Artificial Intelligence to enable Smart Prognostics and Health Management of Manufacturing Systems for Industry 4.0

A Dissertation Presented

By

Sarvesh Sundaram

to

The Department of Mechanical and Industrial Engineering

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the field of

Industrial Engineering

Northeastern University

Boston, Massachusetts

August 2023

To my wife, Emily, and my parents, Meena and T.G. Sundaram.

Contents

List of Figures

List of Tables

List of Acronyms

AI	Artificial Intelligence
ATA	Air Transport Association
CMMS	Computerized Maintenance Management System
CNC	Computer Numeric Control
CNN	Convolutional Neural Network
CPPS	Cyber-Physical Production Systems
CPS	Cyber-Physical Systems
CRISP-DM	The Cross-Industry Standard Process for Data Mining
CV	Computer Vision
DL	Deep Learning
EITBOK	Enterprise Information Technology Body of Knowledge
ERP	Enterprise Resource Planning
FAA	Federal Aviation Administration
GDP	Gross Domestic Product
IIC	Industrial Internet Consortium
IIoT	Industrial Internet of Things
IIRA	The Industrial Internet Reference Architecture
IoT	Internet of Things
JASC	Joint Aircraft System/Component
LDA	Latent Dirichlet Allocation
MES	Manufacturing Execution System
ML	Machine Learning
MWO	Maintenance Work Order
NER	Named Entity Recognition
NIST	National Institute of Standards and Technology
NLP	Natural Language Processing
OPC-UA	Open Platform Communications Unified Architecture

PHM	Prognostics and Health Management
PoF	Physics of Failure
POS	Part-of-Speech
QC	Quality Control
QI	Quality Inspection
QM	Quality Management
RAMI 4.0	Reference Architecture Model for Industry 4.0
RUL	Remaining Useful Life
SME	Small and Medium-sized Enterprise
SPHM	Smart Prognostics and Health Management
SQI	Smart Quality Inspection
TF-IDF	Term Frequency - Inverse Document Frequency
TLP	Technical Language Processing
TQM	Total Quality Management

Acknowledgements

First of all, I am deeply grateful to Dr. Abe Zeid for providing me with the opportunity to pursue a Ph.D. under his mentorship. I am honored to have you as an advisor, and you have helped me achieve my professional and personal goals. I would also like to thank Dr. Mohsen Moghaddam, and Dr. Tucker J. Marion, who have both supported me with their valuable guidance and feedback throughout this process. I am very grateful to you, and your advice has helped me in becoming a better researcher.

Lastly, I would like to thank my family. I would not be here without their love, support, and encouragement. My wife, Emily, has been a constant pillar of strength for me. I thank you for your patience and kindness, you have helped me achieve my dream. My parents, Meena, and T.G. Sundaram, have made many sacrifices to help me get to this point in life. I would not be here if not for your love and efforts.

Abstract of the Dissertation

Artificial Intelligence to enable Smart Prognostics and Health Management of Manufacturing Systems for Industry 4.0

by

Sarvesh Sundaram

Doctor of Philosophy in Industrial Engineering

Northeastern University, August 2023

Dr. Ibrahim Zeid, Advisor

Dr. Mohsen Moghaddam, Committee Member

Dr. Tucker Marion, Committee Member

Modern manufacturing systems are complex and heterogeneous distributed information systems that consist of manufacturing hardware and networking software. Unlike traditional manufacturing environments, modern ones generate large amounts of data in real-time that take the form of sensor measurements from the shopfloor, images from the product inspection, technical text from maintenance records, and many more. The issue of collecting, processing, and analyzing this data is at the core of the Industry 4.0 paradigm's success. What is more important, challenging, and rewarding is that such valuable real-time streaming data can be used for diagnostics, prognostics, and health monitoring of systems and products. This dissertation explores the area of health monitoring of modern manufacturing environments to support the state-of-the-art Industry 4.0 paradigms. We deploy Machine Learning (ML), Deep Learning (DL), and Natural Language Processing (NLP) based approaches to assess the performance of systems and products by identifying anomalies and defects, estimating the Remaining Useful Life (RUL), and determining what sort of corrective maintenance actions may be necessary.

In the context of the foregoing, advanced manufacturing environments are very intricate and have a vast array of complex problems that can be researched. This dissertation focuses on four important questions: (1) what are the requirements to achieve interoperability in smart manufacturing? (2) how can the next hardware failure, be it of a machine tool or a cutting tool, be predicted before it occurs? (3) how can we identify the products that fail inspection, and how can we improve the overall quality inspection process by considering the factors affecting it? and (4) how can we leverage the resource-rich maintenance logs from manufacturing systems to find the most frequently occurring

problems, and more importantly, how can we predict corrective actions for future maintenance problems?

This dissertation addresses the above problems by making the following research contributions. In the first work, we discuss interoperability in manufacturing by reviewing semantic, syntactic, factory, and cloud-based interoperability. We then identify key research challenges that impact the implementation of interoperable manufacturing, and review frameworks for modern manufacturing such as Reference Architecture Model for Industry 4.0 (RAMI 4.0), Industrial Internet Reference Architecture (IIRA), and several others. The second part of the research proposes a framework for Smart Prognostics and Health Management (SPHM) to support manufacturing system health. We apply it to a milling machine operation and assess the condition of the milling machine's tool tip by identifying whether it is safe or degraded and estimate its RUL using ML models. To address the third question, we introduce a methodology for Smart Quality Inspection (SQI) of manufacturing products by using a custom Convolutional Neural Network (CNN) architecture. A shopfloor computer application is developed that helps minimize the effects of the factors affecting the visual inspection process. The fourth component of this dissertation introduces a Technical Language Processing (TLP) framework to deal with the unstructured and complex nature of technical text data from maintenance work orders. We demonstrate its capabilities by applying it to maintenance logs of aircraft to predict corrective actions, and link maintenance problems to standardized maintenance codes as prescribed by the industry.

The rapid growth of Artificial Intelligence (AI) suggests that its potential to transform health monitoring capabilities is going to improve in the coming years. Future research in health monitoring of manufacturing systems and products can be aimed towards the following areas: development of standardized communication protocols for interoperability, further automation of the inspection process by integrating localized fault detection, and incorporation of TLP with predictive maintenance approaches and quality control to form a prescriptive approach, to suggest a few.

Chapter 1
Introduction

1.1 Research Motivation

The manufacturing industry has always been driven by the evolution of technology—from steam engines to electricity, microprocessors, computers, automation, and recently, Internet of Things (IoT) and Cyber–Physical Systems (CPS). In the past few decades, strategies such as Total Quality Management (TQM), Six Sigma, Lean, and Zero-Defect Manufacturing have improved product quality and promoted customer satisfaction. More recently, several initiatives have been launched globally (e.g., Industry 4.0 in Europe [1]; Made in China 2025 [2]; U.S. Advanced Manufacturing [3]) to address the needs for higher efficiency, lower costs, and mass-personalization of products and services. At the same time, the increasing global demand for consumer products has placed an endless pressure on manufacturers, necessitating quicker turnaround times on the shopfloors. Recent strains on global supply chains caused by the Covid-19 pandemic place an even greater emphasis on the importance of minimizing down-times to make up for time lost due to interruptions elsewhere. This highlights the growing significance of health management of manufacturing systems.

Almost every system and product that is in existence today requires some form of maintenance. The needs of a growing population can be directly associated with the increase in consumption of goods and services, requiring the maintenance of more systems and equipment. Many industries are operating at their maximum level of capacity utilization, resulting in reduced maintenance times for machinery and equipment [4]. Studies have shown that maintenance can take up a large fraction of an organization's operational budget [5], with manufacturers in the United States spending approximately $50 billion on maintenance related costs [6]. Monte-Carlo estimations suggest that annual costs concerning overall health management amount to more than $222 Billion in the

United States [7], and recalls due to faulty goods result in losses of more than $7 Billion each year [8]. In recent years, maintenance engineering has emerged as one of the most important areas in manufacturing organizational planning as opposed to it being purely associated with production operations [9]. It is becoming evident that maintenance should not be viewed solely as an isolated function, but instead as a competitive strategy [10]. While maintenance of manufacturing systems is crucial, an equally important area is that of product health monitoring, which reduces consumer's risk. The cost of poor quality can be as much as 30% of the gross sales in manufacturing [11]. Companies with poor quality practices tend to spend an average of $350 million more per year compared to companies that maintain high quality standards [12]. Additionally, the cost of poor product quality can result in other effects such as damage to brand value, loss of customer trust, raw material wastage, and losses incurred due to recall, rework, and scrap [13].

Manufacturing enterprises implement maintenance strategies based on production requirements, the complexity of machinery and equipment, and the costs involved. There are several approaches to maintenance that have been adopted across the industry. Unplanned/reactive maintenance, also known as run-to-failure maintenance, is one of the traditional maintenance strategies in which the machine or component is allowed to fail before any corrective action is taken. This strategy can be expensive due to the high costs associated with repairing and replacing the machines or components, and additional losses associated with unplanned system shutdown [14]. Another widely used approach is preventive maintenance in which the system is inspected at regularly scheduled intervals to identify any potential issues that might arise. In most cases, preventive maintenance is planned well in advance to be implemented at a set-time in the future and in order to minimize failure probability of a specific system/equipment [15]. A more recently introduced maintenance strategy that is growing in popularity is predictive maintenance [16]. It is a data-driven approach in which parameters concerning the health of the system are used to monitor the condition of the equipment and predict when it is likely to fail. Predictive maintenance falls under the umbrella of Prognostics and Health Management (PHM). PHM is an interdisciplinary area of engineering that deals with the monitoring of system health, detecting failures, diagnosing the cause of failures, and making a prognosis of component and system level failures by using metrics such as Remaining Useful Life

(RUL). PHM technologies are being widely incorporated into the modern manufacturing approaches as an in-situ evaluation of the system is made possible.

The health of manufacturing products, on the other hand, is monitored using Quality Control (QC) and Quality Inspection (QI) methods. These domains aim to ensure that products adhere to predetermined quality standards. Quality inspection is a planned and organized process in which the state of the product is assessed by examination, measurement, testing, gauging, or comparison to determine if it conforms to desired specifications [17]. In most cases, quality inspection involves a human operator that inspects the product to ascertain its conformity. However, the accuracy and reliability of the inspection are often unsatisfactory. According to Harris [18], as the complexity of the product increases, the accuracy of inspection conducted by operators decreases. Similarly, in a study conducted by the Sandia National Labs [19], the accuracy of correctly rejecting precision manufactured parts by human operators was found to be 85% while the industry average was 80%. Another recent study concluded that operator errors accounted for 23% of the inaccuracies in quality control in the oil and gas industry [20]. On this front too, the need for increased yield results in greater scrutiny on the improvement of product quality.

Industries that account for the majority of the global Gross Domestic Product (GDP) [21] rely largely on reactive or preventive maintenance approaches [22]. In overall Quality Management (QM), manufacturers face challenges due to the following reasons: there is a lack of information about the importance of QM, they face resource constraints, and there is an inadequacy in standards [23], [24]. Therefore, any improvements made to health management practices will result in a tremendous amount of savings across manufacturing.

Artificial Intelligence (AI), with development of novel Machine Learning (ML), Deep Learning (DL), and Natural Language Processing (NLP) algorithms allows the automation of many tasks. Coupled with the advancements in low-cost sensor technology powered by Internet of Things (IoT), there is a scope to automate and implement effective decision-making. According to a study conducted by PwC, 24% of assembly and quality testing operations involve some form of AI, and 20% of maintenance operations incorporate AI-based technologies [25]. Another report by McKinsey estimates that the integration of AI into factories will generate \$3.7 billion of value by 2025 [26]. A survey of over 1500 executives across fifteen countries worldwide concludes that 89% of the manufacturing companies have plans to incorporate AI into the workflow, and ones that already adopted AI have saved over 14% in manufacturing costs [27]. The transformative role of AI in the manufacturing landscape is quite clear.

This dissertation will provide an understanding of current manufacturing paradigms and the requirements for modern manufacturing and health management, develop detailed frameworks and methodologies for AI-based Smart Prognostics and Health Management, and demonstrate their capabilities to support manufacturing system health, product health, and maintenance operations.

1.2 Research Problem

Modern manufacturing systems are interconnected and have moved on from a hierarchy-based decision-making architecture to a network-free, decentralized decision-making one [28]. The heterogeneity in systems and processes results in the issue of integration in terms of communication standards and protocols, known as interoperability. There are several challenges that manufacturing organizations face in implementing interoperability [29] – transfer of data between systems, compatibility in software, complications due to varying terminologies, non-standardized documentation, difficulty in testing the connected environments. To better understand the requirements of modern manufacturing, we review the different approaches to interoperability and some of the important paradigms that enable interoperable manufacturing.

Upon understanding the needs of modern manufacturing, we establish health monitoring in manufacturing as the area of our focus. We identify two main facets of health monitoring – system, and product health monitoring. Based on these areas, we pose an important question: **How can the system, and product health monitoring be enhanced by AI, and in what capacity?** To address this, we consider data from various sources in manufacturing - numerical sensor measurements from machines, product images from quality inspection, and technical text data from Maintenance Work Orders (MWOs). Figure

1 shows the how Machine Learning, Deep Learning, and Technical Language Processing can be used in monitoring the health of a manufacturing environment.

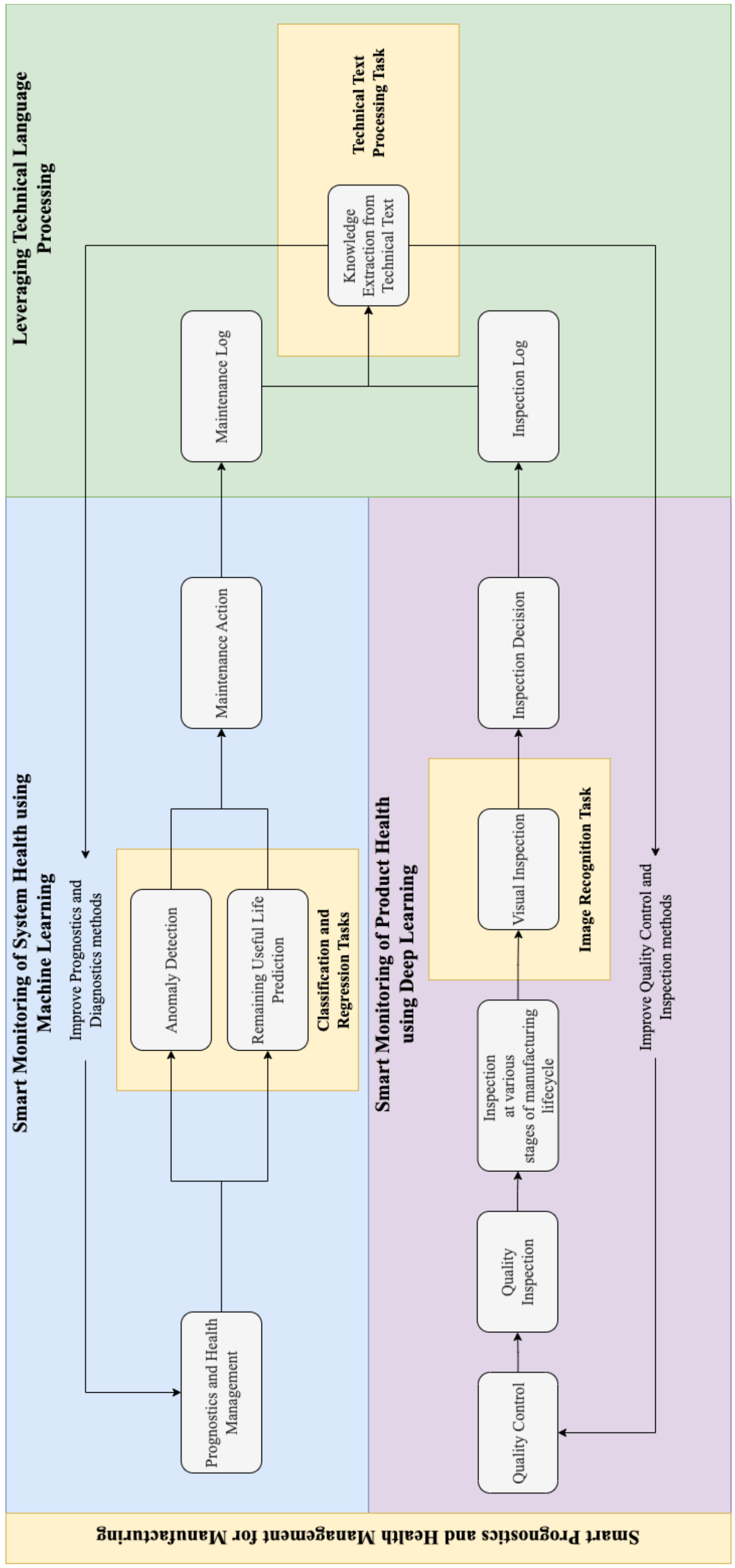

Figure 1. Artificial Intelligence to enable Smart Prognostics and Health Management of Manufacturing Systems for Industry 4.0

In health monitoring of manufacturing systems, the attention of many studies has been toward the implementation of ML and DL algorithms in identifying RUL of machinery and equipment. Most works aim to test the predictive power of data-driven models, and very few enumerate all the steps taken required to implement PHM methods in manufacturing. Traini et al. [30] develop a framework to address predictive maintenance in milling based on a generalized methodology. Lei et al. [31] review the stages in Condition-Based Maintenance (CBM), from data acquisition to RUL estimation. Mohanraj et al. [32] review the steps in condition monitoring from the perspective of a milling operation. A framework for PHM in manufacturing with cost–benefit analysis is developed by Shin et al. [33]. While these studies provide noteworthy steps to implement prognostics models in the industry, the in-depth understanding of all the steps involved in the monitoring of manufacturing system health is often overlooked. **The question that remains is, What are the steps required to implement PHM on the shopfloor – from machine and sensor set up to the identification of prognostics and anomaly detection tasks?** To address this research requirement, we introduce an interoperable framework for SPHM in manufacturing enabled by AI. We demonstrate the capabilities of the SPHM framework by considering a milling machine operation to address two tasks – anomaly detection and RUL prediction, using ML models for classification and regression respectively.

With regards to health monitoring of products in manufacturing, there are several works that implement image-based quality inspection methods for defect detection of manufacturing products. He et al. [34] use a Convolutional Neural network (CNN) for defect detection on product surfaces at the pixel level. A convolutional variational autoencoder is proposed by Yun et al. [35] to study a multi-class surface defect identification problem on metals. In the case of welding products, Sassi et al. [36] apply a transfer learning approach and achieve a good performance on a small dataset. For inspection of casting products, Oborski et al. [37] use a CNN model in a holonic shopfloor based setting. In detecting defects of welded nuts, Lee et al. [38] use a model based on VGG-16. While improving defect detection is the main objective of visual inspection, very few of them take into consideration the various factors affecting the visual inspection process. Even with automation, the inspection process often relies on human inspectors for some critical-decision making steps. An inspection system should attempt to minimize the maximum number of the factors out of the task, environmental, operator, organizational, and social factors, so that there are fewer opportunities for errors to occur. **There is a need for an effective AI-based inspection system that: 1) performs well in its primary objective of defect detection, 2) minimizes the factors affecting the inspection process, and 3) allows the documentation of decisions made**. Considering the above requirements, we propose Smart Quality Inspection—an AI-based approach that uses a custom CNN architecture for the visual inspection process and demonstrate a use case on a benchmark image dataset from a casting process.

Finally, we address a relatively unexplored area that can be used to benefit the monitoring of system and product health – technical text data from Maintenance Work Orders (MWOs). Technical text data in manufacturing are often considered to be 'black-holes', meaning they are a resource rich domain but are rarely used in decision-making**.** A reasonable question that one may ask is, why not use NLP to process technical text? While NLP has been successful at adapting to several domains such as – medicine [39], supply chain [40], finance [41], a key difference between technical and non-technical text is that technical text consist of complex terminologies, technical jargon, and colloquialisms. Non-technical text, on the other hand, is well documented, and there are publicly available corpora or dictionaries that can be used to train language models. Technical text is domain-specific, and each domain is unique in terms of the data it consists of. **To tackle the problem of processing technical text data, there is a need for a text processing system that can 1) understand the complexities of the technical text, 2) preprocess technical jargon, colloquialisms, and terminology without distorting its meaning, and 3) use the data to enhance health monitoring.** We propose a Technical Language Processing (TLP) framework for PHM that addresses the challenges of technical text processing and illustrate then its effectiveness by applying it to MWOs of aircraft. In doing so, we identify two key scenarios that it can be used to address: predicting corrective actions for new maintenance problems and matching new maintenance problems to standardized codes as prescribed by the industry.

