], anomaly detection has been well studied and there exists a plethora of comprehensive surveys and reviews on it, including but not limited to [27, 28, 29, 30, 31, 32, 33, 34, 35, 36]. Particularly, depending on the specific type of data and the application domain, the techniques used to perform anomaly detection can be different. The commonly seen data types include tabular data, time series, text, image, video, audio, graph, and log data. In this dissertation, we primarily focus on the following three types of data:

- Tabular data, which refers to data that is organized into a table, where information consists of rows (observations) and columns (features or attributes);

- Log data, which refers to the detailed records generated by software or devices during their operation. These logs capture information about events that occur within a system, providing valuable insights into its performance, security, and usage patterns.
- Graph data, which represents information that is structured as a collection of nodes and edges that connect these nodes. This structure is used to model relationships between different entities in a way that makes the connections and associations between them easy to understand and analyze.

Anomaly detection techniques can be categorized into three types based on the availability of labels, regardless of the data type: supervised, unsupervised, and semi-supervised [31, 34]. Before introducing these categories, it is essential to understand two key concepts: 1) *Transductive learning*, which makes predictions directly for specific test instances without learning a general model for unseen data, thus lacking distinct training and test stages; and 2) *Inductive learning*, which involves training a model on labeled data to learn general rules that are then applied to unseen test instances [37], thus containing separate training and test stages.

- **Supervised anomaly detection** requires labeled instances of both normal and abnormal data during training, treating the task as an *imbalanced* binary classification problem due to the typically low ratio of anomalies (e.g., less than 5%). It is an inductive learning approach, where the model learns general rules from specific training examples and applies these rules to unseen test instances.
- **Unsupervised anomaly detection**, in contrast, does not require labeled training data and is usually performed in a transductive learning manner, without a separate training phase. These techniques are generally based on the assumption that normal instances occur far more frequently than abnormal ones in the dataset.
- **Semi-supervised anomaly detection** operates with partially labeled data, where the labeled instances may include both normal and abnormal examples, or more commonly, only normal examples. In this common scenario, most semi-supervised methods train on datasets exclusively consisting of normal data, aiming to model the normal data distribution during training. Any instances that significantly deviate from this distribution during inference are considered anomalies. Thus, it is also an inductive learning approach. Notably, many semi-supervised methods can function in an unsupervised manner while still being

inductive rather than transductive: given the typically low anomaly ratio, these methods use a subset of samples from the data to train the model, assuming that the training data contains few anomalies and the model can robustly learn the normal distribution.

Particularly, semi-supervised and unsupervised anomaly detection are the most commonly used techniques in manufacturing due to the scarcity of labels [3]. In this dissertation, we will primarily consider semi-supervised and unsupervised anomaly detection for the same reason.

### 1.2.2 Trustworthy Artificial Intelligence

**Definition 1.8** (Trustworthy AI)**.** "Trustworthy AI is a framework to ensure that a system is worthy of being trusted based on the evidence concerning its stated requirements. It makes sure that the users' and stakeholders' expectations are met in a verifiable way." [38]

As highlighted in [38], with the growing volume of research on trustworthy AI, reaching a consensus on a common set of principles to ensure AI trustworthiness is challenging. However, we argue that at least the following five principles should be considered [38, 39]:

- **Effectiveness**: AI systems must provide accurate predictions, as it is crucial for their use in replacing or assisting human decision-makers. Without accuracy, the utility of AI systems is compromised.
- **Explainability**: Stakeholders involved with the AI system should be able to understand the reasoning behind its decisions, ensuring transparency and trust in the system's outputs.
- **Generalizability/Robustness**: AI systems should function reliably across various conditions. Robustness refers to the ability of an AI system to handle execution errors, erroneous inputs, or unseen data, while generalizability pertains to the system's capability to make accurate predictions on unseen data [39]. It is important to note that the relationship between robustness and generalizability can be complex [40, 39]; for simplicity, they are treated as a unified concept in this dissertation.
- **Fairness**: AI systems should avoid introducing or perpetuating biases and discrimination against any individual or group within society.