1.3 Research Questions

This dissertation aims to address how Artificial Intelligence can transform the Prognostics and Health Management of manufacturing systems and products by exploring the following research questions:

RQ 1. What are the needs of an interoperable manufacturing system, and what are the challenges that affect its implementation?

RQ 2. How can AI be utilized to monitor the health of a manufacturing system, and what are the steps involved in realizing it?

RQ 3. Can AI be used to enhance the performance of the visual inspection process, and can the factors affecting the inspection process be minimized in a user-friendly manner?

RQ 4. How can technical text data from manufacturing and maintenance be leveraged for efficient decision-making?

1.4 Dissertation Structure

The rest of the chapters presented in this dissertation are organized as follows. Chapter 2 introduces the concept of interoperability in modern manufacturing, identifies research challenges in achieving interoperable manufacturing systems, and reviews state-of-the art paradigms that help in realizing an interoperable shopfloor. Chapter 3 proposes a framework to achieve ML-based health monitoring of manufacturing systems and applies it to a milling machine operation for anomaly detection and RUL estimation. Chapter 4 addresses the role of AI in product health monitoring by proposing a Smart Quality Inspection application using a custom CNN architecture and applies it for visual inspection of casting products. Chapter 5 develops a framework for Technical Language Processing (TLP) that tackles the challenges of technical text processing and deploys it to aircraft MWOs by addressing two scenarios – prediction of corrective actions for new maintenance problems and matching new maintenance problems to standardized maintenance codes. Chapter 6 provides a summary of the dissertation and suggests directions for future work.

Chapter 2

SPHM Requirements to Support Industry 4.0 Paradigm

This chapter reviews the notion of interoperability in the context of modern manufacturing environments. First, an overview of the emerging concepts and paradigms in manufacturing is provided, along with their detailed characteristics and elements, comparative analysis, and common vernacular. Then, different definitions and dimensions of interoperability are introduced, and the interoperability problems associated with vertical and horizontal integration are differentiated and formalized with respect to the interoperability stack model. Finally, several reference architectures for smart manufacturing and their approaches to interoperability are evaluated and compared, followed by a summary of R&D challenges.

2.1 Introduction

Rapid industrialization has seen a transformation in the manufacturing industry over the past century. Emerging technologies such as Artificial Intelligence (AI), Cyber-Physical Systems (CPS), and Internet of Things (IoT) are reducing manufacturing costs while improving yield. The synergetic effect between emerging technologies and needs has led to the creation of new manufacturing paradigms characterized by: (1) digitalization and integration of manufacturing resources on cloud-based platforms as adaptive, secure, and on-demand services, and (2) smart and connected objects capable of real-time and autonomous decision-making via embedded electronics and analytical/cognitive capabilities (see Figure 2). Examples of recent paradigms include smart manufacturing [42], cyber–physical production systems (CPPS) [28], Industry 4.0 [1], and cloud manufacturing [43], among others. We henceforth refer to these new paradigms as smart manufacturing.

Smart manufacturing involves networking of heterogenous components and services that reside within the boundaries of a factory (e.g., integration of smart shop–floor devices) or beyond (e.g., integration of a manufacturing cell with a cloud-based service). These two types of integration are sometimes referred to as vertical and horizontal, respectively [1]. Thus, unlike the traditional "automation pyramid" for manufacturing control where integration problems would arise in intra-enterprise hierarchical structures (i.e., Enterprise Resource Planning (ERP)–Manufacturing Execution Systems (MES)–shopfloor), smart manufacturing calls for the integration of diverse and distributed cloud-based services, enterprises, smart factories, smart devices, and processes. The required integration in turn calls for a seamless exchange of information between these heterogenous systems which operate under a wide variety of communication standards [44]. This phenomenon has introduced an unprecedented challenge known as interoperability.

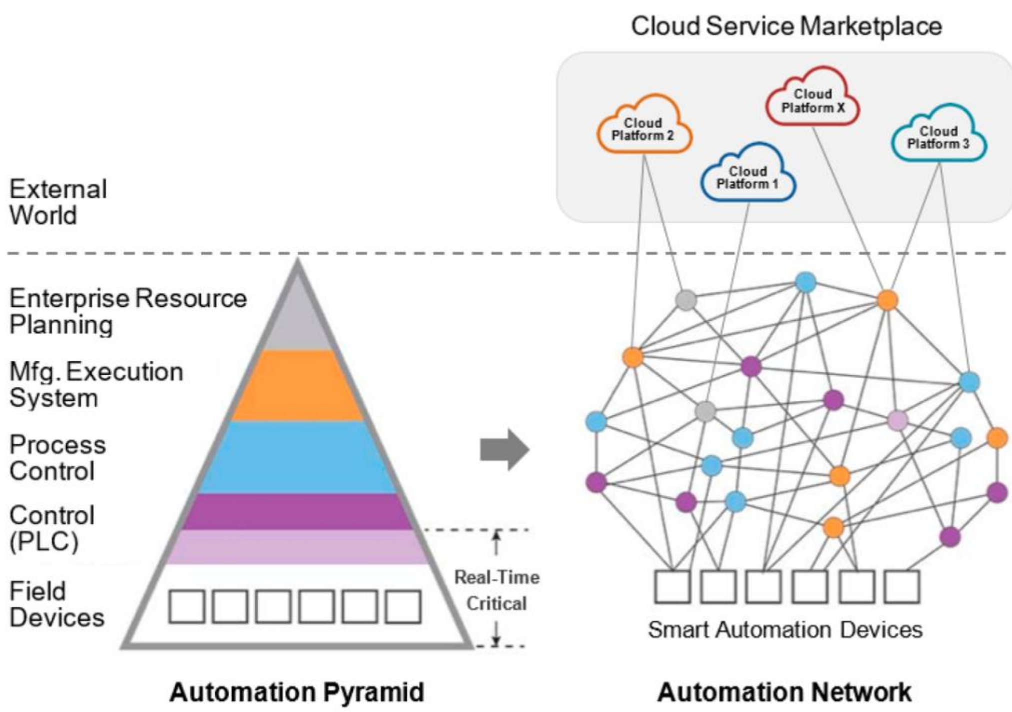

Figure 2. Evolution of the hierarchical model of enterprise control system integration toward an integrated network of smart automation devices, cloud services, cloud platforms, and enterprises. Different types of interoperability problems may arise, due to the diverse nature of interactions in the emerging automation networks.

(Figure adapted from [45])

2.2 Overview of Key Concepts

To better understand and formalize different definitions and dimensions of interoperability, it is necessary to provide an overview of its enabling concepts and technologies (e.g., cyber–physical systems; Internet of things; data analytics).

2.2.1 Cyber-Physical Systems

Cyber–physical systems (CPS) are systems that integrate physical processes with computation through embedded computers, feedback loops, and computer networks used to control the desired process [28]. The CPS framework usually consists of a central control unit (e.g., a microcontroller) which controls the sensors and actuators that perform the desired task and generate data [46]. CPS, as designed, consists of interacting elements with input and output layers, unlike standalone devices in traditional systems. CPS has been defined by the National Science Foundation (NSF) as "systems that consist of physical and software components that are deeply intertwined, each operating on different spatial and temporal scales, exhibiting multiple and distinct behavioral modalities, and interacting with each other in a myriad of ways that change with context" [47]. With continuous improvements in technologies involving CPS, there has been an exponential growth of its applications in the manufacturing industry.

2.2.2 Internet of Things

The Internet of things (IoT) enables access to a full-fledged network of smart devices, systems and products, cloud-based services, humans, and enterprises [48], which are connected to each other over a secure Internet. The devices can be smart devices that consist of sensors and actuators, or legacy devices that record some form of information that can be shared on the cloud. The IoT provides the capability to connect legacy devices and systems to the Internet, which in turn allows for systematic data collection and decision-making. In the industrial setting, the IoT has aided in the growth of smart manufacturing through integration of machines, processes, and systems. IoT is bringing about fundamental transformations to the manufacturing and supply networks through the integration of physical manufacturing processes with cloud-based services [49]. A service, in this context, refers to "a logical representation of a set of activities that has specified outcomes, is self-contained, may be composed of other services and is a black box to consumers of the service" [50]. The encapsulation of knowledge and capabilities in cloud-based services allow users to shift their focus from reinventing those capabilities to utilizing them. Examples of cloud-based services in the context of smart manufacturing include analytics, security, visualization, machine learning, and cognitive services [51], sourced from a variety of platforms, such as Amazon Web Services, Microsoft Azure, and IBM Cloud, among others.

2.2.3 Data Analytics and Machine Learning

The integration of distributed and formerly-siloed devices, systems, and enterprises through CPS and the IoT has led to the generation of massive data—from the data generated by sensors-actuators and automation devices on the shopfloor to the transaction data generated by cloud marketplaces and the life cycle data generated by smart products. These data revolution demands new techniques and technologies for efficient collection, storage, retrieval, communication, and real-time analysis of large and diverse datasets. On the other hand, the fast-paced growth of machine learning algorithms and compute capabilities has provided unprecedented potentials to support manufacturing operations and decisions with further insights from their available datasets. Examples of such capabilities include, but are not limited to, processing high-dimensional data, reducing the complexity of data, adapting to new processes and environments, identifying relations, correlations, and causality [52]. In the manufacturing setting, data analytics and machine learning techniques have been extensively applied in various areas, such as quality management, preventive maintenance, fault diagnosis and prognosis, decision-making, sequencing and production scheduling [53].

2.3 Emerging Paradigms in Manufacturing

Over the past decade, several new paradigms have emerged to represent the various requirements and characteristics of next-generation manufacturing systems. Some common examples are introduced below (see Reference [45] for further details). The common theme of all of these paradigms, as pointed out earlier, revolves around service-orientation through digitalization and integration of resources on the cloud (i.e., horizontal integration) as well as smart reconfigurable shop-floor operations through integrated and collaborative automation devices and systems (i.e., vertical integration).

2.3.1 Smart Manufacturing

The U.S. National Institute of Standards and Technology defines smart manufacturing as “fully-integrated, collaborative manufacturing systems that respond in real time to meet changing demands and conditions in the factory, in the supply network, and in customer needs” [42]. The Smart Manufacturing Leadership Coalition provides a complementary definition for smart manufacturing as “the ability to solve existing and future problems via an open infrastructure that allows solutions to be implemented at the speed of business while creating advantaged value”. The characteristics of smart manufacturing include digitalization and service-orientation, smart and connected automation devices, collaborative supply networks enabled by advanced analytics [42], with the goal of enabling cost-efficient, flexible, and personalized mass-production [54].

2.3.2 Cyber–Physical Production Systems

Cyber–physical production systems (CPPS) consist of automated components that communicate with each other across multiple levels of a manufacturing or production facility, including the shop-floor, warehouse and logistics networks [28]. The characteristics of CPPS include intelligent and autonomous interactions of CPS objects, responsiveness to changes, integration of manufacturing resources on the cloud as adaptive and secure services [55], predictive and prescriptive operations through data analytics, and product life cycle support through digital twin technologies [56].

2.3.3 Industry 4.0

Vertical integration of automation devices, horizontal integration of inter-factory operations throughout value networks, and end-to-end engineering of smart products are the three major characteristics of Industry 4.0 [1]. This paradigm represents a model of a fully integrated industry where “products and services are flexibly connected via the Internet or other network applications like the blockchain. The digital connectivity enables an automated and self-optimized production of goods and services including the delivering without human interventions. The value networks are controlled decentralized while system elements are making autonomous decisions” [57].

2.3.4 Cloud Manufacturing

Cloud manufacturing is defined as “a new networked manufacturing paradigm that organizes manufacturing resources over networks (manufacturing clouds) according to consumers’ needs and demand to provide a variety of on-demand manufacturing services via networks (e.g., the Internet) and cloud manufacturing service platforms” [43]. A derivative of this paradigm is cloud-based design and manufacturing, “a service-oriented networked product development model in which service consumers are enabled to configure, select, and utilize customized product realization resources and services ranging from Computer Aided Engineering (CAE) software to reconfigurable manufacturing systems” [58]. Another derivative of cloud manufacturing is social manufacturing, “a new cyber-physical-social-connected and service-oriented manufacturing paradigm that drives distributed Production Service Providers (PSPs) to self-organize into dynamic resource communities (DRCs) through social network, provide the production- and product-related services to prosumers [producers + consumers], and collaborate with prosumers through cyber-physical-social systems/network (CPSS) [59]. Accordingly, the main theme of the cloud manufacturing paradigm and its derivatives is horizontal integration through service-orientation with emphasis on design and manufacturing services.

2.4 Interoperability in Manufacturing

The integration and networking of smart manufacturing components and services within and beyond the boundaries of the factory, as characterized by the aforementioned paradigms, call for seamless exchanging of information with syntax and semantics understandable by all the heterogeneous systems involved. This ability is referred to as interoperability. In shopfloor operations, for example, manufacturing equipment must update each other and the control systems about the process updates and potential errors or conflicts. They also need to interact with cloud-based platforms to receive required services. In general, the software that these components and services are based on must be able to interact by developing an understanding of the common language or syntax that each uses to communicate with the external world.

2.4.1 Definitions of Interoperability

Several definitions of interoperability have been proposed by different standard organizations and institutions. In general terms, interoperability can be defined as the ability of two or more entities to interact and cooperate. ISO 16100 defines interoperability as “the ability to share and exchange information using a common syntax and semantics to meet an application specific functional relationship across a common interface” [60]. The IEEE Standard Computer Dictionary defines interoperability as “the ability of two or more systems or components to exchange information and to use the information that has been exchanged” [61]. There are many other definitions of interoperability that are used in different contexts. The most relevant ones have been compiled and listed below.

- The ability of one system to receive and process intelligible information of mutual interest transmitted by another system [62].

- Interoperability means the ability of two or more parties, machine or human, to make a perfect exchange of content. Perfect means no perceptible distortions or unintended delays between content origin, processing, and use [63].

- Interoperability among components of large-scale, distributed systems is the ability to exchange services and data with one another [64].

- The condition achieved between systems when information or services are exchanged directly and satisfactorily between the systems and/or their users [65].

- The ability to integrate data, functionality, and processes with respect to their semantics [66].

- Interoperability, in a broad sense, refers to the use of computer-based tools that facilitate coordination of work and information flow across organizational boundaries, focusing mainly on inter-enterprise distributed business plans and flows [67].

- The ability of systems, units, or forces to provide services to and accept services from other systems, units, or forces and to use the services so exchanged to enable them to operate effectively together [68].

In smart manufacturing, interoperability takes two general forms. The first form corresponds with vertical integration, e.g., interoperability between the manufacturing software, the shop–floor departments, the processes performed by different equipment, the various shop–floor systems, and so forth [69]. The second form corresponds with horizontal integration; the interoperability between smart automation devices, cloud services, cloud platforms, and enterprises (see Figure 2). Successful implementation of enterprise-wide interoperability would result in effective and smooth operations of the manufacturing industry, thus cutting costs, increasing production and product quality.

2.4.2 Challenges in Implementation of Interoperability

The factors that affect interoperability are bound to be multivariate considering the complexity of the processes. The Manufacturing Interoperability Program at NIST (the National Institute of Standards and Technology) list several factors that impact the effectiveness of interoperability [29]:

- Transfer of data between systems that may be similar or dissimilar (commercially).
- Transfer of data between software made by the same vendor (or creator) but having different versions on the systems.
- Compatibility between different versions of software (newer and older versions).
- Misinterpretation of terminology used or in the understanding of the terminology used for exchange of data or information.
- The use of non-standardized documentation on which the exchange of data is processed or formatted.

- Not testing the applications that are deemed conformant, due to the lack of means to do so between systems.

Other barriers to interoperability include inconsistent data formats or standards, connectivity in the IoT realm, and the wide variety of commercially available products.

2.4.3 Generic Approaches to Implement Interoperability

Over the years, there have been many approaches to successfully implement interoperability. The IEEE Guide to Enterprise IT Body of Knowledge (EITBOK) has categorized the approaches mainly into two types: Syntactic and semantic [70].

2.4.3.1 Syntactic Interoperability

The European Telecommunications Standards Institute (ETSI) defines syntactic interoperability as follows: "Syntactical Interoperability is usually associated with data formats. Certainly, the messages transferred by communication protocols need to have a well-defined syntax and encoding, even if it is only in the form of bit-tables. However, many protocols carry data or content, and this can be represented using high-level transfer syntaxes such as Hypertext Markup Language (HTML), Extensible Markup Language (XML) or Abstract Syntax Notation One (ASN.1)" [71]. Syntactic interoperability merely considers the format of the data—it does not take into account the meaning of the data to be transferred. Standardization of data formats, as well as the mode of communication can greatly improve this type of interoperability.

2.4.3.2 Semantic Interoperability

The ETSI offers a concrete definition for semantic interoperability as follows: "Semantic Interoperability is usually associated with the meaning of content and concerns the human rather than machine interpretation of the content. Thus, interoperability on this level means that there is a common understanding between people of the meaning of the content (information) being exchanged" [71]. Semantic interoperability not only looks into the meaning of the content, it also applies logic to the fact being transferred and used [72]. XML and Resource Definition Framework (RDF) are widely used as standards for semantic interoperability. Nevertheless, RDF is proven more effective due to providing models extendable to multiple techniques that are represented by unique ontologies [73]. A thorough list of ontologies by the devices that use them has been published by Lelli [74] in which they discuss and compare World Wide Web Consortium (W3C) Semantic Sensor Network, Fiesta-IoT, Smart Energy Aware Systems (SEASD), Machine-to-Machine (M2M) solutions (http://www.onem2m.org), and Schema.org technologies, to name a few.

2.4.4 Factory Interoperability (Vertical Integration)

The term manufacturing interoperability refers to the capability of manufacturing enterprises to exchange information that maybe technical or enterprise related in a coherent manner within and between each other [44]. A recent study conducted on the U.S. automotive industry reports an annual economic loss of $1 billion due to the lack of interoperability throughout the supply network [75], which reveals the significant impact of interoperability on the manufacturing sector in terms of both cost and performance quality. Jones and Ray [44] suggest three approaches to tackle the high cost problem associated with manufacturing interoperability. The first is machine-to-machine solution. The idea behind this approach is making each individual machine interoperable with every other machine it is linked to or integrated with. The challenge is that each of these machines may communicate based on their own manufacturer-specified communication protocol. Achieving interoperability under this scenario requires a thorough understanding of their unique semantics along with translation of their syntax. This approach is clearly not effective and is likely to impose significant cost.

The second approach is industry-wide standardization solution. The idea here is to ensure that all the manufacturing industry service partners follow a single solution. For example, a manufacturing unit may be integrated and interoperable until the period of production. However, if further processes have to be performed on the product in later stages, it may have to move to another plant to complete those operations. In this case, not only is a common protocol required for operations within the manufacturing setup, but there is also a need for another protocol between different manufacturing units. This approach can be extended to various industries. The drawback of this approach, however, is that the manufacturing units used to process a given product may belong to different enterprises with different integration/communication protocols. The third, and the most effective approach is the application of open standards or platforms to achieve interoperability. See Reference [76] for a review of recent developments in the area.

2.4.5 Cloud-Manufacturing Interoperability (Horizontal Integration)

The ISO/IEC 19941 standard [77] specifies a facet-based model to approach both interoperability and portability in cloud computing. Extending the notion of cloud computing to cloud manufacturing [43], it is imperative to generalize the interoperability definitions and approaches to the domain of cloud manufacturing. In addition to syntactic and semantic interoperability, cloud-based platforms must address three additional types of interoperability:

- Transport interoperability. This type makes the transfer and exchange of data interoperable by the use of protocols such as REpresentational State Transfer (REST) [78] over HyperText Transfer Protocol (HTTP), and Message Queuing Telemetry Transport (MQTT) [79].

- Behavioral interoperability. This type selects from a list of anticipated responses when requests are received for our services. By setting conditions on the requests, computer simulations with the help of human responses can anticipate the outcomes of the service request and be ready to offer a solution.

- Policy interoperability. This type ensures that all the systems in the cloud follow and conform to the regulations and policies while interacting within the cloud environment.

In cloud manufacturing, the majority of operations and resources are virtualized on cloud platforms as cloud-based services supported by the IoT, cloud computing, and cyber–physical technologies. In recent years, several cloud-based platforms have emerged to provide such cloud-based manufacturing services. Common platforms include Amazon Web Services (AWS), IBM Watson, Microsoft Azure, Google Cloud Platform, Oracle, and Alibaba, among others. The users of these services can subscribe and use services remotely and on-demand, from multiple providers for their manufacturing enterprise. Database management, big data analytics, scheduling and supply-chain operations are among some of the services provided.

A generic cloud manufacturing architecture consists of five layers (see Figure 3): Application layer, application interface layer, core service layer, virtualization layer, and physical resource layer [80]. The application layer develops applications on the cloud for a user or subscriber. These requests can be specialized in nature relative to the manufacturing enterprise that is integrating with the cloud. The next three layers, the application interface layer, the service layer, and the virtualization layer, work in tandem to ensure that cloud manufacturing services are realized with acceptable knowledge, security, and virtualization. The physical resource layer or the resource layer actually implements the manufacturing service that is offered and can be initiated by the user.