- **Privacy**: AI systems must ensure that sensitive information is protected throughout its entire lifecycle. In particular, unauthorized use of data that can identify individuals or households should be strictly avoided [39].

### 1.2.3 Trustworthy Anomaly Detection in the Context of Smart Manufacturing

Anomaly detection has achieved a wide range of successful applications in many decision-critical domains. For example, anomaly detection algorithms are being used to diagnose diseases in healthcare [41]. In financial services, many banks use anomaly detection methods to detect abnormal behavior in credit card transactions [42]. In addition, the self-driving car manufacturing industry applies anomaly detection algorithms on camera data to detect corner cases [43]. In other decision-critical areas—such as spacecraft design—anomaly detection algorithms are used to detect sensor faults [44]. As we can see, anomaly detection systems for high-stakes decisions are deeply impacting our daily lives and society.

Despite significant successes, anomaly detection still faces many limitations. Particularly, given that these tasks are often safety-critical and can have life-changing consequences, a major concern is whether we can truly trust the results provided by these anomaly detection systems. Therefore, ensuring that anomaly detection systems operate in a trustworthy manner is essential when deploying them in real-world applications [45].

Anomaly detection is an important subfield of machine learning and data mining, both of which are critical components of the broader field of AI. Therefore, to assess the trustworthiness of an anomaly detection system, we should consider the five key principles established for trustworthy AI systems, as outlined in Chapter 1.2.2: effectiveness, explainability, generalizability, fairness, and privacy. However, this dissertation focuses specifically on anomaly detection in smart manufacturing, which typically involves data from machines rather than humans. Consequently, the principles of fairness and privacy will not be addressed. In the following, we will briefly introduce the principles of effectiveness, explainability, and generalizability to achieve trustworthy anomaly detection in the context of smart manufacturing.

**Principle 1: Effectiveness of Anomaly Detection.** Effectively identifying anomalies is the fundamental requirement for all anomaly detection systems [45]. Given a dataset, high accuracy in an anomaly detection system is typically achieved by selecting and deploying an appropriate detection model. However, in real-world

industrial applications, we encounter two key challenges: 1) Most anomaly detection research, since the seminal work in [26], has focused on simple tabular data, whereas real-world scenarios in smart manufacturing often involve complex data such as log events, time series, or graphs; 2) Even after converting complex data into simple form (e.g., through feature extraction), the resulting high dimensionality can reduce the effectiveness of many traditional anomaly detection methods in smart manufacturing. Particularly, Chapters 2 (to Challenge 1) and 3 (to Challenge 2) mainly make contributions in this aspect.

**Principle 2: Explainability of Anomaly Detection.** According to [46], we can define eXplainable Anomaly Detection (XAD) as *the extraction of relevant knowledge from an anomaly detection model concerning relationships either contained in data or learned by the model*, where the knowledge is considered relevant if it can provide insight into the anomaly detection problem investigated by the end-user. Particularly, providing anomaly detection results with corresponding explanations can help gain the trust of end-users in anomaly detection systems. Moreover, the explanations can also assist end-users to validate the anomaly detection results in unsupervised settings. Even more, explanations can potentially enable end-users to find the root causes of anomalies and thereby take remedial or preventive actions. For a long time, however, the anomaly detection community has mainly focused on detection accuracy, largely ignoring the interpretation of corresponding outcomes. More importantly, there is an increasing demand for explainability when deploying anomaly detection systems in industrial manufacturing. Chapters 4 and 5 mainly make contributions in this aspect.

**Principle 3: Generalizability of Anomaly Detection.** We define generalizability as the ability of an anomaly detection system to operate reliably under various conditions, which includes two main aspects: First, the anomaly detection system should perform reliably on unseen data, especially when there is concept drift in the data. Second, given that anomaly detection tasks are often unsupervised or semi-supervised, the system should perform consistently across different datasets, requiring the capability to automatically tune model hyperparameters without relying on data labels. Chapter 6 contributes to the first aspect, focusing on generalizability to unseen data, while Chapter 7 addresses the second aspect, emphasizing adaptability across various datasets.