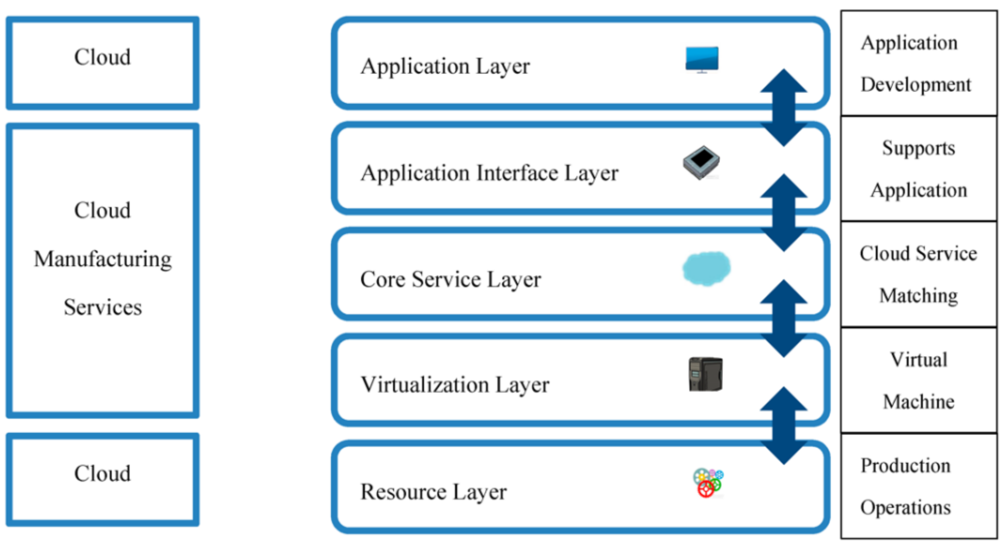

Figure 3. A generic cloud manufacturing architecture.

An example of interoperability in the domain of cloud manufacturing is the Interoperable Cloud-based Manufacturing System (ICMS) [81]. The four layers in the ICMS are the manufacturing resource layer, virtual service layer, global service layer, and application layer. The manufacturing resource layer combines the actual manufacturing systems into chunks that can be invoked by the user by incorporating STEP-NC. The virtual service layer maintains records of the services that are being requested while ensuring safety and privacy. The global service layer ensures that the enterprise has control over the services and also provides means for conducting analysis into the diagnostics of the service offered. The application layer provides an interface to the ICMS that the user or subscriber can access based on privileges granted.

Another example is the Hybrid Manufacturing Cloud (HMC) framework [82]. HMC uses different models of the cloud: private, community, and public. This model prioritizes security and privacy of data and information by the use of a cloud management engine. Li et al. [83] introduce a new approach to cloud architectures by the use of a New Generation Artificial Intelligence-driven Intelligent Manufacturing (NGAIIM) architecture. The authors introduce the applications of artificial intelligence in the current computing habitats and compare its progress over the years. The four-layer architecture proposed consists of a resource layer, communications layer, platform layer and service layer that are integrated with the NGAIIM concept. The resource layer is analogous to the resource layer seen in other architectures, i.e., the base layer consists of the physical elements of the manufacturing unit. The communication layer uses IoT based network models to interconnect the layers, thus providing interoperability. The platform layer provides access for the user to be a part of the system and also provides application support. The service layer provides intelligent services for manufacturing. Chai et al. introduce a platform called INDICS, which is China's Internet-based platform operating on IoT and other technologies [84]. This platform architecture consists of five layers: Resource layer, industrial IoT layer, access layer, cloud layer and the APP layer. The resource layer is again analogous to other architectures. The industrial IoT layer supports and integrates wireless networks and Object Linking and Embedding for Process Control Unified Architecture (OPC-UA) protocol to name a few. The access layer and cloud layer handle the communications and application development with other related tasks respectively. The APP layer provides intelligent services that are supported by the INDICS platform.

Various other frameworks and architectures have been proposed by researchers and industry professionals that are yet to be tested and deployed in the cloud manufacturing environment. With the advancements in the smart manufacturing domain, however, open communication standards and platforms that are shared across by enterprises are starting to attract more research interests.

2.5 Reference Architectures for Interoperable Manufacturing

A number of reference models and architectures have been developed in recent years to address the issues of integration and interoperability in the context of smart manufacturing. Reference architectures have been categorized into different types of architectures [85]:

a) The physical architecture of the components of a system (e.g., automation devices, machines, software, departments).

b) The functional architecture representing the set of functions and processes to be accomplished by the systems.

c) The allocated architecture describing the mapping between the functional architecture and the physical architecture.

The most widely discussed architecture models have been proposed by Platform Industrie 4.0 and Industrial Internet Consortium (IIC), two of the largest organizations that research on topics related to Industry and the Industrial Internet respectively. Platform Industrie 4.0's Reference Architecture Model for Industry 4.0 (RAMI 4.0) is an architecture that was designed primarily for applications in the manufacturing industry, and IIC's Industrial Internet Reference Architecture (IIRA) was designed for all industries related to the Industrial Internet of Things (IIoT) [86]. Other prominent architecture models in this context are IBM's Industry 4.0 architecture and the NIST service-oriented architecture. In the context of this chapter, we consider these architecture models and describe them in the context of manufacturing interoperability.

2.5.1 RAMI 4.0

The RAMI 4.0 was developed by several industry partners based out of Germany. RAMI 4.0 addresses interoperability by proposing a set of abstract interoperability layers for the manufacturing industry, as seen in Figure 4b [1]. Those layers are better explained as follows:

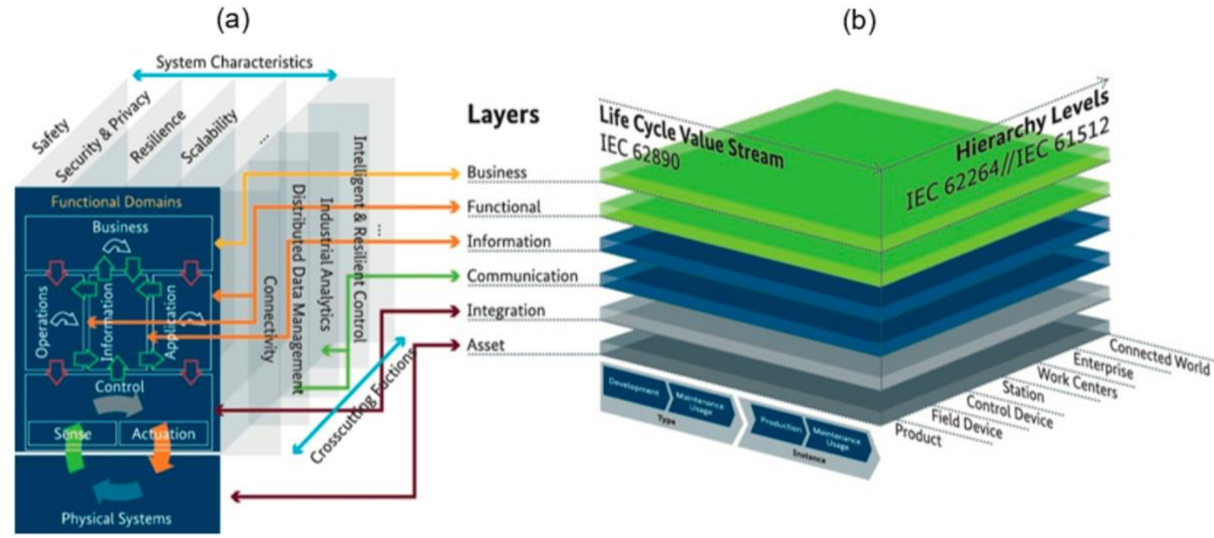

Figure 4. (a) The Functional Viewpoint of IIRA. (b) RAMI 4.0 ©Plattform Industrie 4.0 and Zentralverband Elektrotechnik-und Elektronikindustrie e.V. (ZVEI)

a) Business layer: This layer consists of the business models and information about the business's components, such as the service and or the products that it offers. The business rules and practices are pre-defined and serve for governing the manner in which business processes are executed.

b) Functional layer: The functions of the architecture model are characterized and detailed and their relationships are established. The functions are designed and rendered and are autonomous, not dependent upon the processes or its utilization in the architecture.

c) Information layer: This layer controls and governs the data and information that is used in the architecture. It also serves the purpose of analyzing the information that is being exchanged and the level of data quality.

d) Communication layer: The communication between layers, systems and all the components that are executed, while at the same time interoperable are described by this layer.

e) Integration layer: This layer provides a connection between all the layers of this architecture and the physical components. It also addresses network and software integration.

f) Asset Layer: This layer includes all the systems at the physical level such as shop floor machines, the human interaction with the systems as well as other physical objects.

The RAMI 4.0 model was built after studying existing approaches and incorporating them into the interoperability stack. Approaches such as OPC-UA: Basis IEC 62541 [87] (for the Communication layer), IEC Common Data Dictionary (for the Information layer), Field Device Integration technology (for the Functional and Information Layer) and AutomationML and ProSTEP (for design and end to end engineering) have been used in designing and building the interoperability stack [1]. The architecture also incorporated IEC 62890 functions to address life cycle improvement and value stream mapping.

2.5.2 IIRA

The IIRA was built as an open architecture with a wide array of applications across industries [86]. It provides a model to create systems based on IIoT without any constraint on the system's use of any specific standards or protocol. The IIRA's domains that are analogous to RAMI 4.0's layers (as seen in Figure 4a) are the following:

a) Business domain: This domain is at the top of the IIRA functional viewpoint and consists of business-related information at the enterprise level. It also manages systems that are used across the business such as Manufacturing Execution Systems (MES) and Enterprise Resource Planning (ERP) in line with objectives and goals pertinent to the business.

b) Information domain: This domain consists of functions that are tasked with the purpose of data analysis and assessment of data quality to procure knowledge of all the system's components.

c) Application domain: This domain builds on the information logged by the information domain and applies logic to it to improve system performance and efficiency of operations.

d) Operations domain: This domain assesses the operations that are planned and ongoing, performs scheduled maintenance and diagnostics, with the end goal being ensuring optimal performance.

e) Control domain: This domain consists of functions that control the implementation of processes at the physical system with feedback.

One can see from Figure 4 that the functional layers of RAMI 4.0 and the domains of IIRA are related and have the same high-level tasks distributed amongst them. Similar to RAMI 4.0, IIRA also applies OPC-UA to model its communications and network. This enables syntactic interoperability between these two reference architectures. Although IIRA is not specifically designed for manufacturing applications, the linkage with RAMI 4.0 establishes a foundation for its application in the manufacturing domain. Not only has interoperability within these architectures been established, but interoperability between these architectures has also been mapped.

2.5.3 IBM Industry 4.0 Reference Architecture

IBM introduced the Industry 4.0 Architecture (Figure 5) with a reference model that consists of two layers with functions distributed amongst them [88]:

a) The Edge integrates the Plant and the Enterprise layers, with the ability to connect with legacy systems and equipment and also the protocol that they follow. The Edge can work independent of the Plant and Enterprise layers, without any integration by controlling the systems components at the device level directly. Smart devices and equipment can communicate all the way up through to the Enterprise level from the Edge when integrated. The Edge undertakes the task of data analytics and transformation and communicates this information through the layers.

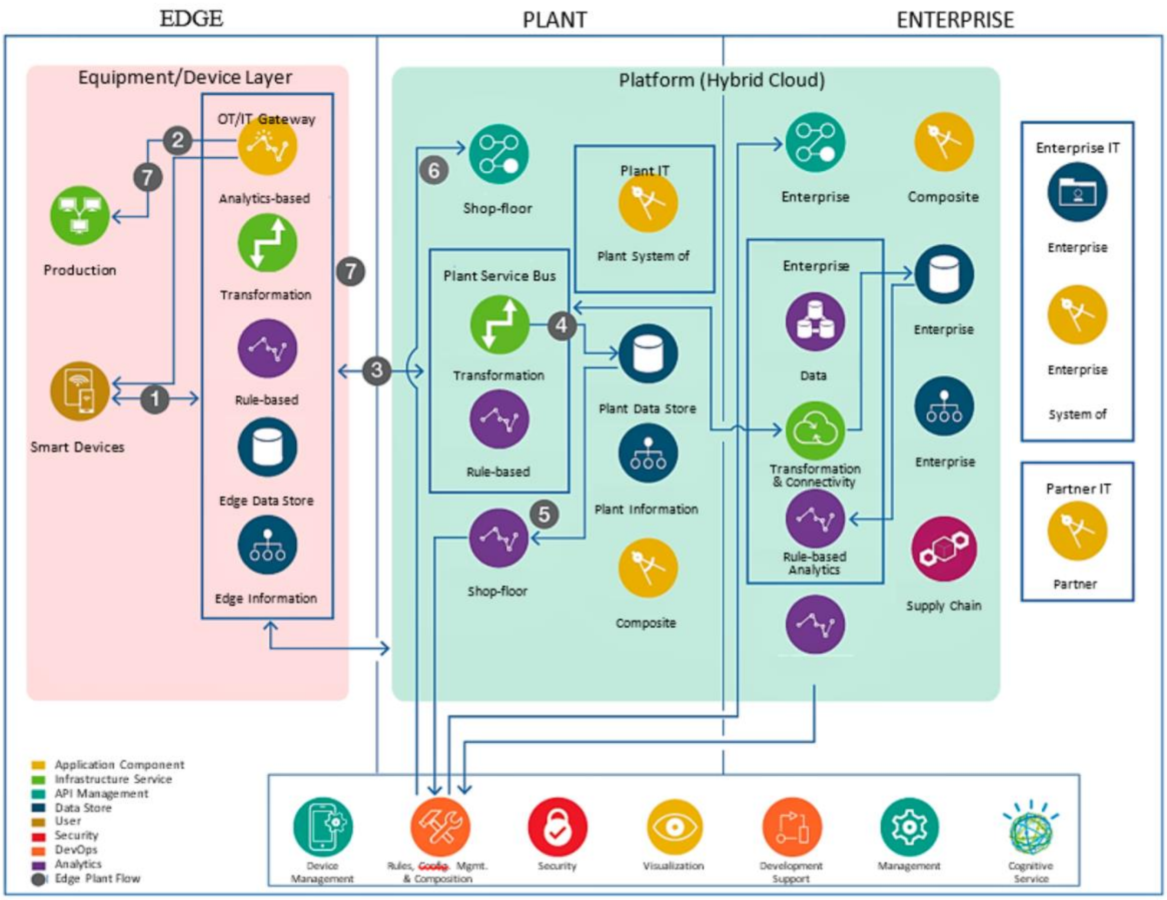

Figure 5. The IBM Industry 4.0 Architecture consisting of an Edge layer and Platform layer.

(Figure adapted from [88])

b) The Hybrid Cloud or the Platform consisting of the Plant and the Enterprise levels perform plant wide and enterprise wide analytics. The Plant level can direct MES to take appropriate decisions based on the analyzed data and the Enterprise level can similarly make decisions based on factories, locations, etc.

It is important to note that the architecture incorporates the OPC-UA communication standards (similar to RAMI 4.0 and IIRA) and is also designed keeping in mind the ISA-95 specifications.

2.5.4 NIST Service-Oriented Architecture for Smart Manufacturing

NIST proposed a service-oriented architecture (Figure 6) that is specifically aimed at smart manufacturing [89] by linking operational technology (OT) and information technology (IT) via a manufacturing service bus. It also provides a Business Intelligence (BI) service that ensures all stakeholders are in contact. The IT domain services consist of all IT operations right from the system level to the enterprise level (ERP, MES, etc.). The OT domain services address the processes and tasks that are assigned to the physical level, the components on the shop floor. There is also another domain that addresses the virtual capabilities of the architecture, the virtual domain. This domain's services are primarily to ensure data quality and data management are achieved at the highest level as well as simulating models for the manufacturing tasks at hand. The other high-level services such as enterprise level security, product life cycles, manufacturing value stream and information management are attended to by management or common services.

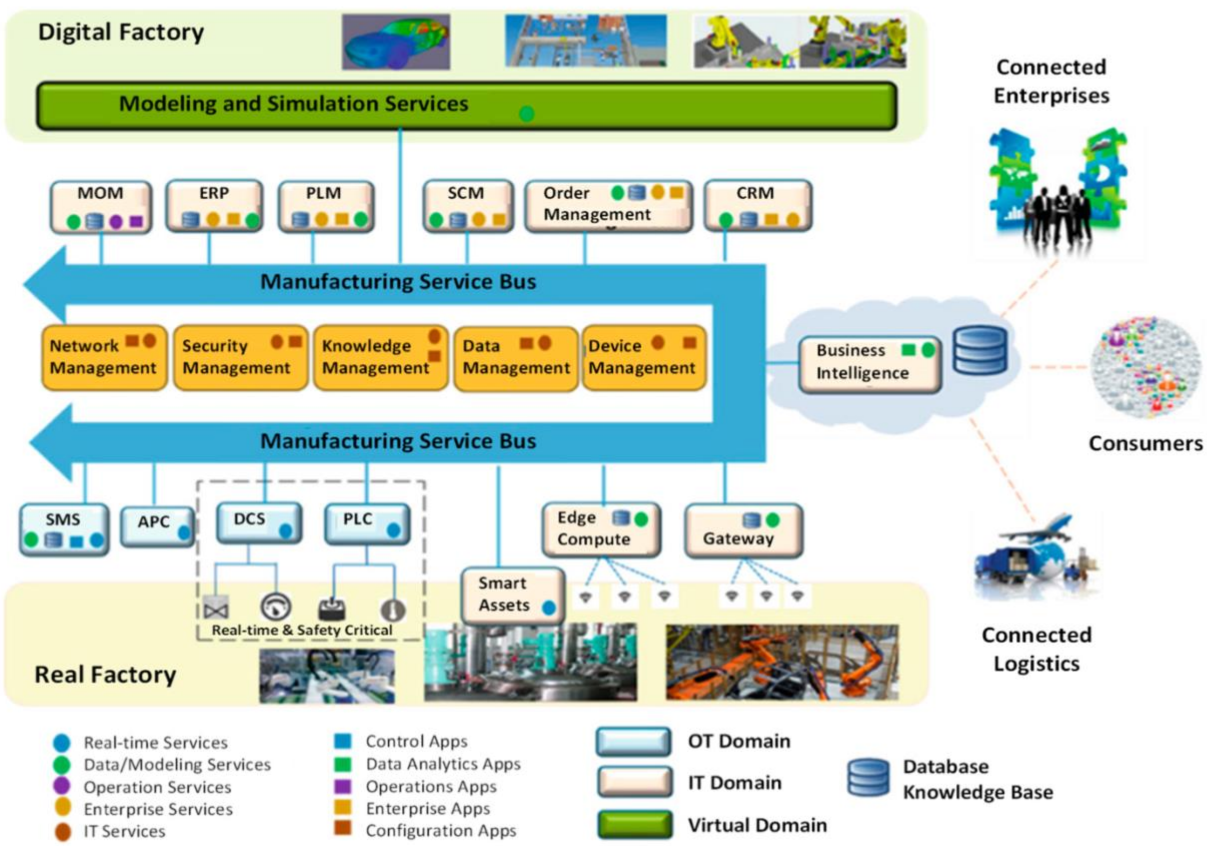

Figure 6. NIST service-oriented architecture for smart manufacturing working on a shared manufacturing bus.

(Figure adapted from [89])

Terminology used in Figure 6: Smart Manufacturing Architecture (SMS), Distributed Control Systems (DCS), Customer Relationship Management (CRM), Manufacturing Operation Management (MOM), Supply Chain Management (SCM), Product Lifecycle Management (PLM).

2.6 Real-World Implementation of Industry 4.0 Components

Since most of the reference architecture models are still in the conceptual phase, there are very few implementations that have been documented. A successful implementation of emerging technologies has been achieved by Lüder et al. in which they integrate a production plant model to an OPC UA client by the use of AutomationML and an administration shell that is built on IEC 61131-3 [90]. AutomationML is a neutral format for the exchange of data developed by Daimler, ABB, Siemens, Rockwell, Kuka, Zühlke, netAllied and universities of Karlsruhe and Magdeburg [91]. The model (Figure 7) uses a combination of Modbus Transmission Control Protocol (TCP) and a fieldbus to connect to the Programmable logic controller (PLC) and is controlled by a Raspberry Pi. The OPC-UA client is on the computer that is able to access the plant and make changes according to the functions provided by the PLC. The OPC UA information model is generated and the communication with the AutomationML model is set up. The information generated by the OPC UA model is transferred into the OPC UA server based on OPC standards. The network connection is configured automatically by AutomationML.

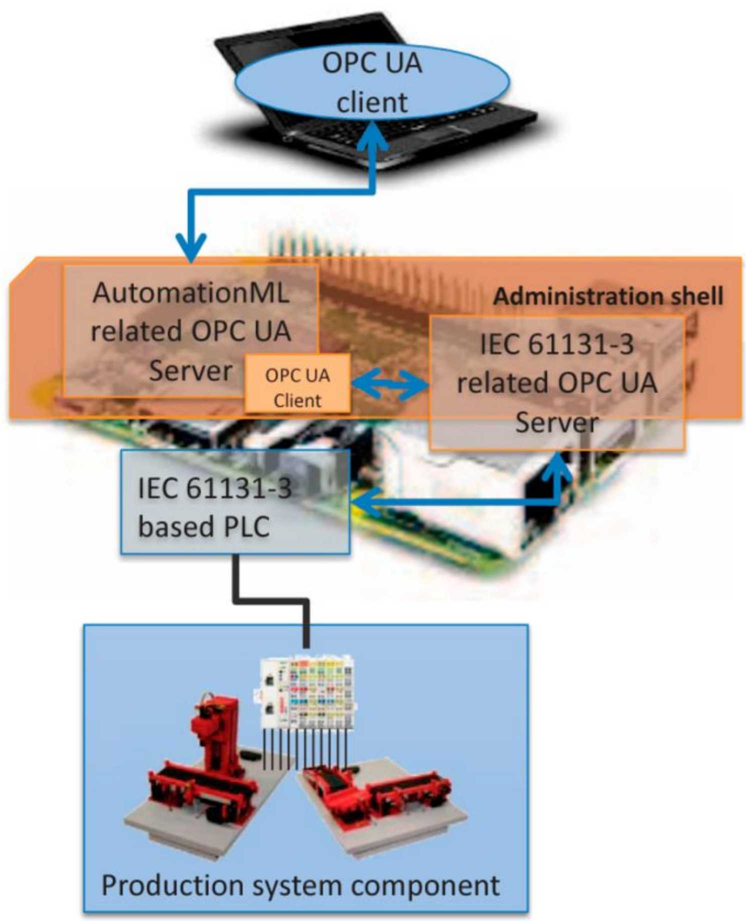

Figure 7. AutomationML implementation.

(Figure adapted from [91])

2.7 R&D Challenges

With the advent of the numerous architectures that have been proposed, there are many areas of interest for research that are gaining importance. A comprehensive list of the most significant areas of focus has been compiled by researchers after interviewing seven industry experts in the area of manufacturing [45]. The results can be seen in Table 1, in which the R&D challenges are listed and corresponding to each challenge, views of experts are recorded. A check mark denotes that the expert agrees and proposes the challenge as a significant area for research.

We can see that integration of shop floor, business processes, and cloud services for plug-and-produce work received the support of four experts. Similarly, asynchronous, and interoperable communication mechanisms and publish/subscribe also received four votes. Common language for standardizing object and services too received four votes. We notice that the three aforementioned propositions have one thing in common: they all address interoperability directly. Integration across the enterprise is a core for the successful implementation of interoperability. Interoperable communication mechanisms address the issue of the exchange of data and information which would include syntactic, as well as semantic interoperability. The proposition of the use of a common language for standardizing object and services also addresses the core issue of interoperability. While these are important areas of research for interoperability for the future, there are also other areas in the smart manufacturing domain that are very important. NIST has conducted workshops on enabling composable service oriented manufacturing systems [51]. The topics discussed include the setting up of messaging standards in smart manufacturing, the reference models and architectures of smart manufacturing, the applications of smart manufacturing and marketplaces that offer manufacturing services and the development of ontologies.

Table 1. Summary of R&D challenges identified by experts.

R&D Challenges	**Expert**						
	1	**2**	**3**	**4**	**5**	**6**	**7**
Integration of shop floor, business processes, and cloud services for plug-and-produce work (P1)			✓	✓	✓	✓	
Fully-integrated industry—inter-enterprise, cloud-based publication and sharing of services (P1)						✓	
Service representation, composition, discovery, registration, and matching (P1)			✓				
Service definition—shift of focus from macro to micro services in manufacturing SOA (P1)							✓
Representation of complex capabilities such as cognitive systems or analytics as services (P1)		✓					
Standardized taxonomy of manufacturing processes as services (P1/P2)		✓	✓			✓	
Asynchronous and interoperable communication mechanisms and publish/subscribe API (P1)	✓			✓	✓	✓	
Need for ‘smarter’ objects to execute more complex services (P2)							✓
Standardized description of objects as prerequisite for standardization of services (P2/P4)			✓			✓	
Common language (e.g., OPC-UA) for standardizing object and services (P2)	✓		✓		✓	✓	
Mechanisms for orchestration, filtering, aggregation, and sharing of cloud services (P3/P4)		✓			✓	✓	
New models for the economics of acquisition, implementation, and integration (P3)						✓	
New models for automated, on-the-fly creation and governance of value networks (P4)				✓		✓	
Horizontal integration of product life-cycles through micro-services (P4)		✓					
Need for autonomous, self-/environment-aware, intelligent devices for service-orientation (P4)			✓				
Concurrent optimization of order-to-cash and design-manufacture-maintenance cycles (P4)							✓
Lack of horizontal integration and reconfigurable manufacturing capabilities in ISA-95 (P5)		✓					
Integration of ISA-95 into a cloud-based architecture (P5)						✓	
Transition from thousands of enterprise-wide applications to consistent cloud-based services (P6)		✓					
Lack of modularity of legacy manufacturing systems hindering ‘composability’ (P6)		✓					

PX: Proposition X

(Table adapted from [45])

2.8 Discussion

The evolution of the traditional hierarchical models of enterprise control system integration, or the automation pyramid, towards integrated networks of smart devices, cloud-based services, and extended enterprises demand seamless communication and information exchange between heterogeneous and traditionally-siloed systems. Thus, there is a need for both vertical interoperability between shop-floor automation devices and services as well as horizontal interoperability between enterprises and cloud service platforms. The hierarchical structure of a pyramid that often inhibits interoperability between non-adjacent layers must be transformed into an interoperable, heterarchical network that facilitates decentralized operations and decision making. CPPS, cloud manufacturing, smart manufacturing and other emerging paradigms follow a highly interoperable, hierarchy-free structure.

Chapter 3

ML-based SPHM Framework to Support Manufacturing System Health

This chapter proposes a Smart Prognostics and Health Management (SPHM) framework for the monitoring of manufacturing system health. An overview of the existing approaches to PHM, the challenges involved in its implementation, and state-of-the-art modeling approaches is provided. We then apply the SPHM framework to a milling machine operation and present its res

3.1 Introduction

A Smart Manufacturing (SM) paradigm consists of interoperable layers that are capable of vertical as well as horizontal integration [92]. Figure 8 shows the different layers according to the ISA-95 Automation Pyramid. The physical layer of SM consists of the field devices that comprise of sensors and actuation equipment. To monitor and analyze the devices on the physical layer, data from all the individual components need to be incorporated into a single stream, which then provides us with a context of the entire operation. Not only do the data give us context about the operation, but properly formatted data allow faster deployment of AI techniques. Quicker implementation of these condition monitoring techniques allows early detection of potential failures. However, the data generated comes from various sources involving multiple parameters, resulting in complex formats and sometimes, redundancies. Challenges in effective implementation of PHM techniques for predictive maintenance include 1) setting up of machines and sensors, 2) ability to collect and understand data, 2) preprocessing of sensor signals, 3) identification of PHM tasks, and 4) selection of appropriate AI methods for modeling. To ensure that manufacturing data can be acquired, preprocessed, and analyzed to achieve PHM related goals, all the steps involved must be detailed in a systematic manner.

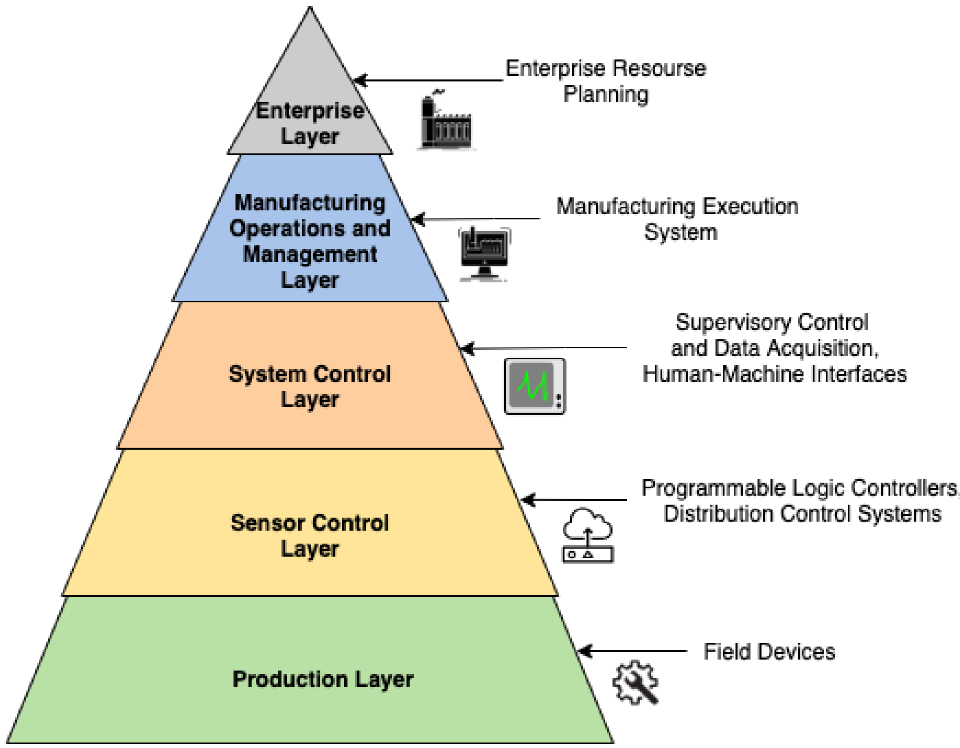

Figure 8. Top-down approach to manufacturing system according to ISA-95 Automation Pyramid.

3.2 Prognostics and Health Management

3.2.1 Overview of Maintenance Strategies

Maintenance has evolved over the years, and with the advent of AI, proactive approaches to maintenance are being adopted not only as a cost-cutting measure, but also as a competitive strategy [93]. Before diving into the details of the state-of-the-art maintenance approaches, we briefly discuss maintenance practices that are adopted across manufacturing. Broadly speaking, maintenance strategies follow one of the three main approaches:

- Unplanned or reactive maintenance—typically allows for machinery to breakdown, after which it is analyzed and repaired.
- Planned or preventive maintenance—an assessment of the system is conducted at regular time intervals to determine whether any repair/replacement is necessary. It

is important to note that the health of the system is not taken into consideration in establishing the time intervals.

- Predictive maintenance—a data-driven approach in which parameters concerning the health of the system are used to monitor the condition of the equipment and in determining the RUL.

3.2.2 Multi-Faceted Approach to PHM

Condition-Based Monitoring (CBM) techniques and Remaining Useful Life prediction are important features of predictive maintenance strategies leading to PHM methods. It would be remiss to state that CBM and RUL define PHM methodologies, since PHM methods are multi-faceted approaches. PHM methods consist of: data acquisition and preprocessing, degradation detection, diagnostics, prognostics and timely development of maintenance policies for decision making [94], [95]. A brief overview of PHM's many facets is provided as follows:

1. Data acquisition and preprocessing: For any predictive problem in maintenance to be solved, the availability of data is of utmost importance. IoT devices and smart sensors are typically used to acquire data in manufacturing settings. The data are recorded and evaluated in real-time as certain anomalies may be detected at an early stage by maintenance engineers or control systems. The collection of such data is extremely important as it provides vital information that helps to understand the relationships between the heterogenous components of the system. Once the data are collected, they are analyzed and preprocessed to ensure that crucial information which helps in failure detection is obtained.

2. Degradation detection: Identifying that a component is degrading or that it is bound to fail is the next step once the data have been collected and prepared. Anomalies and failures can be detected using sensor readings and by other specified criteria, such as surface roughness, temperature, size of tools/equipment, etc.

3. Diagnostics: Once a determination is made that a failure is occurring, understanding the cause of the failure is the next step. Failure types can be categorized to evaluate

the extent of the failure, helping in finding its root causes. Operating conditions of individual components can be analyzed along with their interactions to help diagnose the cause of failures.

4. Prognostics: With the ability to detect failures using diagnosing mechanisms, predictive methods are used to predict the system health to avoid potential failures. Model-based prognostics involve Physics-of-Failure (PoF) methods to assess wear and predict failure. However, such approaches are limited as even minor changes to the operations can result in poor predictive power. Data-driven approaches are becoming more common for prognostics with the use of DL and ML techniques. By using data-driven methods along with crucial information from physics-based methods, highly accurate predictions can be made about systems.

5. Maintenance decisions: Based on results from the predictive methods developed, manufacturing enterprises can determine policies to be followed for maintenance planning that will help with less downtime, higher yield, and a reduction in losses.

The importance of enumerating the many phases of PHM is to help us understand that predictive maintenance is an aggregation of methods from data engineering, reliability and quality engineering, material sciences, DL, ML, and organizational decision making. While PHM methods in manufacturing face several challenges, a fundamental one is a requirement for an interoperable approach that allows its implementation across different industries.

3.2.3 Challenges in Implementing PHM in the Industry

While there are significant advances being made across different fronts of PHM, there are some challenges that arise as well. Researchers from the National Institute of Standards and Technology (NIST) outline some of the most significant challenges in PHM that can be categorized as follows [96].

3.2.3.1 In Prognostics

- Insufficient failure data or excessive failure data may skew prediction of RUL

- Inadequate standards to assess prognostic models
- Lack of precise real-time assessment of RUL
- Uncertainty in determining accuracy and performance of prognostic models

3.2.3.2 In Diagnostics

- Expertise required in diagnosis of failures
- Limitations due to lack of training and formal guidelines in authentication of diagnostic methods
- Difficulty in diagnosis due to outliers, noise in signal data and operating environment

3.2.3.3 In Manufacturing

- Ability to effectively assess electronic components
- Integration of sensors and field devices with PHM standards
- Inconsistencies in data, data formats, and interoperability of data in manufacturing facilities
- Inadequate correspondence between production planning and control units and maintenance departments
- High level of complexity and heterogeneity in manufacturing systems

3.2.3.4 In Enterprises

- Proactive involvement required towards maintenance to view PHM as a cost-saving approach and not a cost-inducing one
- Enterprises with legacy machines and equipment tend to go with one of the traditional approaches to maintenance, even though PHM methods are more effective

- Securing funding for PHM projects

3.2.3.5 In Human Factors

- User friendly interfaces and applications

- Collection of expert knowledge

- Improvement in outlook towards implementing changes to existing mechanisms

Most of the research being conducted is aimed at monitoring system health and RUL in PHM, while the other areas of PHM are not given as much importance. Our aim is to address some of the difficulties faced in the different categories of PHM. Since prognostics is one of the most significant areas in PHM, we look at some of the modeling approaches.

3.2.4 Overview of Prognostics Modeling Approaches

There are three main prognostics approaches to PHM modeling: physics-based models, data-driven models and hybrid models that are a combination of physics-based and data-driven models [97]. While all three approaches are used in industry, the application of prognostics modeling also faces several challenges [98]:

- Lack of readily available data in a standardized format

- Insufficient failure data due to imbalance in data classes

- Lack of physics-based parameters in the data

Industrial data from manufacturing systems are also often complex and require a great deal of preparation to be acceptable for modeling. Due to these reasons, efforts are required to prepare and preprocess the data to make them suitable for prognostics modeling. Data-driven and hybrid models are often preferred over physics-based models given the flexibility of analytical techniques that can be used. A synopsis of physics-based, data-driven and hybrid models based on [97]–[100] is outlined in Table 2.

Table 2. Advantages and disadvantages of prognostics modeling approaches.

Modeling Approach	Advantages	Disadvantages
Physics-based models	1. No-randomness involved, resulting in accurate analysis 2. Can be used with small datasets	1. Complexity in implementing and require intricate laboratory settings 2. Expertise in system modeling is required
Data-driven models	1. Little expertise in system modeling is required 2. Easy implementation 3. Cost-effective since there is no need to simulate operating conditions	1. Lack of suitable data 2. Low quality of available data 3. Difficulty in attributing causes of failure
Hybrid models	1. Can be used with small datasets 2. Not that difficult to implement 3. Flexibility in modeling	1. Selection of parameters involves high level of complexity 2. Balanced data with failure events required

In the last two decades, there have been great strides made in improving physics-based prognostic approaches. Several of these approaches are reviewed and applied to rotating machinery by Cubillo et al. [101]. Physics of Failure (PoF) methods have been tested on electronic components in monitoring the health of electronic components by Pecht et al. [102]. The RUL of lithium-ion batteries has been predicted by physics-based models in [103]. There are also several publications that address prognostics modeling based on evolutionary methods derived from bio-inspired [104], [105] and neuro-inspired algorithms [106], [107]. However, based on current research and industry trends, we will limit our focus to data-driven approaches.

3.2.5 Current Trends in PHM Research

Advanced algorithms and optimization techniques are at the forefront of problem solving in PHM areas, and a review of the current state of research is necessary to understand these topics. ML and DL have become the choice of modeling techniques in studies that undertake data-driven approaches. While there is an abundance of analytical techniques available, there are few publicly available datasets for PHM research. Most datasets are limited to those released by academic institutions and government organizations. This has resulted in certain datasets being benchmarked to test prognostics models, as seen in [98], [108]. Datasets that have been used in PHM research are often from PHM data challenges, and the modeling objective can be grouped into four main tasks: prognosis, fault diagnosis, fault detection and health assessment. An in-depth discussion of these datasets can be found in the review conducted by Jia et al. [109]. Another area from which PHM can be approached is from the perspective of manufacturing models. These manufacturing approaches are designed keeping in mind the objectives of maintenance and PHM policies. We will now shift our focus to ML, DL, Health Index, and manufacturing approaches that are the frontiers of PHM research.

3.2.5.1 Applications of Machine Learning in PHM

ML models have applications in a wide range of PHM areas. Although most of the research is focused on CBM and RUL prediction, some works focus on fault detection as well. An extensive study on the use of Support Vector Machines (SVM) in RUL prediction was conducted by Huang et al. [110]. The authors investigate how SVM works in condition monitoring in a real-time setting, as well as in future RUL predictions. Mathew et al. [111] propose several supervised ML algorithms, such as Decision Trees, Random Forest, k-Nearest Neighbors (kNN) and regression, in estimating the remaining lifecycles of aircraft turbofan engines by a comparison of the Root Mean Square Error (RMSE) metric. It was identified that the random forest model performed best in this setting. Researchers in [112] compare the performance of neural networks, Support Vector Regression (SVR) and Gaussian regression on data from slow-speed bearings that consist of acoustic emission readings. Implementations of techniques such as Least Absolute Shrinkage and Selection Operator (LASSO) Regression, Multi-Layer Perceptron (MLP), SVR and Gradient-Boosted Trees (GBT) are tested on data collected from Unmanned Aerial Vehicles (UAV) in [113]. In this case, non-linear techniques were preferred over linear models, with the best performance achieved by GBT. In fault detection, an SVR outperforms multiple regression on a milling machine dataset, especially when more data are used from sensors [114]. A review of Machine Learning techniques used in intelligent fault detection was conducted by Lei et al. [115], and their challenges were outlined. It is important to note that there are several studies that use semi-supervised ML methods in fault detection of manufacturing equipment, as seen in [116]–[118], but we limit our discussion of these topics to the scope of this chapter.