## 1.3 Model-Centric AI and Data-Centric AI

In this section, we begin by introducing concepts in model-centric AI (and we will discuss how the included papers can be interpreted from this perspective in Chapter 1.4). Next, we explore concepts in data-centric AI (and we will illustrate how the papers presented in this dissertation relate to them in Chapter 1.4). Finally, we explain the necessity of integrating these two complementary approaches to further advance data-driven smart manufacturing.

**Definition 1.9** (Model-centric AI)**.** "Model-centric AI is the paradigm emphasizing the choice of the suitable model type, architecture, and hyperparameters from a wide range of possibilities for building effective and efficient AI systems." [47]

More specifically, model-centric AI focuses on enhancing the performance of AI systems by improving algorithms to work more effectively with existing, often fixed, datasets [48]. This approach emphasizes designing better model architectures, fine-tuning hyperparameters, developing more effective and efficient optimization techniques, and proposing new models types or learning paradigms [47]. Typically, the data is created once, and its quality and quantity remain consistent throughout the development cycle of the AI system, while substantial efforts are directed toward building more advanced learning models. Despite the huge successes of Model-centric AI in the past decades, it still suffers from some notable limitations [48]:

- Vulnerability to adversarial samples,
- Low generalization capacity.

**Definition 1.10** (Data-centric AI)**.** "Data-centric AI is the paradigm emphasizing that systematic design and engineering of data are essential for building effective and efficient AI systems." [47]

Data is an essential element in AI systems, which include anomaly detection systems. Recently, the significance of data has been greatly amplified by the rise of data-centric AI [49, 50]. Although data-centric AI is a relatively new concept [49, 50, 47, 51, 52, 53], many classic research topics such as data augmentation and feature selection can be considered as subfields of data-centric AI. Data augmentation aims to enrich the training dataset by adding slightly modified data instances, while feature selection attempts to reduce data complexity by keeping only relevant features. According to [49], there are three main data-centric AI objectives (the underlined parts are involved in this dissertation):

- **Training data development**, which includes *data collection*, *data labeling*, *data preparation* (such as data cleaning, feature extraction, and data transformation), *data reduction* (such as feature selection, dimension reduction, instance selection), and *data augmentation* (e.g., basic manipulation, deep learning approaches); Particularly, data collection and data labeling are considered data creation, and the rest is data processing;
- **Inference data development**, which consists of *in-distribution evaluation* (such as data slicing, algorithmic recourse), *out-of-distribution evaluation* (such as adversarial perturbation, distribution shift), and *prompt engineering*;
- **Data maintenance**, which contains *data understanding* (such as data visualization, data valuation), *data quality assurance* (such as quality assessment, quality improvement), and *data storage & retrieval* (such as resource allocation, query acceleration).

In summary, Data-centric AI focus on improving the quality, consistency, and richness of the data used to train and test AI models. In this paradigm, the focus shifts from constantly refining the model to ensuring that the data is clean, well-labeled, diverse, and representative of the problem domain. The idea is that with high-quality data, even simpler models can perform exceptionally well. Data-centric AI is often advocated for in situations where models are already highly optimized, and further gains can be achieved by addressing data issues rather than the models.

**Complementary views of model-centic AI and data-centric AI.** While there has been a recent shift in focus from model-centric AI to data-centric AI, we argue that these two paradigms are inherently complementary. In other words, although we emphasize the importance of data-centric AI, this should not diminish the role of model-centric AI. Successfully addressing challenges in smart manufacturing requires considering both the methods of action (namely algorithms on "how-to") and the insights hidden within the data (namely knowledge on "what-is") [48]. In this dissertation, we will approach the trustworthy anomaly detection problem from both data-centric and model-centric AI perspectives, recognizing them as twin drivers for advancing smart manufacturing.

## 1.4 Research Questions and Contributions

In this dissertation, we develop anomaly detection approaches primarily with the aim to enhance the manufacturing process of high-tech systems. However, it is important to

note that many of our developed approaches and findings are applicable to (and often even were designed for) more generic scenarios. As achieving trustworthy anomaly detection for smart manufacturing is non-trivial, we tackle them from two different and complementary perspectives (i.e., data-centic AI and model-centric AI), leading to two primary research questions, each with several sub-questions as outlined below:

Q1 **How to develop and/or improve anomaly detection for smart manufacturing from a data-centric AI perspective?**

First, we face challenges from a data-centric AI perspective (which systematically engineers the data used as input for the anomaly detection system [54]):

- **Complex data**. Not limited to simple tabular datasets, a real-world use case could involve complex data such as log events, time series, graphs, etc. We propose the following research question and make corresponding contributions to answer it:
  - *Q1.1*: How to deal with complex data in system logs to effectively detect and explain anomalies?
  - *Answer to Q1.1* (Contribution 1): *Graph Neural Network based Log Anomaly Detection and Explanation* [55], which will be presented in Chapter 2. In this chapter, we propose a graph-based method for unsupervised log anomaly detection, dubbed *Logs2Graphs*, which first converts event logs into attributed, directed, and weighted graphs, and then leverages graph neural networks to perform graph-level anomaly detection. Specifically, we introduce One-Class Digraph Inception Convolutional Networks, abbreviated as OCDiGCN, a novel graph neural network model for detecting graph-level anomalies in a collection of attributed, directed, and weighted graphs. Crucially, we furnish a concise set of nodes pivotal in OCDiGCN's prediction as explanations for each detected anomaly, offering valuable insights for subsequent root cause analysis.
- **High-dimensional data**. Even after transforming complex data to simple data (e.g., via feature extraction), the dimensionality of the resulting data can be very high, rendering many traditional anomaly detection approaches less effective. We propose the following research question and make corresponding contributions to answer it:

- *Q1.2*: How to handle high-dimensionality in system logs to effectively detect anomalies?
- *Answer to Q1.2* (Contribution 2): *Feature Selection for Fault Detection and Prediction based on Event Log Analysis* [56], which will be presented in Chapter 3. In this chapter, we develop a feature selection method for log-based anomaly detection and prediction. Specifically, our method consists of three main modules: the *Log Event Vectorization* module that converts semi-structured log texts into time series; the *Selection of Relevant Features* module that leverages Kendall rank correlation and Granger causality test to select log events for fault detection and prediction; and the *Removal of Redundant Features* module that utilizes Kendall rank correlation to reduce redundant log events.

Q2 **How to develop and/or improve anomaly detection for smart manufacturing from a model-centric AI perspective?**

On the other hand, we may face challenges from the model-centric AI perspective (which aims to produce the best anomaly detection system for a given dataset):

- **Explainability.** Many algorithms, especially those based on neural networks, lack transparency and are therefore not easily understandable to end-users. We propose the following research questions and make corresponding contributions to answer these questions:
  - *Q2.1*: How to achieve intrinsic explainability of anomaly detection systems?
  - *Q2.2*: Are post-doc explanations for Graph Neural Networks robust to adversarial attacks?
  - *Answer to Q2.1* (Contribution 3): *A Survey on Explainable Anomaly Detection* [46], on which Chapter 1 (namely this chapter) is partially based. Specifically, this work provides a comprehensive and structured survey on state-of-the-art explainable anomaly detection techniques. We propose a taxonomy based on the main aspects that characterize each explainable anomaly detection technique, aiming to help practitioners and researchers find the explainable anomaly detection method that best suits their needs.
  - *Answer to Q2.1* (Contribution 4): *Explainable Contextual Anomaly Detection Using Quantile Regression Forests* [57], which will be presented in Chapter 4. In this chapter, we develop connections between

dependency-based traditional anomaly detection methods and contextual anomaly detection methods. Based on resulting insights, we propose a novel approach to inherently interpretable contextual anomaly detection that uses Quantile Regression Forests to model dependencies between features.

  - *Answer to Q2.2* (Contribution 5): *Explainable Graph Neural Networks under Fire* [58], which will be presented in Chapter 5. In this chapter, we demonstrate that post-hoc Graph Neural Networks (GNNs) explanations cannot be trusted, as common GNN explanation methods turn out to be highly susceptible to adversarial perturbations. That is, even small perturbations of the original graph structure that preserve the model's predictions may yield drastically different explanations. This calls into question the trustworthiness and practical utility of post-hoc explanation methods for GNNs. To be able to attack GNN explanation models, we devise a novel attack method dubbed *GXAttack*, the first *optimization-based* adversarial white-box attack method for post-hoc GNN explanations under such settings.