3.2.5.2 Applications of Deep Learning in PHM

Deep Learning (DL) methods have evolved as frontrunners in RUL assessment largely due to the deep architectures deployed and the ability to tweak the optimization parameters. Recurrent Neural Networks (RNNs) are popular DL methods used in PHM due to their wide range of applicability. Malhi et al. [119] focus on preprocessing of signals using wavelet transformation and apply RNN to investigate its effects on performance. Heimes [120] uses RNN with an Extended Kalman Filter (EKD), backpropagation and Differential Evolution (DE). Research conducted by Palau et al. [121] implemented a Weibull Time-To-Event (WTTE) method with an RNN to predict time-to-failure and demonstrate how it affects real-time distributed collaborative prognostics. A novel method using embedded time series measurements that does not take into consideration any prior knowledge about machine degradation was developed by Gugulothu et al. [122]. Recently, probabilistic generative modeling using Deep Belief Networks (DBN) are being used for feature extraction and in RUL estimation. Authors in [123] argue that feature extraction from data belonging to SM and I4.0 manufacturing can be troublesome due to requirements of extensive prior knowledge, and deploy a Restricted Boltzmann Machine (RBN)-based DBN to estimate RUL. Another interesting study by Zhao et al. [124] uses DBN to extract features, supplemented by a Relevance Vector Machine (RVM) in the prediction of RUL of battery systems. A multi-objective DBN ensemble using evolutionary algorithms was employed in RUL prediction of turbofan engines by [125]. Methods such as RBM have also been implemented with regularization to generate features that are correlated with fault detection criteria [126]. Convolutional Neural Networks (CNNs) have also been used for machine health monitoring with one-dimensional data in [127]–[130] and for feature extraction and automated feature learning with two-dimensional data in [131]–[136]. Comprehensive reviews of Deep Learning methods in PHM, such as Autoencoders, RNNs, RBM and DBN, were conducted by Khan et al. [137]

3.2.5.3 Health Index Construction

Another important area in health monitoring and management is the construction of a Health Index (HI) from input data parameters and using the HI in fault detection and prognostics. HI's are developed using Principal Component Analysis (PCA), and similarity matching by using distance measures for RUL estimation of a factory slotter is analyzed by Liu et al. [138]. In lifecycle prediction of battery systems, Liu et al. [139] develop a novel technique to extract HI while preserving important degradation information. RUL predictions using HI compared to ones without explicit HI on data from induction motors show that HI-based RUL prediction is preferred [140].

3.2.5.4 PHM Using Manufacturing Paradigms

Over the last few decades, manufacturing environments have been revolutionized from a multi-objective viewpoint. Not only are costs and yield the sole objectives, but product customization, sustainability, modularity in the shopfloor and servitization are equally important. This multi-objective approach to manufacturing has allowed the area of PHM to be built-in while designing new systems. Approaches such as mass customization, reconfigurable manufacturing, service-oriented manufacturing and sustainable manufacturing allow the incorporation of PHM goals within the manufacturing system's setup [141].

Mass customization is a multi-dimensional approach to manufacturing that deals with product design, manufacturing processes and the manufacturing supply chain [142]. This approach presents a shift in the traditional manufacturing objective, changing it from a high production volume with a low variation in products to a high variety of products with a lower production volume. This of course presents its own challenges to PHM due to the sheer number of customizations required in production processes. To tackle this, maintenance policies are integrated based on condition monitoring of systems and order volumes by Jin and Ni [143]. Decisions can also be made from a cost-based perspective by including all the maintenance and production costs in the objective [144].

Reconfigurable manufacturing systems (RMS) are modular in their design, allowing changes in their structure to adjust to any inherent changes or shifts in market demands [145]. RMS systems have maintenance policies dependent on their structure: parallel, series, series-parallel, etc. [146]. Preventive maintenance-based RMS was developed by Zhou et al. [147]. The objectives of reduce, reuse, recycle, recover, redesign and remanufacturing were incorporated into RMS to improve the response time to manage system health by Koren et al. [148].

Service-oriented manufacturing offers a Product Service System (PSS), allowing products and services to be picked based on customer needs [149]. PHM services can be offered depending on the manufacturer's needs, enabling a highly customized approach to PHM services. To maximize the prognostic and diagnostic capabilities of Original Equipment Manufacturer (OEM), a cloud-based approach has been developed by Ning et al. [150].

3.3 An Interoperable Framework for SPHM of Manufacturing Systems

Current approaches to PHM are tailor-made to individual systems and components on the shopfloor. Predictive maintenance methodologies, although applicable to heterogeneous machinery, are generic in their approach, without specifying the particulars of data acquisition, modeling, and applications. SPHM is a concept that addresses the many facets of PHM in Smart Manufacturing in an interoperable manner. SPHM's uniqueness lies in its interoperable framework, that addresses all the specifics of PHM by using Industry 4.0 and SM standards in a phased approach. The structure of the framework is loosely based on The Cross-Industry Standard Process for Data Mining (CRISP-DM) [151], an industry standard to apply DM and ML keeping business objectives in mind. The framework consists of three interconnected phases, each with the objective of addressing some of the challenges discussed previously. The proposed SPHM framework for SM with all its phases is shown in Figure 9. The setup and acquisition of data from the shopfloor are covered in the first phase, the preparation of the data and an analysis of the various parameters collected in the data, including an understanding of signal processing, are enumerated in the second phase, and the modeling approaches to SPHM along with their evaluation are discussed in the third phase.

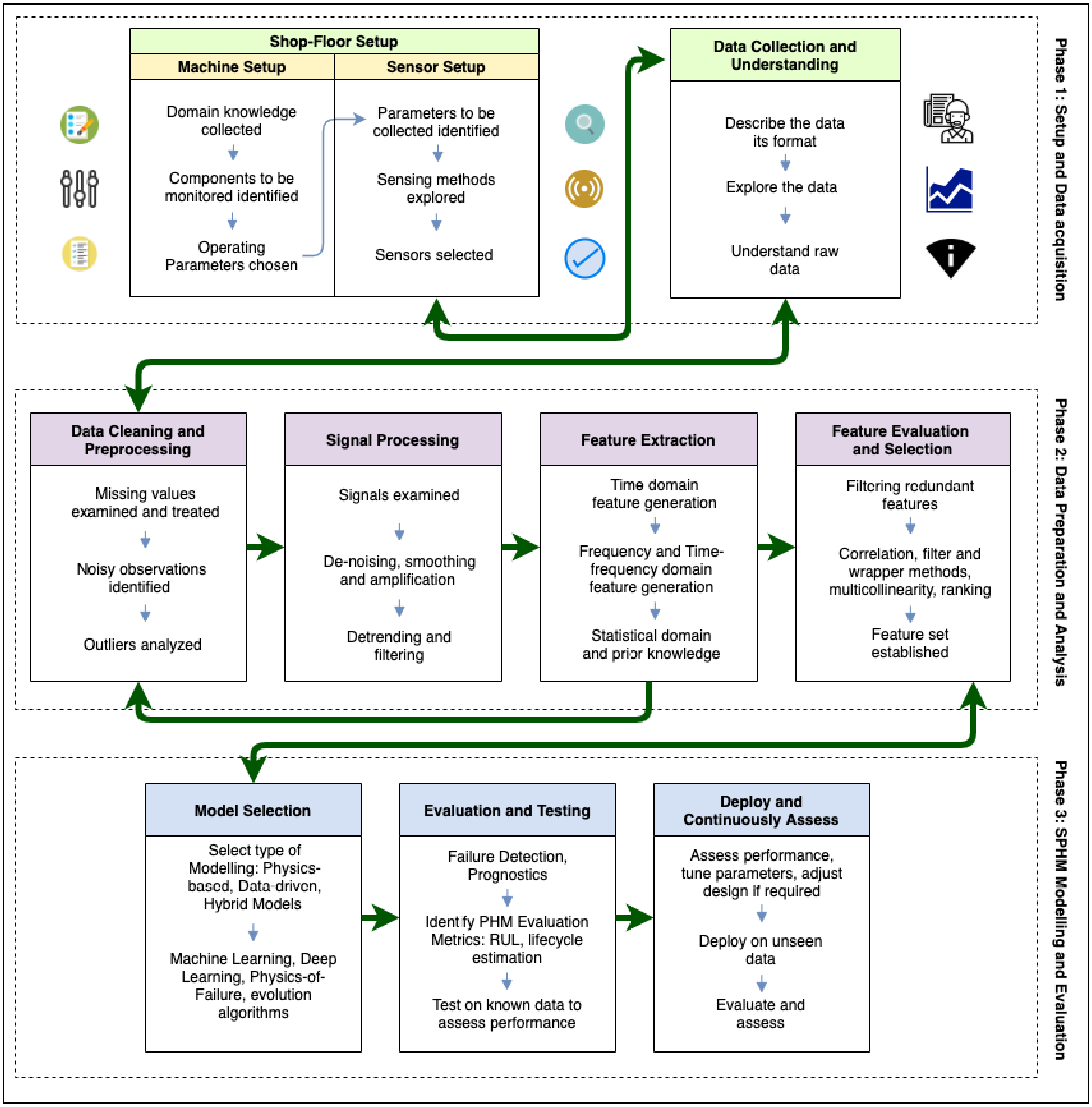

Figure 9. An interoperable framework for Smart Prognostics and Health Management (SPHM) in Smart Manufacturing (SM).

3.3.1 Phase 1: Setup and Data Acquisition Phase

3.3.1.1 Shopfloor Setup

The first phase in the framework involves identifying the machinery or equipment that are going to be assessed, using knowledge from the maintenance and production departments. Prior domain knowledge from engineers and technicians will help us identify which components are crucial to the operation. In manufacturing operations, these components are often tool tips or bearings. Once the identification of components has been established, the next step would be to collect information about the operating parameters and environmental conditions. It is important to note that interacting factors also need to be considered in this step. A detailed report with all data about the operating parameters is prepared and a discussion is held with engineers about the relevance of the parameters. Setting up the equipment and sensors is the next part of this step. Most machinery already come with preinstalled sensors, for example Computer Numeric Control (CNC) machines often consist of sensors to capture electric current, vibration, acoustic emission, spindle, torque, etc. The information from these sensors is vital in analyzing the health of the machine. If additional sensors are required, or if the identified equipment consist of legacy machinery, sensors will have to be retrofitted. Often, there may be sensors that are installed for human-factor purposes, essentially aimed at operator safety. These sensors may have no effect on any prognostics or fault detection methodology, so they may be ignored for selection based on existing knowledge. Using information from operating parameters and domain knowledge, a detailed step-by-step guide to setup and run the machine is produced. Once a strategy is in place to conduct the experiment under standard operating conditions, the next step is to identify an appropriate data collection methodology.

3.3.1.2 Data Collection and Understanding

One of the first and arguably most important considerations in this step is to identify how and where to store the data from operations and devices. A requirement in most organizations is the ability to access data in a secure manner. A key aspect of interoperability is the ability to access data based on the tasks that share it [92]. Cloud-based systems and Big Data platforms provide secure access to data by using cybersecurity mechanisms, ensuring no misuse of the data. Systems can be setup to directly upload and download data that have been collected from the shopfloor. The collected data need to be analyzed and thoroughly reviewed to ensure there were no inconsistencies encountered during the acquisition of data. The data files should also be recorded in a format such that is readable and its suitable for information extraction. Preliminary investigation of the data by using exploratory analyses such as signal plots, range of attribute values and a basic statistical review will help to gain a better understanding of the experiment. Information such as sampling rate and frequencies pertaining to signal measurements should be recorded. Any other a priori information that would aid in describing the data should also be included in this step.

3.3.2 Phase 2: Data Preparation and Analysis

In the data preparation phase, there are a few key steps that need to be followed: data cleaning and preprocessing, signal preprocessing, feature extraction and feature evaluation and selection. These steps are paramount to successful implementation of CBM and predictive maintenance methods as they preserve and extract information from the data that best represents the physical experiment.

3.3.2.1 Data Cleaning and Preprocessing

The procedure to clean and preprocess the data can be understood best by posing the following four questions: (1) Are there any missing values? (2) Is there any noise in the data? (3) Are there any outliers or skewed measurements? (4) Are the data on the same scale? Data cleaning and preprocessing involve many more techniques depending on the type of data recorded, but the treatment of missing values, noise, outliers or skewed instances and scaling and normalization are some of the most crucial ones to be considered.

Missing values can be problematic in any data. Large numbers of missing values can affect the analysis by introducing a bias, causing skewness. Missing value treatment generally involves either the deletion of the record if permissible or replacement of the missing value by using imputation methods [152], [153]. Commonly used imputation techniques are mean, median or mode imputation. In such cases, the missing values are replaced by the respective mean, median or mode of all the values of that attribute. Predictive modeling methods such as regression and kNN are also used for imputation. However, it is critical to understand which attributes contain missing values. If it is the dependent variable, the strategy most likely to be used would be the removal of the instances that contain the missing values, since we have no control over the values in that attribute. In the case of independent variables, we may have the freedom to use one of the imputation methods discussed previously.

Noise in the data is generally caused due to errors in measurement, causing corrupted data. This may cause inconsistencies in modeling and analysis, and possibly provide incorrect results. Examples of noise are duplicate records and mismatch in data types. The ideal method to address noise is to remove those instances affected. If there is significant noise in the attributes, those values may be removed individually, and replaced using one of the imputation methods. Noise in the data may also be caused due to probabilistic randomness. It is important to note that this type of noise, also known as random errors, can be very difficult to predict. These values will have to be examined and are generally retained as they can be accounted for as randomness in measurements. It is essential to note that the noise we discuss here is different from the noise in signals. We will discuss noise in signal readings in the signal preprocessing step.

To answer the third question, outliers can be due to a variety of reasons. Outliers may be genuine observations that exist for a reason. For example, the readings from the data the moment a tool breaks can be considered as outliers. The values of the readings from that instance may be outliers compared to other readings, but they are still genuine readings since they convey essential information. Outliers in the data may also be due to systematic errors or measurement errors. These outliers are problematic since they can be attributed to flaws in readings and convey no genuine information. The preferred solution to address these outliers is by simply removing them from the data. There may also be differences in measurement criteria that may cause inherent skewness and will have to be treated. Sometimes, attributes in the existing format may not convey the information that is necessary due to some form of skewness. The skewness does not necessarily mean there is something wrong with the data. Instead, it could be due to a small sample size or due to intrinsic factors. In such cases, a transformation may be necessary. Commonly used transformations are log, power, square root, etc. It may also be deemed necessary to use a transformation when the attributes are non-linear and need to be linear to form a meaningful relationship with the other attributes or the target.

Another consideration to be made in this step is the scaling or normalization of the data. Normalization or standardization may be required if the data recorded are on varying scales. Having feature values on different scales can cause some features to have more strength in predictions, which is often erroneous. Following are some commonly used methods to scale and normalize data:

1. Min-max normalization

Normalizing data using min–max is a method that shifts all the data values to a scale between 0 and 1. The minimum and maximum values are recorded, and Equation (1) is applied to scale it.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{1}$$

2. Mean normalization

Mean normalization is another method that centers the data around the mean, as shown in Equation (2):

$$x' = \frac{x - \text{mean}(x)}{\max(x) - \min(x)} \tag{2}$$

3. Unit Scaling

For vectors that consist of continuous values, scaling them to a unit length maintains the same direction, but changes its magnitude to 1. See Equation (3):

$$x' = \frac{x}{||x||} \tag{3}$$

4. Standardization

Standardization involves making a loose distributional assumption about the data and scaling it around the mean. Z-Score standardization involves making a normal distributional assumption and using the mean value (μ) and standard deviation (σ), as seen in Equation (4):

$$x' = \frac{x - \mu}{\sigma} \tag{4}$$

The steps involved in preprocessing are applied to new features that are extracted/generated as well. Hence, it is required to note that in the data preparation and analysis phase, the steps go back and forth.

3.3.2.2 Signal Preprocessing

The data collected from sensor measurements in most cases are in the form of signals. Signal processing is a complex field of study and involves identifying the type of signal and modifying it in some form to improve its quality. There are a few actions that need to be taken to enhance the quality of signals. First, we need to consider the type of signals that exist in the data. In manufacturing data, we usually observe signals from sound, vibration, and power. Since these signals are observed across different machines in the industry, we will focus our discussion on preprocessing these signals. Signal preprocessing involves some key tasks, such as denoising, amplification and filtering. We will discuss these topics in brief in this section.

Noise reduction methods or de-noising is a process that diminishes noise in the signals. The entire removal of noise is not possible, so curtailing it to an acceptable limit is the goal of this step. The Signal-to-Noise Ratio (SNR) is a metric used to determine how much of the signal is composed of true signal versus noise. Signals are decomposed using techniques such as wavelet transforms and median filters [154], which preserve the original signal while reducing the amount of noise that it is composed of. Signal amplification is also a method used in signal processing that improves the quality of the signal by using one of two approaches: boosting its resolution or reducing SNR. Signals are generally amplified to meet threshold requirements of equipment being used.

Filtering is another step, pertaining to signal conditioning, in which low-pass and high-pass filters are used in attenuating signals based on a specified cut-off. Low-pass filters block high frequencies while allowing low frequencies to pass, and high-pass filters block low frequencies while allowing high frequencies to pass through. Low-pass filters help in removing noise, and high-pass filters filter out the unwanted portions and fluctuations of signals. The cut-off frequencies are generally chosen depending on the noise observed.

There are also methods to pre-process signals using statistical techniques. Estimation methods such as Minimum Variance Unbiased Estimator (MVUE), Cramer-Rao lower bound method, Maximum Likelihood estimation (MLE), Least Squares Estimation (LSE), Monte-Carlo method, method of Moments and Bayesian estimation, along with several others, are discussed in the context of signal processing in [155].

3.3.2.3 Feature Extraction

Signal measurements recorded are high in dimensionality and consist of readings that cannot be directly used in any form of modeling. This is due to the non-linearity of the machine operation that is also dependent on time [156]. To understand the signal readings in the context of the manufacturing process in question, the relevant information from the signals should be extracted or generated. This information from the signals is extracted as features for the dataset. These features are extracted in the time domain, frequency domain and time-frequency domain. Zhang et al. [157] have compiled a list of features that can be extracted in all three of these domains for machining processes. An overview of these features is as follows.

Time-Domain Feature Extraction

There are several statistical features that can be extracted from signals in the time domain. Features such as maximum value, mean value, root mean square, variance and standard deviation are extracted. Additionally, higher-order statistical features such as kurtosis and skewness are calculated. These values are dependent on the probability distribution function, with kurtosis providing information about the peak of the distribution and skewness explaining if the distribution is symmetrical or not. The Peak-to-Peak feature computes the difference between extreme values of the amplitude, i.e., difference between maximum and minimum values. Crest factor is the ratio of the maximum value and mean values of the signal. A list of time-domain features and their description is provided in Table 3. We refer readers to [157], [158] for in-depth explanations of the features.

Table 3. Time-domain features for commonly observed sensor signals from machining

Index	Feature	Description
1	Maximum	$X_{MAX} = Max(x_i)$
2	Mean	$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$
3	Root Mean Square	$X_{RMS} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}$
4	Variance	$X_V = \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}$
5	Standard Deviation	$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}}$
6	Skewness	$X_V = \frac{1}{n}\frac{\sum_{i=1}^{n}(x_i - \mu)^3}{\sigma^3}$
7	Kurtosis	$X_V = \frac{1}{n}\frac{\sum_{i=1}^{n}(x_i - \mu)^4}{\sigma^4}$
8	Peak-to-Peak	$X_{P2P} = max(x_i) - min(x_i)$
9	Crest Factor	$X_{CF} = \frac{max(x_i)}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} {x_i}^2}}$

(Table adapted from [157])

Frequency-Domain Feature Extraction

Features in the frequency domain are obtained by using a transform on the signal signatures. The Discrete Fourier Transform (DFT) is a method used in spectral analysis of signals. The DFT is based on the Fourier Transform method, see Equation (5) [159]:

$$X(\omega_x) = \sum_{n=0}^{N-1} x(t_n) e^{-jw_k t_n} \qquad \text{for } k = 0, 1, 2, ..., N-1 \qquad (5)$$

where,

$x(t_n)$ = input signal at time t_n,

t_n = nT = n-th sampling instant, for n ≥ 0,

$X(\omega_x)$ = spectrum of x at frequency ω_k,

ω_k = sample from k-th frequency in radians per second,

T = sampling interval in seconds,

$f_s = 1/T$ = sampling rate or samples per second,

N = total number of samples in signal.

The Discrete Time Fourier Transform (DTFT) is a limiting form of the DFT allowing infinite samples, see Equation (6):

$$X(\omega^*) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \tag{6}$$

where,

$x(n)$ = signal amplitude at n^{th} sample,

$X(\omega^*)$ = DTFT of x at n^{th} sample.

The Fast Fourier Transform (FFT) is an algorithm that uses the Discrete Fourier Transform (DFT) method to convert the signal measurements from the original order to the frequency domain by sampling them over time, which is most used for practical purposes. The number of samples is an important parameter to be noted in this stage and is used in the calculation of power spectral density of the signals using a periodogram [160], which is nothing but a ratio of the squared magnitude of DTTF and the number of samples, see Equation (7):

$$P_{x,M(\omega)} = \frac{|DTTF|^2}{N} \tag{7}$$

For a more comprehensive understanding of Fourier Transforms, DFT, DTFT and periodograms, refer to Smith [159], [160]. Features such as maximum, sum, mean, variance, skewness, kurtosis, and relative spectral peaks extracted from power spectra are commonly extracted from signals that are produced by machining operations. The list of frequency-domain features is shown in Table 4. Zhang et al. [157] and Caesarendra et al. [158] provide more details for these frequency-domain features.

Table 4. Frequency-domain features for commonly observed sensor signals from machining

Index	Feature	Description
1	Maximum Band Power Spectrum	$S_{MAX} = Max(S(f)_i)$
2	Sum of Band Power Spectrum	$S_{SBP} = \sum_{i=1}^{n} S(f)_i$
3	Mean of Band Power Spectrum	$S_{\mu} = \frac{1}{n} \sum_{i=1}^{n} S(f)_i$
4	Variance of Band Power Spectrum	$S_V = \frac{\sum_{i=1}^{n}\left(S(f)_i - S_{\mu}\right)^2}{n-1}$
5	Skewness of Band Power Spectrum	$S_S = \frac{1}{n} \frac{\sum_{i=1}^{n}\left(S(f)_i - S_{\mu}\right)^3}{S_V^{\frac{3}{2}}}$
6	Kurtosis of Band Power Spectrum	$S_S = \frac{1}{n} \frac{\sum_{i=1}^{n}\left(S(f)_i - S_{\mu}\right)^4}{S_V^{\frac{4}{2}}}$
7	Relative Spectral Peak per Band	$X_{CF} = \frac{max(S(f)_i)}{\sqrt{\frac{1}{n} \sum_{i=1}^{n} S(f)_i}}$

(Table adapted from [157])

Time-Frequency Domain Features

Like frequency-domain feature extraction, time-frequency features are extracted by using wavelet transforms. Time-frequency analysis provides relationships represented both over time and frequency. This two-dimensional view of the signal, in some cases, can generate features that are not captured by time-domain features or frequency-domain features. Methods such as Short-Term Fourier Transform (STFT), Continuous Wavelet Transform (CWT), Wavelet Packet Transform (WPT) and Hilbert-Huang Transform (HHT) are used to extract features in the time-frequency domain [157], [161]. These methods analyze the signals on a two-dimensional view by using a combined function for the two domains.

3.3.2.4 Feature Evaluation and Selection

The feature extraction or feature generation from signals results in a high-dimensional space with many features. There are often more features than the number of recorded instances after the feature extraction step. This could be troublesome, causing any modeling technique to potentially overfit the data, causing misleading results. Overfitting is caused by some form of redundancy in the high-dimensional space, with not all features being related to the predictor or dependent variable. A solution to this problem is reducing the number of features that are used in modeling. Feature evaluation and selection methods select features that are better in predicting the response variable than all the other features in the feature space. They primarily fall into three main categories: filter methods, wrapper methods and embedded methods.

Filter methods calculate the performance of features across the entire dataset and select the top-performing features. The most used filter method for feature selection is correlation analysis. There are different computational approaches to calculating the correlation coefficient, and Pearson's, Kendall's and Spearman's correlation are the ones that are commonly used. In correlation analysis, pairwise correlation coefficients are calculated between the variables and compared. The analysis is aimed at eliminating features from highly correlated pairs. In a highly correlated pair, either one of the features can be used in prediction while the other is eliminated. There is a concern with this approach, however, as not all features in the dataset may have a relationship with the dependent variable. This may result in a useful feature being dropped during the correlation analysis. To circumvent this issue, one may sometimes calculate the correlation of all the features in the data against the response and use this to eliminate the weaker feature in the pair. Correlation analyses are effective in feature selection. Cross-validation, a method in which data are divided to validate any analysis or modeling, is sometimes used with correlation for feature selection. Methods such as Random Forest to calculate feature importance, Mutual Information to obtain the entropy, analysis of variance (ANOVA) and several others are compared in [162].

Wrapper methods use subsets of the features to identify which ones are more important to the dataset. These methods generally deploy search-based algorithms to find the best features from the feature space. Wrapper methods can be broadly classified into two categories: heuristic search methods and sequential search algorithms [163]. Heuristic methods include Genetic algorithm (GA), Variable Neighborhood Search (VNS), Simulated Annealing (SA), Particle Swarm Optimization (PSO), etc. Sequential search algorithms include Forward Selection, a method in which an empty feature set is used, and features are added individually and evaluated using modeling techniques. Backward Selection is another sequential search method in which the entire feature set is used, and features are evaluated and eliminated depending on model performance. In an exhaustive search method, subsets of features are evaluated against model performance and the best subset is chosen.

Embedded feature selection methods are deployed on subsets of the data along with the modeling techniques. Regularization is an important method in which a penalty is added to the model as a constraint. The regularizer penalizes the coefficients of features in a model, thereby reducing the feature's strength in the model. Popular methods include LASSO, Ridge and elastic nets in regression, and Tree-based methods in Decision Trees, Random Forest, XGBoost, etc.

3.3.3 Phase 3: SPHM Modeling and Evaluation

For this phase, we provide a brief overview of the methods that can be used in modeling, evaluation, and the deployment of the framework. An example of task specific modeling is demonstrated as a part of the use-case.

The most popular modeling approaches in the last decade have been data-driven, with some novel hybrid models being developed as well. Supervised learning methods, unsupervised methods as well as DL have come to the forefront in CBM and in the prediction of RUL. As we have presented in Section 3.2.5, there are a multitude of options to choose from for PHM modeling. Modeling methods not only include regression-based prediction, but also classification models for anomaly detection. Methods such as k-fold cross-validation, holdout, and repeated cross-validation can improve the model performance. The model selection stage usually involves performing many experiments with different hyperparameters until the optimal performance is achieved. Evaluation metrics for models can be chosen based on the type of data-driven or hybrid models. Metrics such as RMSE, Mean Absolute Error (MAE), coefficient of determination (R-Squared), adjusted R-squared and Mallow's CP are used in evaluation and assessment of regression models. Classification models are generally evaluated using a confusion matrix, in which the True Positive, True Negative, False Positive and False Negative classifications are recorded. Based on the confusion matrix, metrics such as accuracy, precision, sensitivity, specificity, F1 score, etc., are calculated and used in evaluating a classifier.

Deploying SPHM models in the field requires development of apps, either web-based or mobile, so that engineers and technicians can analyze, assess, and record the current state of operations based on real-time data. In this work, we suggest how these approaches can be undertaken.

3.4 Case Study: Milling Machine Operation

Milling is one of the fundamental operations in manufacturing engineering and is essential in most, if not all shopfloors. It is an ideal starting point to analyze the SPHM framework in manufacturing and is useful in providing a basis to understanding how even a simple operation can generate such complex data. A typical milling machine setup comprises of the following components: spindle, cutting tool, base, workpiece, *X*-axis, and *Y*-axis traversing mechanism, and a table upon which the workpieces are mounted. The cutting tool is used to remove material from the workpiece by moving it along the different axes via the machine table motion. Old milling machines are operated manually by using the mechanisms to move the cutting tool, whereas newer machines with Computer Numerical Control (CNC) controllers are equipped with a wide range of sensors and automated tool changing mechanisms depending upon the specifics of the operation. The experimental data considered for this work were developed by Berkeley University, California [164], [165].

In this use-case, we apply all the phases of the proposed SPHM framework. The aim is to understand the shopfloor setup; generate, clean, and preprocess data; preprocess the signals; extract and select the final set of features; and identify classification and regression tasks to find anomalous instances and estimate RUL respectively.

3.4.1 Phase 1: Milling Machine Setup and Data Acquisition

3.4.1.1 Milling Machine and Sensor Setup

The setup used consists of an MC-510V Matsuura machine along with the table that it is mounted on. There are three sets of sensors measuring: acoustic emissions, vibrations, and electric current. The acoustic emission sensor is the WD 925 model by the Physical Acoustic Group that has a frequency range of up to 2 MHz. This sensor is secured to a clamping support. The acoustic emission signals are passed through a model 1801 preamplifier built by Dunegan/Endevco. The preamplifier has an in-built 50 kHz high-pass filter. Further amplification of the signals is performed by using a DE model 203 A (Dunegan/Endevco). These signals then pass through a custom-made RMS meter with the time constant set to 8.0 ms. Then, these signals are fed through a UMK-SE 11,25 cable made by Phoenix Contact that is linked to an MIO-16 board made by National Instruments for high-speed data acquisition. Another acoustic emission sensor is mounted on the spindle and the signals follow the same path via the preamplifier, filter, amplifier, and RMS meter to the data acquisition board. The vibration sensor used is an accelerometer built by Endevco, Model 7201-50, that has a frequency range of up to 13 kHz. Vibration signals pass through a model 104 charge amplifier made by Endevco and then through an Itthaco 4302 Dual 24 dB octave filter (low-pass and high-pass). These signals then pass through a custom-made RMS meter and to the MIO-16 board using a UMK-SE 11,25 cable. The vibration sensors are mounted on both the table and the milling machine’s spindle. An OMRON K3TB-A1015 current converter feeds signals from one spindle motor current phase to the high-speed data acquisition board. Another current sensor, model CTA 213, built by Flexcore Division of Marlan and Associates, Inc., also feeds signals into the data acquisition board. See Figure 10 for the experimental setup showing the connections of acoustic and vibration sensors.

The selection of parameters for the experiment was chosen based on manufacturers’ standards and industry specifications. Two types of inserts are selected for the cutting tool, KC710 and K420. They are resistant to wear and can function in environments that involve high friction. The materials chosen for the workpieces are stainless-steel J45 and cast iron. Other important parameters included the setting of the speed of the cutting tool to 200 m/min, Depth of Cut (DOC) of the two settings of 1.5 and 0.5 mm and feeds to two settings of 413 and 206.5 mm/min. The combinations of the numbers of parameters result in 8 different settings under which the milling machine could operate.

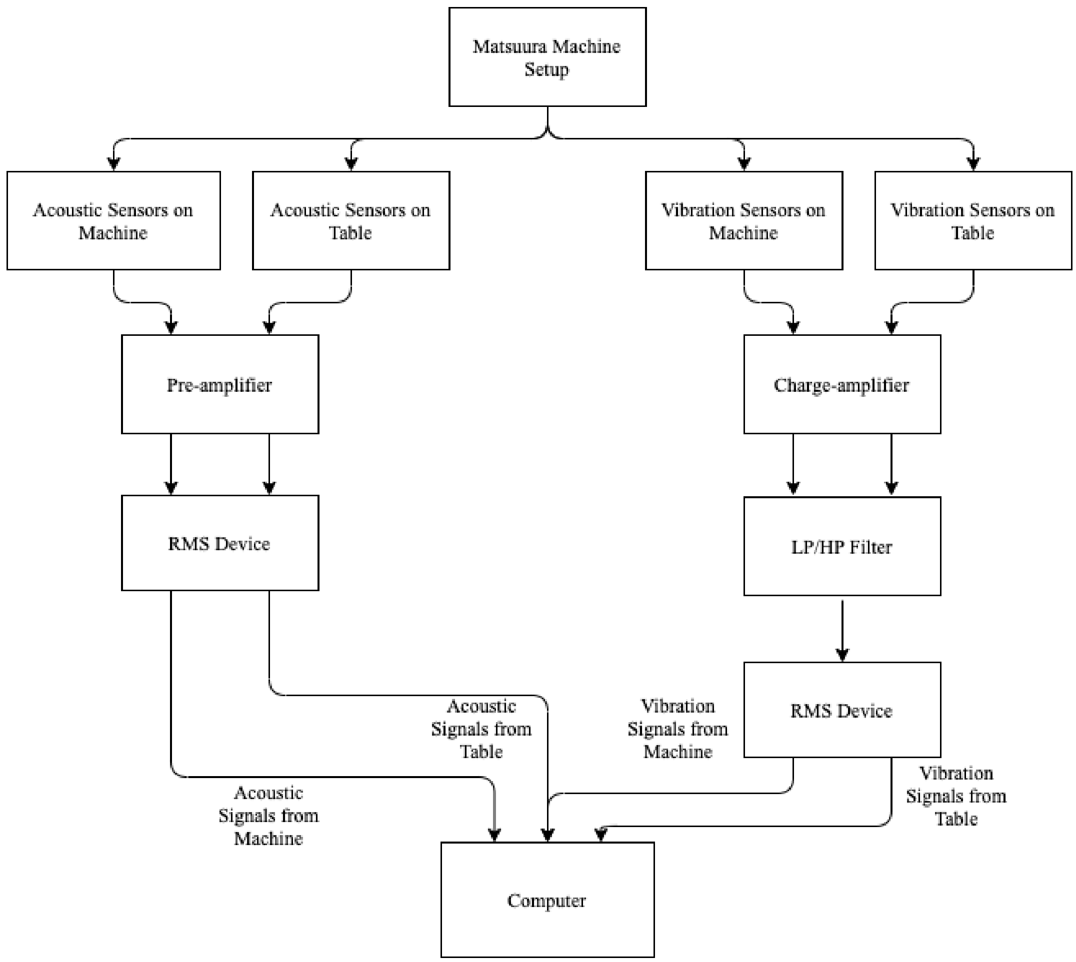

Figure 10. Milling operation setup, adapted from.

(Figure adapted from [164])

The experimental data from [164], [165] consists of 16 cases with varying DOC, feed and materials. These 16 cases are used as experimental conditions and run multiple times. The numbers of runs re determined by evaluating flank wear on the cutting face of the tool, by taking measurements at intermittent but non-uniform intervals.

3.4.1.2 Data Collection and Understanding

The data are recorded as a `struct array' using MATLAB [166] software. The dataset consists of 13 features, out of which 6 are derived from sensor readings. The description of the dataset can be found in Table 5. There are 16 cases, with the DOC, feed and material kept constant for each case. Each case consists of a varying number of runs that are dependent on VB, the degree of flank wear. VB measurements were recorded at irregular intervals up to the limit when significant wear was observed. If we look closely at the MATLAB file, we notice that each of the values under the sensor features (smcAC, smcDC, vib_table, vib_spindle, AE_table and AE_spindle) comprise of a 9000×1 dimensional vector. This is because the sensor's signals are amplified and filtered before being captured, resulting in measurements that are of high dimensions. A view of the first few rows of the dataset can be seen in Table 15 of the Appendix, showing the highly dimensional values in sensor readings. These features need to be preprocessed and transformed so that they can be analyzed more thoroughly. We also note some missing values, and possibly some outliers that can be troublesome while conducting a data-driven approach. This dataset requires preparation and preprocessing for it to be suitable for use in modeling.

Table 5. Features of the milling dataset and their description.

Feature Name	Feature Description
case	Cases from number 1 to 16
run	Counting the runs in each case
VB	Flank wear observed in the cutting tool, not observed after each run
time	Time taken for each experiment, resets after completion of each case
DOC	Depth of Cut, kept constant in each case
feed	Feed, kept constant in each case
material	Material, kept constant in each case
smcAC	AC current at spindle motor
smcDC	DC current at spindle motor
vib_table	Vibration measured at table
vib_spindle	Vibration measured at spindle
AE_table	Acoustic emission measured at table
AE_spindle	Acoustic emission measured at spindle

3.4.2 Phase 2: Data Preparation and Analysis

3.4.2.1 Data Cleaning and Preprocessing

VB is the most important feature in this dataset since RUL assessment and condition monitoring are performed based on VB values. If we observe the dataset closely, we notice that there are missing values in the VB column. This is because VB measurements were taken are irregular intervals until the degradation limit. There are 21 instances identified that contain missing VB values. Since VB is crucial to any analysis that we wish to conduct, the appropriate strategy in this case is to delete the instances in which missing values are observed. After removing the instances with the missing VB values, the dataset is reduced to 146 instances. The next step in this process is to identify any outliers in the data. Figure 11 shows the signal signatures from the six sensors for run 1 of case 2. Compared to signals from other instances (see Figure 32 in Appendix A), the ones from Run 1 of Case 2 are of a much higher magnitude. The sensor measurements for this case have peaks at the following magnitudes: smcAC at 10^{29}, smcDC at 10^{19}, vib_table at 10^{29}, vib_spindle at 10^{34}, AE_table at 10^{34} and AE_spindle at 10^{26}. The rectangular-shaped peaks observed for this case cannot be attributed to any filtering or operation from the experiment. After comparing this anomalous instance against other instances, we can confirm that it is an outlier that is most likely due to measurement error. Similarly, all the instances in the dataset are scanned for abnormalities. We observe that there is only one instance of an outlier, and we choose to discard it under the circumstances, with the final dataset consisting of 145 instances. Before we move on to the next step, it is crucial to note that the data have not been scaled or normalized. This is because scaling/normalization is performed after all the features have been generated.

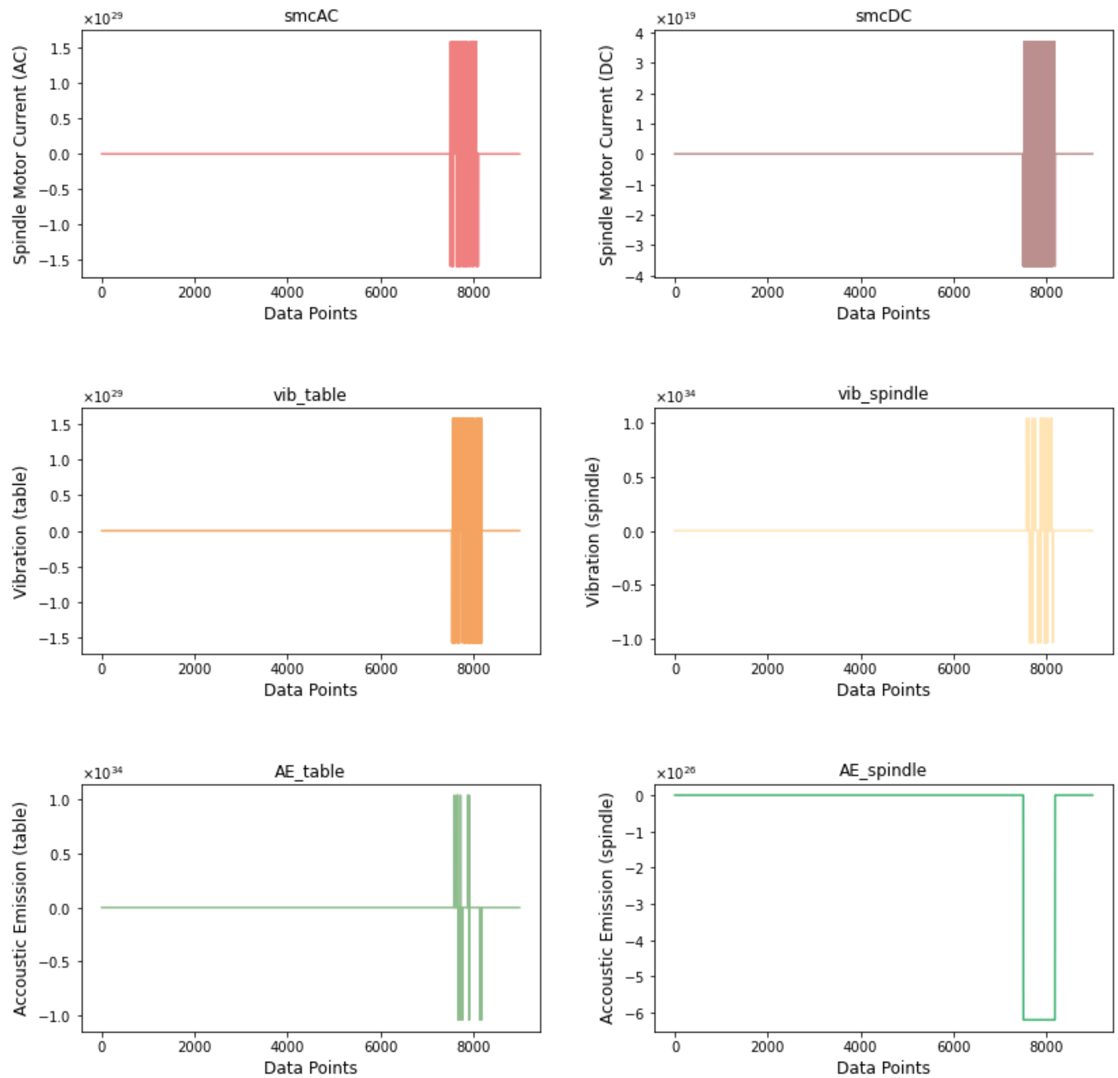

Figure 11. Signatures from the six sensors showing an outlier for case 2, run 1.

3.4.2.2 Signal Preprocessing

The preprocessing of signals from sensors was conducted during the experimental setup, by Goebel et al. [164], [165]. The signals from acoustic and vibration sensors were amplified in the range of ±5 V, according to the equipment threshold. The vib_table and vib_spindle signals were routed through a low-pass and high-pass filter, attenuating any frequency that did not meet the cut-off. The acoustic signals were fed through a high-pass filter to filter out any unwanted frequencies. Cut-off frequencies were identified based on graphical displays on an oscilloscope, with cut-offs of 400 Hz and 1 kHz set for the low-pass and high-pass filters, respectively. An equipment threshold of 8 kHz was set for the acoustic emission sensor, meaning that any frequency observed above that would not be due to machining operations and is filtered out. An RMS meter allowed the signals to undergo some additional preprocessing by smoothing them.