- **Generalizability.** Algorithms developed based on specific datasets suffer from performance degradation when deployed on new datasets that are generated by new but similar mechanisms (e.g., due to robot, or software upgrade). We propose the following research question and make corresponding contributions to answer it:
  - *Q2.3*: How to detect graph-level anomalies in an unseen target domain with the help of labeled normal graphs from a different but related source domain?
  - *Answer to Q2.3* (Contribution 6): *Cross-domain Graph Level Anomaly Detection* [59], which will be presented in Chapter 6. In this chapter, we propose a *cross-domain graph level anomaly detection method*, aiming to identify anomalous graphs from a set of unlabeled graphs (*target domain*) by using easily accessible normal graphs from a different but related domain (*source domain*). Our method consists of four components: a feature extractor that preserves semantic and topological information of individual graphs while incorporating the distance between different graphs; an adversarial domain classifier to make graph level representations domain-invariant; a one-class classifier to exploit label

information in the source domain; and a class aligner to align classes from both domains based on pseudolabels.

- **Automatability**. Due to the lack of labels, it is challenging to perform model selection (e.g., hyper-parameter optimization) for unsupervised anomaly detection algorithms. This is a critical part of achieving an automated anomaly detection system. We propose the following research question and make corresponding contributions to answer it:
  - *Q2.4*: How to automatically tune hyperparamaters in anomaly detection systems without relying on labels?
  - *Answer to Q2.4* (Contribution 7): *Towards Automated Self-Supervised Learning for Truly Unsupervised Graph Anomaly Detection*, which will be presented in Chapter 7. In this chapter, we empirically demonstrate that three important factors can substantially impact detection performance of self-supervised learning (SSL) based graph anomaly detection methods across datasets: 1) the specific SSL strategy employed; 2) the tuning of the strategy's hyperparameters; and 3) the allocation of combination weights when using multiple strategies. Most SSL-based graph anomaly detection methods circumvent these issues by arbitrarily or selectively choosing SSL strategies, hyperparameter settings, and combination weights. To mitigate this issue, we propose to use an internal evaluation strategy (with theoretical analysis) to select hyperparameters in SSL for unsupervised anomaly detection.

## 1.5 Outline of This Dissertation

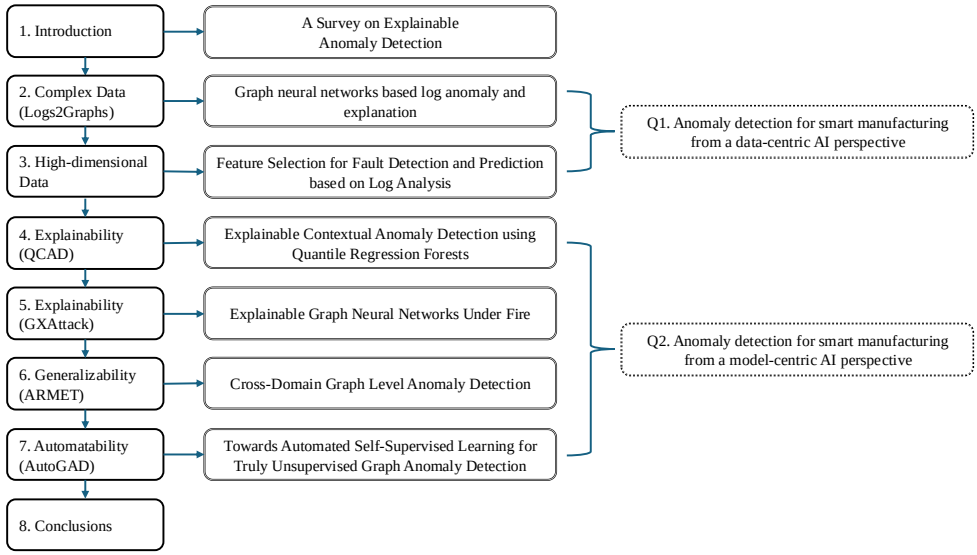


**Figure 1.3:** Organization of this dissertation

As shown in Figure 1.3, this *Introduction* chapter provides necessary background information for understanding this dissertation, including the introduction of Smart Manufacturing, Trustworthy Anomaly Detection, Model-Centric AI and Data-Centric AI. Moreover, we also present the research questions and contributions, and the outline of this dissertation in this chapter.