3.4.2.3 Feature Extraction

In this step, feature extraction methods are applied to generate features in the time domain and frequency domain. The methods applied in feature extraction are ones that have been proven to be suitable for machining operations. Time-domain features are extracted using the prescribed feature set in Table 3. This method generates 54 features, i.e., 9 new features for each of the 6 signals. Frequency-domain features are generated using the prescribed feature set in Table 4, generating an additional 42 features, i.e., 7 new features for each of the 6 input signals. The total generated features are 96, which is a high dimensional feature set.

Once the features are extracted, we note that some of the new features are on varying scales that could skew the modeling approach. Features based on Kurtosis of Band Power and Relative Spectral Peak per Band consist of values that are significantly higher than values of features based on Mean of Band Power and Variance of Band Power. Therefore, we choose to apply min–max normalization, a method shown in Equation (1). This ensures that all the features are on the same normalized scale.

3.4.2.4 Feature Evaluation and Selection

The feature set after extracting features from signals consists of a total of 103 features. Seven features are parameters of the experiment: case, run, VB, time, DOC, feed, and material. The other 96 features are extracted as discussed in the previous step. The data in their current state consist of 145 instances and 103 features. To ensure that the curse of dimensionality is avoided in the modeling phase, the most important features to predict VB and the RUL need to be identified. As a first step, we perform a univariate analysis by calculating the correlation coefficients between the individual features and the response variable VB. All 96 newly generated features are considered, ignoring the 7 experimental parameters since they are important and must be included in the analysis. The correlation coefficients are then ranked in descending order, allowing us to observe which of the newly generated features have a relationship with the response. Next, we calculate the correlation matrix for all 96 features and use the univariate ranking with VB to choose the best features. Feature pairs that have a Pearson correlation coefficient of 0.75 or more are considered for potential elimination by comparing their relationship with VB. From the feature pair, the one that has a higher correlation coefficient with VB is retained, while the other is dropped. This method allows us to identify the most important features without having to specify the number of features, which is a big research problem by itself. The final dataset consists of 34 features, out of which 27 features are extracted from the input signals, 6 of them are experimental parameters, and 1 is the response variable. The correlation matrix for the final set of features is shown in Figure 33 of the Appendix. As we can see, features that have a pairwise correlation coefficient of 0.75 or more are noted. The ranked correlation coefficients of these features with VB are compared, and features are removed accordingly. This is a simple, yet effective strategy to eliminate features that have no strength in the analysis.

3.4.3 Phase 3: Model Selection, Testing, and Evaluation

Before identifying the PHM tasks, we normalize the data using the min-max normalization method described earlier. All the variables associated with sensor values are normalized, while the original data is left untouched. The prepared data can now be used for data-driven tasks. Even though there are a very limited number of instances, i.e., 145 rows in the preprocessed dataset, it can be prepared for two important tasks. The first task of classification involves annotating the dataset with labels, and then using the labelled dataset for training and testing purposes to see how ML classifiers perform on numerical data consisting of sensor measurements. Regression analysis is the second task, and it comprises of identifying an attribute that can be used as the dependent variable for prediction. We deploy several ML models for both the tasks and evaluate them using some of the metrics discussed in Section 3.3.3.

3.4.3.1 Classification Task: Detecting Anomalies

The goal of this task is to determine whether the milling machine tool tip is safe to use or is degraded. The ISO 3685-1993 standard [167] states that the maximal flank wear for Titanium Carbide tools is 0.6 mm. This means that 0.6 mm of wear is considered the failure threshold for the tool tip. For instances in which VB is 0.6 mm or more, the tool is classified as 'degraded', else, it is classified as 'safe'. We annotate all the 145 instances from the data as either 'degraded' or 'safe' and the class distribution can be seen in Figure 12.

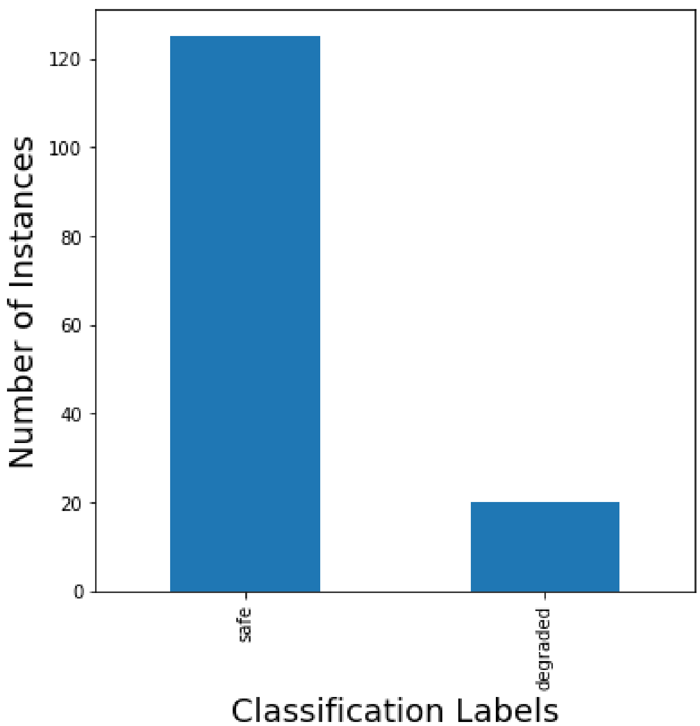

Figure 12. Class distribution of 'safe' and 'degraded' classes from milling data.

The dataset is split into train and test sets in a 70:30 ratio with random subsampling. There are a few factors that can be taken into consideration while selecting an ML model for classification. We need to estimate the performance of various models before selecting the best one. In order to select the best model, the optimal hyperparameters must first be identified. However, this can be a computationally expensive process. A method called grid search allows us to specify a list of hyperparameters and returns the model with the most optimal hyperparameters. This approach can be problematic too since the models can overfit the training data. To avoid this, we attempt to minimize the generalization error, i.e., how accurately or the model is able to classify unseen data points. We use a nested cross-validation approach [168] that has two objectives: hyperparameter tuning and performance estimation (see Figure 13)**.** It is crucial to point out that the data is split into training and test sets initially. The test set is left untouched, and the training data is used exclusively for the nested cross-validation approach. The training data is split for the outer and inner loops that use k-fold cross-validation, and each fold further splits the data into training and test folds. The outer-loop is used to estimate the generalization error, and the inner-loop is used for hyperparameter optimization. The test fold is not the same as the test data, which is set out to assess how the models perform in classifying 'safe' or 'degraded' instances. The test data is set out separately because even though nested cross-validation is typically good at estimating the final model performance, it can be unreliable in situations such as ours, where the dataset is small. For the inner-fold, we use a grid search with different values for hyperparameters. The optimal hyperparameters are then used in the outer test fold to find the average performance of the models.

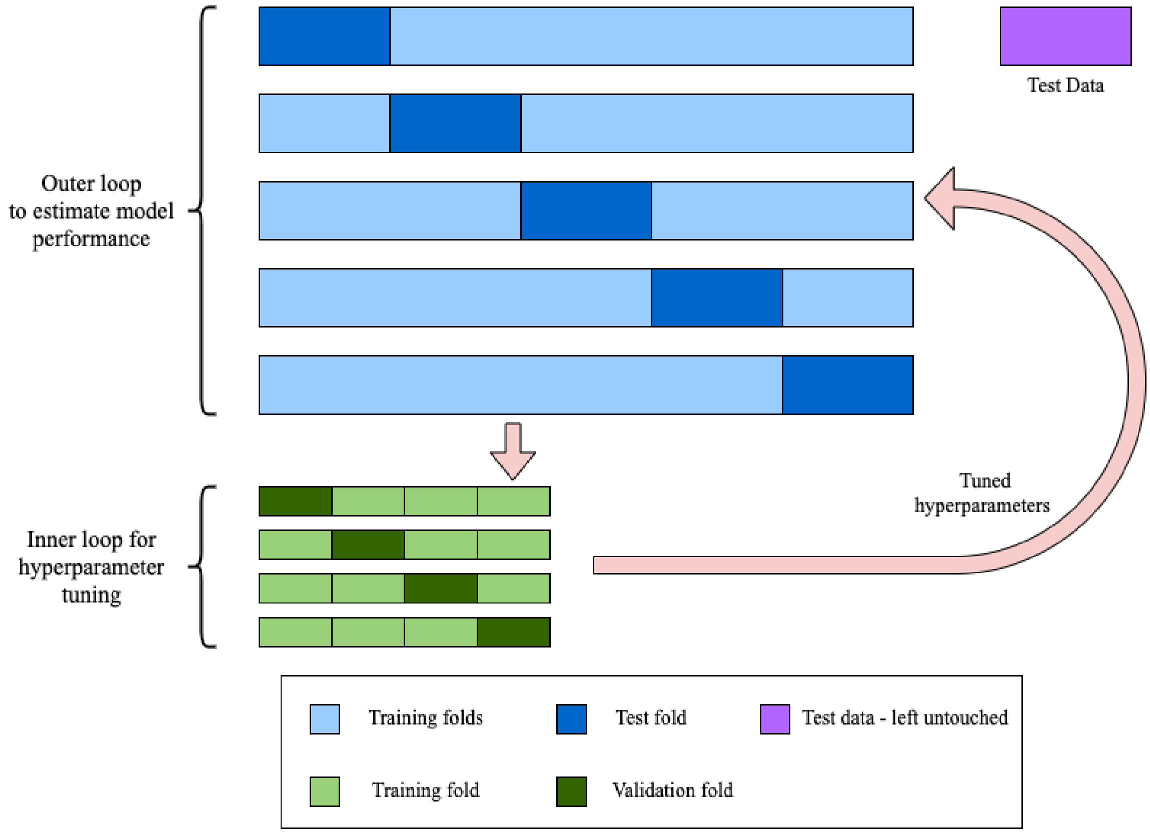

Figure 13. Nested cross-validation with 5 folds for outer-loop and 4-folds for inner loop respectively.

We use several classification models with the nested-cross validation method for the anomaly detection task:

1) Logistic Regression
2) Support Vector Machine
3) Multi-layer Perceptron
4) Decision Tree
5) Random Forest

Each of the above mentioned models are fed different hyperparameter values and solvers to find the most optimal hyperparameters for each of the models. The best parameters are selected and used to find the average performance on the outer test fold. These models are then deployed on unseen test data, which gives us a better idea of how the models would perform in the real-world.

3.4.3.2 Regression Task: Predicting Remaining Useful Life

RUL is a metric that is used to estimate the length of time that a tool, component, machine, or system is able to function for before it is to be repaired or replaced. It is a subjective metric, i.e., it depends on the application area, the particular tool, system, or component under consideration. RUL can be measured in days, years, number of cycles, etc., depending on the life of the tool. Figure 14 shows how the condition of a tool, component, machine, or system deteriorates with time, and highlights what its RUL is at a given state.

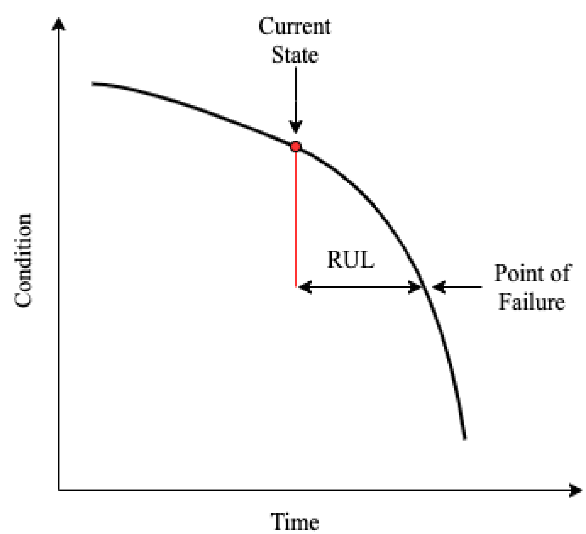

Figure 14. Remaining Useful Life of a machine, system, or component.

For the milling machine tooltip, we estimate RUL in terms of the remaining number of runs that it can operate for before it warrants replacement. For a particular ‘case’ of the milling experiment that has constant operating parameters, we can calculate RUL using the following formula as seen in Equation (8), where R_{max} is the run at which $VB_{max} = 0.6\ mm$ is met, and $R_{current}$ is the current run:

$$RUL = R_{max} - R_{current} \tag{8}$$

A few regression models are considered for the task of predicting RUL. A grid search is employed to find the optimal hyperparameters for the models. Models considered are:

1) Linear regression
2) Ridge regression
3) AdaBoost
4) Multi-layer Perceptron
5) Support Vector Regression

3.5 Results

3.5.1 Anomaly Detection

For the classification task of anomaly detection, we use accuracy, precision, recall, and F-1 score to determine the best performer. The Logistic Regression classifier achieves the best accuracy of 91%, followed by Support Vector classifier with an accuracy of 89%. More performance metrics of all the models is shown in Table 6, and the confusion matrices for each of the models are shown in Figure 34 of the Appendix. Although the precision or positive predictive value is high for most models, the recall values are relatively low. This means that even though the Logistic Regression model demonstrates high accuracy and precision, only 56% of the degraded tool tips were identified correctly. This makes sense, considering the fact that the test data only consisted of 44 instances overall. The performance should generally improve on larger datasets.

Table 6. Anomaly detection classification results

Model	Accuracy	Precision	Recall	F-1 Score
Logistic Regression	0.91	1	0.56	0.71
Support Vector	0.89	1	0.44	0.63
Multi-Layer Perceptron	0.84	0.62	0.56	0.59
Decision Tree	0.86	1	0.33	0.50
Random Forest	0.86	1	0.33	0.50

3.5.2 Remaining Useful Life Prediction

For the regression task of predicting the RUL, Root Mean Squared Error (RMSE), Mean Squared Error (MSE), Mean Absolute Error (MAE), R-Squared (R2) and Explained Variance metrics are used. The Explained Variance metric considers biased variance to compute what fraction of the variance is explained. R2 on the other hand uses the sums of squares in its computation. If the error from the predictor is unbiased, the R2 and Explained Variance scores will be the same. Therefore, it is interesting to observe if there is any biased variance involved from the predictor. The Multi-layer Perceptron produces the best R2 and Explained Variance scores, followed by the AdaBoost regressor, then Ridge regression. The Linear regression and Support Vector regression model underperform in our case. Table 7 shows the details of the regression model evaluation metrics, and Figure 35 of the Appendix shows the plots depicting predicted and actual values.

Table 7. RUL estimation results.

Model	RMSE	MSE	MAE	R2	Explained Variance
Linear Regression	2.41	5.84	1.88	0.76	0.80
Support Vector	2.80	7.85	2.16	0.68	0.72
Ridge Regression	2.17	4.74	1.87	0.81	0.83
AdaBoost	2.05	4.20	1.64	0.83	0.85
Multi-Layer Perceptron	1.55	2.40	1.29	0.90	0.90

3.6 Discussion

This chapter provides an in-depth understanding of PHM and maintenance methods in manufacturing. The different approaches and challenges to PHM are outlined, and the current trends in PHM research are reviewed. A framework for SPHM is proposed and is described in 3 phases—Phase 1: Setup and Data Acquisition, Phase 2: Data Preparation and Analysis and Phase 3: SPHM Modeling and Evaluation. The capabilities of the framework are demonstrated by applying it to a milling machine operation. The link between phases of SPHM and its application to the milling machine case are shown in Table 8. The milling machine experimental setup is reviewed, and the operating parameters are noted. Details on the acquisition of the data are listed along with a data dictionary consisting of the attributes and their description. The final dataset is cleaned of missing values and outliers. Signals are preprocessed, and suitable features are extracted based on prior knowledge and proven extraction methods. The final set of features is selected based on a comparison of pairwise correlation coefficients and the ranking of the correlation coefficients with the response variable, VB. Two ML tasks are identified – classification of anomalous instances and prediction of RUL. The results from these tasks reiterate the importance of the detail in planning and setting up the shopfloor for PHM.

Table 8. SPHM Phases implemented on milling data.

SPHM Phase	Steps	Implementation on Use-Case
Phase 1: Setup and Data Acquisition	• Shopfloor Setup	• Milling operation setup and sensors used reviewed
	• Data Collection and Understanding	• Dataset explored, features described, and preliminary investigation of data conducted
Phase 2: Data Preparation and Analysis	• Data Cleaning and Preprocessing	• Missing values identified and eliminated • Outliers visualized and removed
		• Feature Scaling
	• Signal Processing	• Signal Preprocessing steps: Amplification, filtering, RMS
	• Feature Extraction	• Features extracted in time domain and frequency domain
	• Feature Evaluation and Selection	• Correlation-based feature selection
Phase 3: SPHM Modelling and Evaluation	• Model Selection	• Classification and Regression Model Selection
	• Evaluation and Testing	• Anomaly detection and Remaining Useful Life Prediction
	• Deploy and Continuously Assess	• Future work

The SPHM framework demonstrates how PHM can be implemented to various systems and field devices on the shopfloor in a versatile manner. This work illustrates that every step involved, from planning of machine and sensor setup to the identification and evaluation of PHM tasks, is equally important while considering health monitoring of a manufacturing system.

Chapter 4

DL-based SPHM Framework to Support Manufacturing Product Health

This chapter proposes an AI-based Quality Inspection methodology that improves the visual inspection process. First, the role of health monitoring in manufacturing from the perspective of the system and the product is discussed. The current state of quality inspection and visual inspection in the industry is provided, and of some of the key factors that affect the inspection process are highlighted. A brief overview of the casting operation in manufacturing and the challenges presented in inspection of casting products are identified. Then, the role of AI in the quality inspection process is discussed, and a methodology for AI-based Smart Quality Inspection is introduced. The proposed methodology uses a custom Convolutional Neural Network (CNN) architecture that is deployed using a shop floor application. The proposed work not only improves the defect detection rate, but also minimizes some of the factors affecting the visual inspection process. The capabilities of the proposed work are demonstrated by testing it on image data from visual inspection of a casting process.

4.1 Introduction

Over the years, computer vision-based systems have been incorporated into the inspection process of various products such as disk heads, steel strips [169], syringes [170] and semiconductors [171]. Vision-based systems generally consist of an algorithm that is taught to identify discrepancies between features of the product undergoing inspection and the desired features. Although these systems help in automating the inspection process to a certain extent, there are still some challenges to their implementation on the shop floor [172]. There are also numerous works in which ML and DL-based models are applied to the quality inspection process. However, a lot of researchers focus on improving the performance of models and do not consider a holistic approach to inspection. While that is one of the main goals of the inspection process, there are several factors affecting the inspection process that go unaddressed. There is also a need for a methodology or approach that establishes how a data-driven method can be deployed on the shop floor in a user-friendly and hassle-free manner. We propose a DL based approach to address the requirements of visual inspection that improves defect detection rate while at the same time minimizing the effect of the factors affecting the inspection process.

4.2 Health Monitoring in Manufacturing

It would be remiss to discuss quality inspection, a crucial component of the QC process, without providing a context about the health of the entire manufacturing environment. There are two major areas of health monitoring in manufacturing. The first concerns the system's health, ensuring that the machinery and equipment are functioning satisfactorily. The second area pertains to monitoring the product's health throughout its lifecycle. For system health monitoring, PHM deals with component-level and machine-level health monitoring. On the product side, QC techniques are relied upon to guarantee the health and quality of the product. Figure 15 portrays the different phases of PHM and QC in the context of health monitoring in manufacturing. The rest of this section presents brief overviews of these topics.