Chapters 2 through 7 contain the research papers as published or submitted, where Chapters 2 and 3 aim to answer research question Q1 (including sub-questions *Q1.1* and *Q1.2*) and Chapters 4, 5, 6, and 7 attempt to answer research question Q2 (including sub-questions *Q2.1*, *Q2.2*, *Q2.3* and *Q2.4*).

Chapter 8 –*Conclusions and Future Directions* concludes the dissertation by summarizing the findings and answers to research questions from Chapters 2 to 7. Moreover, we also discuss the limitations of current work, point out challenges and outline possible future directions.

## 1.6 List of Publications

The chapters in this dissertation are based on the following publications and manuscripts. Chapter 1 is partially based on publication [46], while the remaining chapters use the publications without any changes to their content, edited only for style cohesion.

| Chapter | Publication |
|---|---|
| 1 | **Zhong Li**, Yuxuan Zhu, & Matthijs van Leeuwen (2024). *A Survey on Explainable Anomaly Detection.* ACM Transactions on Knowledge Discovery from Data 18(1), 1-54. [46] |
| 2 | **Zhong Li**, Jiayang Shi, & Matthijs van Leeuwen (2024). *Graph Neural Networks based Log Anomaly Detection and Explanation.* In Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings, 306-307. (Extended version submitted to Engineering Applications of Artificial Intelligence) [55] |
| 3 | **Zhong Li** & Matthijs van Leeuwen (2022). *Feature Selection for Fault Detection and Prediction based on Log Analysis.* SIGKDD Explorations Newsletter 24(2), 96–104. [56] |
| 4 | **Zhong Li** & Matthijs van Leeuwen (2023). *Explainable Contextual Anomaly Detection using Quantile Regression Forests.* Data Mining and Knowledge Discovery 37(6), 2517-2563. [57] |
| 5 | **Zhong Li**, Simon Geisler, Yuhang Wang, Stephan Günnemann, & Matthijs van Leeuwen (2024). *Explainable Graph Neural Networks Under Fire.* Manuscript submitted to IEEE Transactions on Knowledge and Data Engineering. |
| 6 | **Zhong Li**, Sheng Liang, Jiayang Shi, & Matthijs van Leeuwen (2024). *Cross-Domain Graph Level Anomaly Detection.* IEEE Transactions on Knowledge and Data Engineering 36(12), 7839–7850. [59] |
| 7 | **Zhong Li**, Yuhang Wang, & Matthijs van Leeuwen (2024). *Towards Automated Self-Supervised Learning for Truly Unsupervised Graph Anomaly Detection.* Manuscript submitted to Data Mining and Knowledge Discovery. |

Particularly, the following publications or manuscripts are not used in this dissertation.

| Chapter | Publication |
|---|---|
| — | **Zhong Li**, Matteo Quartagno, Stefan Böhringer, & Nan van Geloven (2022). *Choosing and changing the analysis scale in non-inferiority trials with a binary outcome.* Clinical Trials, 19(1), 14-21. [60] |
| — | Yuhang Wang*, **Zhong Li***, Shujian Yu, & Matthijs van Leeuwen (2024). *Labels Are Not All You Need: Evaluating Node Embedding Quality without Relying on Labels.* Manuscript submitted to NeurIPS 2025. ('*' means equal contributions) [61] |
| — | **Zhong Li**, Qi Huang*, Lincen Yang*, Jiayang Shi, Zhao Yang, Niki van Stein, Thomas Bäck, & Matthijs van Leeuwen. *Diffusion Models for Tabular Data: Challenges, Current Progress, and Future Directions.* Manuscript submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence. ('*' means equal contributions) [62] |