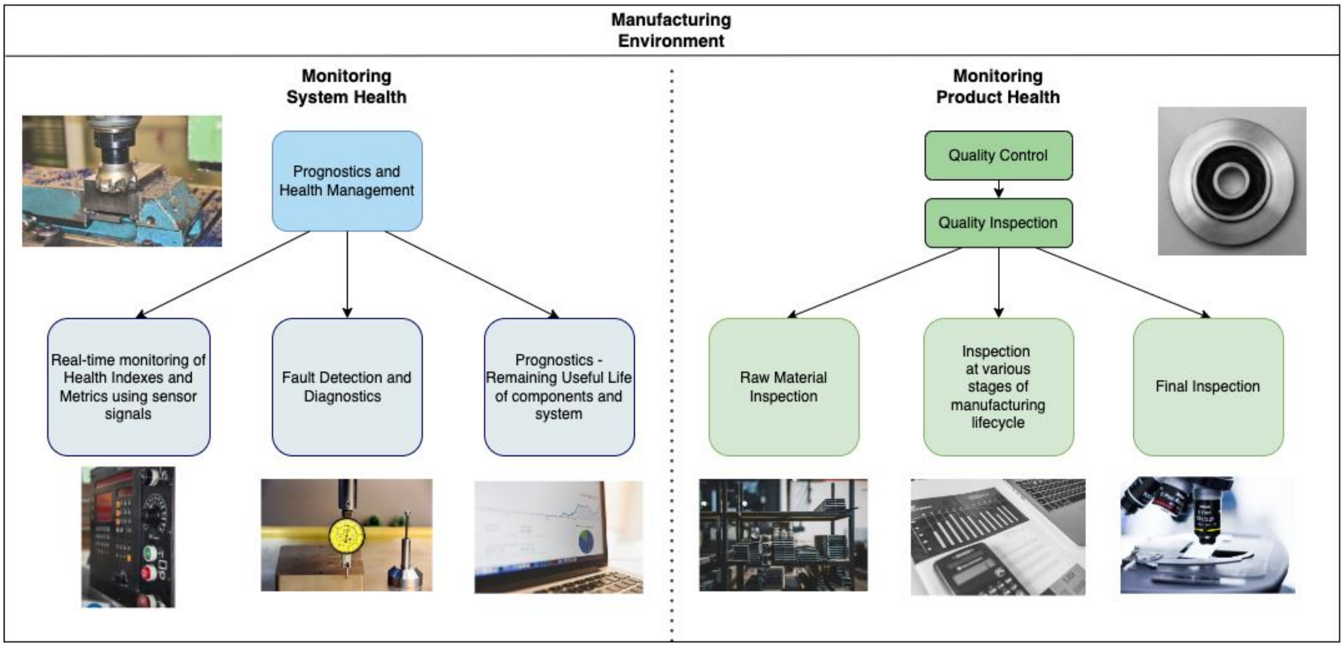

Figure 15. Health Monitoring of the manufacturing environment with PHM and QC.

4.2.1 Prognostics and Health Management (PHM)

Prognostics and Health Management (PHM) is a discipline that monitors the system's health, detects failures, diagnoses failures, and predicts the Remaining Useful Life (RUL) of components [173]. Using the Internet of Things (IoT) -powered sensors and field devices, operating conditions of critical tools and components can be monitored in real-time. Nowadays, the availability of low-cost embedded devices and microcontrollers such as Raspberry Pi, STM32, Arduino, etc. enable SMEs to incorporate PHM on the shop floor. Once sufficient data has been collected, the health indexes and metrics developed can be used in models to predict failures of components and provide Remaining Useful Life (RUL) estimates. In the past, data-driven approaches to PHM have relied on ML models. However, successful implementations of ML models for prognostics or failure detection are often reliant upon expert knowledge to extract meaningful characteristics or features from the data [174]. DL techniques have the capabilities to automatically extract high-level features from inputs such as acoustic signals, vibration signals, image data, etc. Hence, there is an advantage to using DL for prognostics and diagnostics applications. A detailed methodology developed in [173] reviews the various approaches to PHM: data-driven, physics-based, and hybrid and use-case on health monitoring of a milling machine tool.

4.2.2 Quality Control (QC)

With the development of highly complex products, quality management has taken an essential role in organizational planning and strategies [175]. Quality Control (QC) is a process that involves setting quality standards and ensuring that the product meets those standards. Quality Inspection is a part of the QC process in which the product is inspected by operators during the various stages of manufacturing. Figure 15 depicts the QC process from the perspective of health monitoring, which involves product inspection. The overall QC process is continuously changing due to the dynamic nature of the manufacturing environment. Approaches such as Design of Experiments (DoE), Failure Mode and Effects Analysis (FMEA), Quality Function Deployment (QFD) and Acceptance Sampling prescribe their own methodologies for product inspection. While these approaches have been very successful in QC, there is an opportunity to assess 100% of the products on the modern shop floor. Continuous assessment of product quality has become a reality with developments in sensor networks and AI.

4.3 Overview of Quality Inspection

4.3.1 Quality Inspection Process

The traditional quality improvement process is cyclical—it involves generating inspection plans, implementing the plans, and checking the results [176]. Similarly, the inspection process is comprised of inspection plans that identify the different areas of manufacturing where inspection is required. It typically begins with the inspection of raw materials—also known as incoming or receiving inspection. Then there are inspections conducted periodically after various operations. The nature of these inspections is industry-specific in most cases. For instance, the inspection of structural steel products would differ greatly from the inspection of microcontrollers. At the end of the assembly line, a final inspection is conducted—where it is determined whether the product is acceptable or is to be rejected. This is analogous to outgoing inspection. In some cases, outgoing inspection refers to the inspection of the packaged product during shipping.

The inspection process is an important decision process in the manufacturing/production system [177]. According to the Signal Detection Theory (SDT), probabilistic decisions are made at every step by the decision maker (operator) to determine whether the product is to be accepted or rejected [178]. Inspection is not only an independent process in the manufacturing value chain, but it also impacts many other operations. The decision-making process for inspection involves multiple elements and should display the following characteristics as noted by [177]:

1. Precision: The decisions made should be well-informed, to ensure that there are no biases or errors.

2. Validity: Decisions made must be valid and must not differ if the product were to be available for use.

3. Reliability: There must be consistency in the decisions made—repeatability and reproducibility. The decision process should not require recalibration.

4. Robustness: The decision-making must demonstrate versatility in detecting different types of defects.

5. Rapidness: The process must be quick and must be able to act before any more defective products are produced.

Note that the above characteristics are desired from all inspection processes regardless of whether it is conducted by either human operators or by some form of automation.

4.3.2 Visual Inspection

An important type of quality inspection in manufacturing is visual inspection. Operators visually assess the state of the product at different stages and decide whether it can be moved on to the next process. Sinclair [179] suggested a four-step visual inspection operation comprising the following tasks:

1. Present: Present the product for inspection.

2. Search: Examine and analyze the product for possible flaws/defects.

3. Decision: Assess the flaws/defects and determine if it falls out of the desired specifications.

4. Action: Accept or reject the item based on the decision.

Similarly, Wang and Drury [180] characterized the visual inspection process as having a number of sub-tasks or activities: (1) orient the item, (2) search the item, (3) detect the defects/flaws, (4) recognize and classify the flaws/defects, (5) make a decision about the item, (6) dispatch the item, and (7) record any information about the item. In both approaches, the goal of the visual inspection process is to identify defects efficiently and accurately and make decisions accordingly.

4.3.3 Factors Affecting Visual Inspection

Any inspection process or system requires some form of human action. There cannot be a system that is entirely automated or manual [177]. Inspection involves a lot of mental effort, attention to detail, communication, and the usage of long-term and short-term memory [181]. In most cases, inspection is also required to be done quickly, i.e., defects must be identified swiftly before a decision is made. With human involvement, there arise several factors that could affect or impede the efficient implementation of visual inspection. According to research conducted by Peters et al. [182] and See et al. [183], some of the known factors that impact inspection can be categorized into task factors, environmental actors, operator or individual factors, organizational factors, and social factors.

Task factors refer to the manual and physical aspects of the inspection task. The task itself can affect the operator and influence their performance. Environmental factors can also significantly impact the outcome of visual inspection. Factors such as temperature, humidity, lighting, etc. can make the environment unsuitable which in turn influences the operator’s ability to conduct the inspection. Operator or individual factors refer to features such as an operator’s physical and mental attributes. Physical attributes could be an operator’s vision, visual acuity, gender, etc. Mental attributes could be their state of mind, aptitude, personality, biases, etc. Organizational factors concern the administration and management under which the inspection process takes place. It also includes the organizational importance given to quality inspection and visual inspection, training provided, etc. Social factors include relationships that the operator has with their peers and management, whether communication in their working environment is effective or not, and the other aspects of the social environment in which the inspection task occurs. A synopsis of all the factors that affect the visual inspection process based on [182], [183] is provided in Table 9.

Table 9. Factors that affect visual inspection.

Task	Environmental	Operator or Individual	Organizational	Social
▪ Defect Rate ▪ Type of defect ▪ Defect detectability ▪ Location of defect ▪ Complexity of task ▪ Standards for comparison ▪ Time available to complete task ▪ Multiple inspections for each task ▪ Inspection aids ▪ Level of automation	▪ Temperature ▪ Humidity ▪ Lighting ▪ Noise ▪ Time of the day ▪ Duration of shifts ▪ Workplace ergonomics ▪ Alertness or level of vigilance	▪ Age ▪ Intellectual Aptitude ▪ Level of intelligence ▪ Gender ▪ Visual acuity ▪ Depth perception ▪ Concentration level ▪ Biases	▪ Support from management ▪ Training and retraining ▪ Incentives, bonuses ▪ Feedback on performance ▪ Job rotation	▪ Relationship with peers ▪ Communication ▪ Isolation ▪ Pressure

4.4 Casting Process

The manufacturing process of casting usually involves pouring liquefied metal into the cavity of a mold that is of the desired shape [184]. There are different types of casting processes in manufacturing. The type of process is dependent on the materials (mostly metals) used to manufacture the final product.

4.4.1 Types of Casting Processes

Some of the types of casting processes along with the types of materials used are listed blow [185]:

(a) Sand Casting—most metal types

(b) Investment Casting—most metal types

(c) Resin Shell Molding—Primarily Iron and Copper

(d) Gravity Die Casting—Primarily Aluminum, Zinc, Magnesium, Copper, and some of their alloys

(e) Low-Pressure Die Casting—Primarily Aluminum and Magnesium

(f) High-Pressure Die Casting—Primarily Aluminum, Magnesium, and Zinc

(g) Squeeze Casting—Primarily Aluminum

4.4.2 Steps in the Casting Process

Most of the aforementioned casting processes generally follow similar steps to go from the raw material to the finished product. A list of these steps is as follows [184]:

1. Patternmaking—Designing and preparing a pattern
2. Preparing the mold that is approximately the same shape/size as the desired pattern
3. Identifying the material to be used in casting (usually metals or allows)
4. Liquefying the material in a furnace
5. Pouring the liquefied metal into the cavity of the mold
6. Opening the mold to access the casting
7. Fettling—removing excess material, surface cleaning, and finishing
8. Heat treatment based on requirements
9. Final inspection

4.4.3 Inspection in the Casting Process

The quality inspection of casting products is the most critical step in determining whether the product is acceptable for use or must be rejected and scrapped/reworked. There are a few types of inspections for casting products: visual inspection, dimensional inspection, metallurgical inspection, chemical and physical inspection, and other methods involving Non-Destructive Testing (NDT) [186]. While multiple inspection methods are used concurrently during different stages of the casting process, we shall limit the discussion to a brief on visual inspection methods.

This visual inspection process for casting generally involves the examination of the product by an operator or a group of operators. Operators look for surface defects, cracks, tears, molding flaws, scabs, blowholes, runouts, adhesions, and various other types of defects [187]. Many of the defects can be attributed to flaws in mold design, the incorrect composition of materials used in mold construction, the equipment used in pouring liquefied metals into the molds, etc. Some visual inspection processes for casting products can be automated. Vision-based inspection systems rely on software for color matching and in some instances contour matching and dimension checking [188]. In recent years, ML and DL techniques have been used to perform visual inspections of casting products.

4.5 Deep Learning for Quality Inspection

Owing to the numerous factors that can affect an operator during the visual inspection process, data-driven approaches are being used increasingly to detect defective products. While traditional ML methods often require domain knowledge in the feature generation or feature engineering process, DL methods can automatically select and learn abstract features [189]. DL methods based on CNN, Autoencoders, and Recurrent Neural Networks (RNN) provide excellent results on a variety of inspection applications [190].

Chang et al. [191] apply a deep ensemble learning model to inspect defects on car body surfaces. Their method outperforms human inspectors in performing the same task. Researchers in [192] use a CNN to identify defects in textured surfaces. Results show high accuracy of defect detection on a multi-class dataset. For defect identification in semiconductor manufacturing, Imoto et al. [193] use a transfer learning approach based on

CNN while Lee et al. [194] propose a CNN model that is receptive to time series data. In the inspection of sewer systems, Kumar et al. [195] propose deep CNNs. They use image data with high variation and claim that the CNN-based methods outperform other methods requiring manual feature extraction. For the inspection of laser welding, Yang et al. [196] use an optimized VGG model. A transfer learning approach is used where the VGG model is pre-trained on a large variety of images. Ullah et al. [197] propose an approach that uses a pre-trained AlexNet for feature extraction combined with Random Forest (RF) and Support Vector Machines (SVM) for defect detection. Their proposed method outperforms LeNet and VGG algorithms in an experiment conducted on high voltage electrical equipment. To inspect rivet joints in aircraft products, Amosov et al. [198] apply YOLOv5 and MobileNetv3 to images. In binary classification and in the multi-class scenario, they achieve very high accuracy of defect detection. To inspect the gas lighter manufacturing process, researchers in [199] develop a DL model based on YOLOv4. Results show good performance in detecting defects with changing illuminance and distance.

4.6 Smart Quality Inspection

The Smart Quality Inspection (SQI) approach aims to enhance the inspection process by improving the defect detection rate and addressing several factors that affect the visual inspection process. By automating the inspection process to an extent, the effects of many of the task factors, environmental factors, and individual factors can be controlled. Figure 16 displays a flowchart that depicts the process used to develop SQI. The research gaps identified via the literature review process tie in directly with the development of the inspection algorithm. The approach outlined to develop SQI is informative in proposing the methodology to implement AI-based visual inspection on the shopfloor. Figure 17 shows the different stages involved in implementing SQI in the manufacturing/production area. There are a total of six stages—from receiving the product at the inspection area to inspecting it using AI and documenting the results. The processes and steps involved in each of the stages are described as follows.

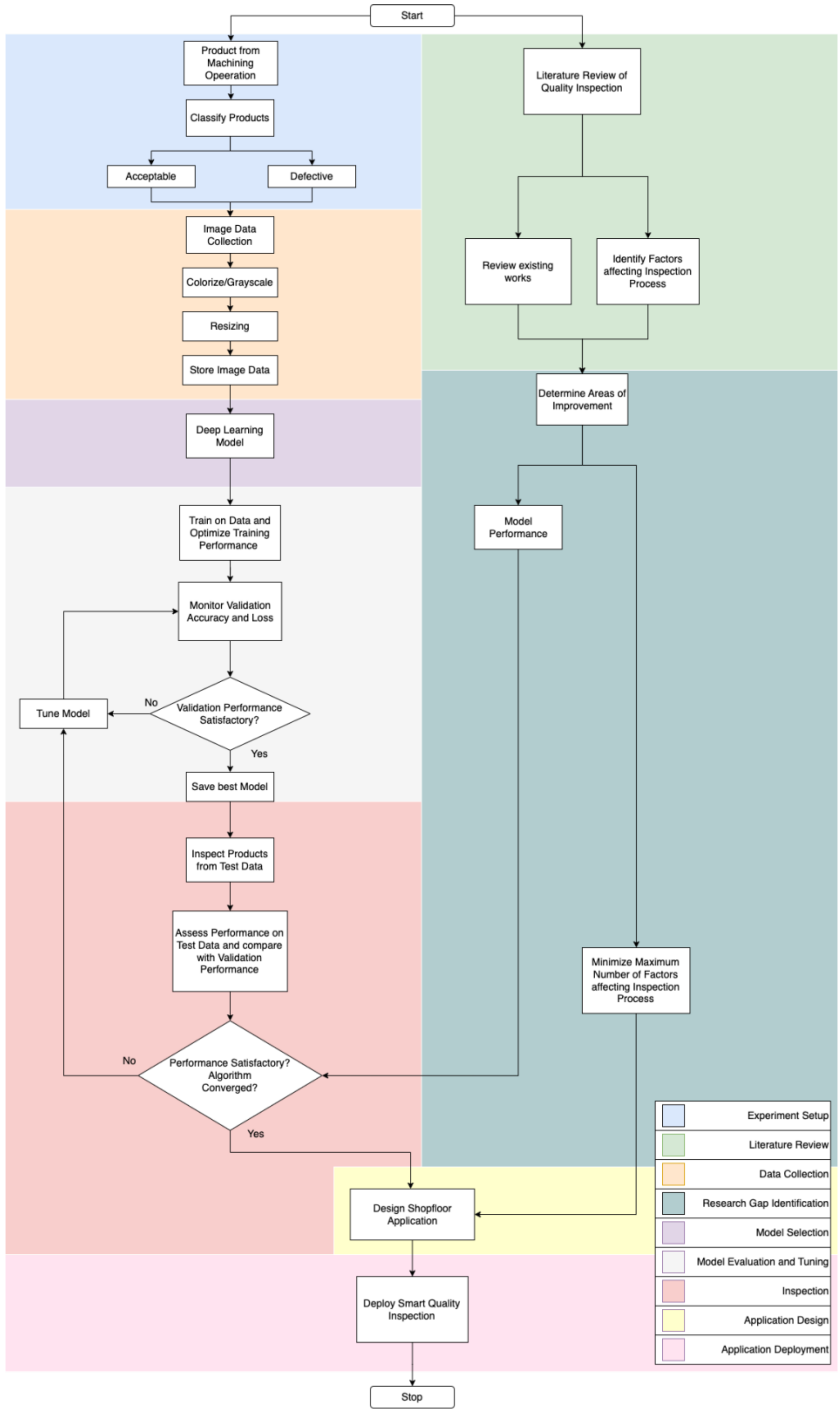

Figure 16. Flowchart depicting the process used to develop Smart Quality Inspection—algorithm construction and identification of research gap via literature review.

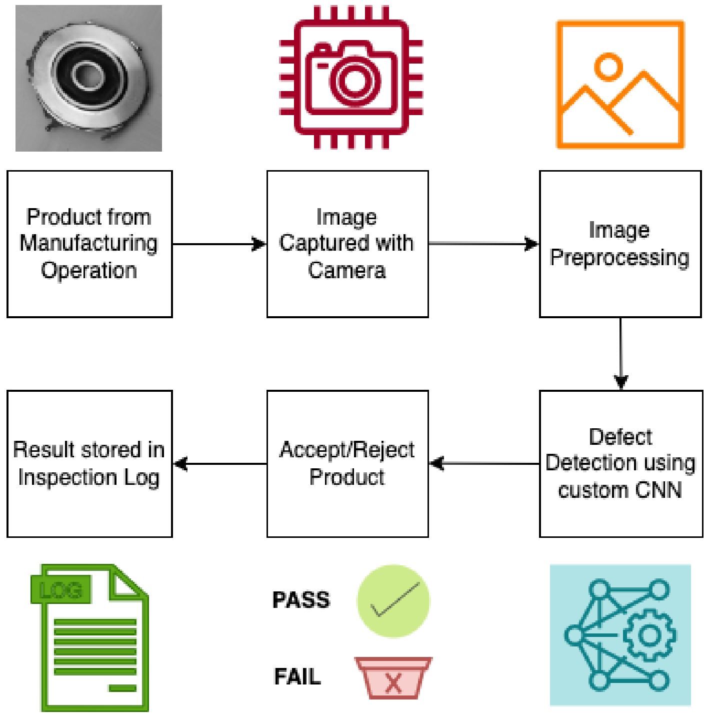

Figure 17. Artificial Intelligence based Smart Quality Inspection Methodology.

1. Stage 1: *Manufacturing product arrives at the inspection area.*

   In the first stage, the product from the assembly line is brought to the inspection area. The item is placed in a designated location to allow the inspection process to begin.

2. Stage 2: *Product image is captured.*

   In this stage, a high-quality camera is used to capture images of the product undergoing inspection. The lighting conditions and distance from the product are determined based on the product size and camera equipment in use.

3. Stage 3: *Image preprocessing.*

   It is established whether grayscale or color images are appropriate based on the availability of computational resources, and desired precision and accuracy of predictions. Any augmentation or transformation is done at this stage—flips, shears, rotation, shifts, whitening, contrast adjustment, etc.

4. Stage 4: *CNN-based defect detection.*

   A custom CNN architecture is used to detect defects in images. The architecture has the versatility to handle different types of images with just a small number of changes. The model is trained on images of defective products and non-defective products to learn the necessary feature representations. The defect detection model is built into an application that can be used on the shop floor to make the inspection process trouble-free.

5. Stage 5: *Decision stage—accept/reject the product.*

   The operator inspects the product using the defect detection algorithm and instantaneously receives the inspection results from the computer application. Based on the results, a decision is made whether to accept or reject the product.

6. Stage 6: *Document results in the inspection log.*

   The results of the inspection process are input into the SQI shop floor application and are automatically stored in a spreadsheet.

4.7 Casting Product Inspection

The dataset used in this work is obtained from Pilot Technocast, an SME that manufactures casting products in Gujarat, India. The data has been made available publicly by Ravirajsinh Dabhi [200]. The data consists of 7348 top-view images of submersible pump impellers. The products are made with stainless steel in a shell molding casting process. The images are captured under stable lighting using a Canon EOS 1300D camera kit produced by Canon Inc. located in Tokyo, Japan. Each image is converted to a size of 300 × 300 pixels. The data is pre-labeled into two classes: 'def_front' and 'ok_front', meaning defective and acceptable, respectively. A sample of six images that show three defective three acceptable castings are shown in Figure 18.

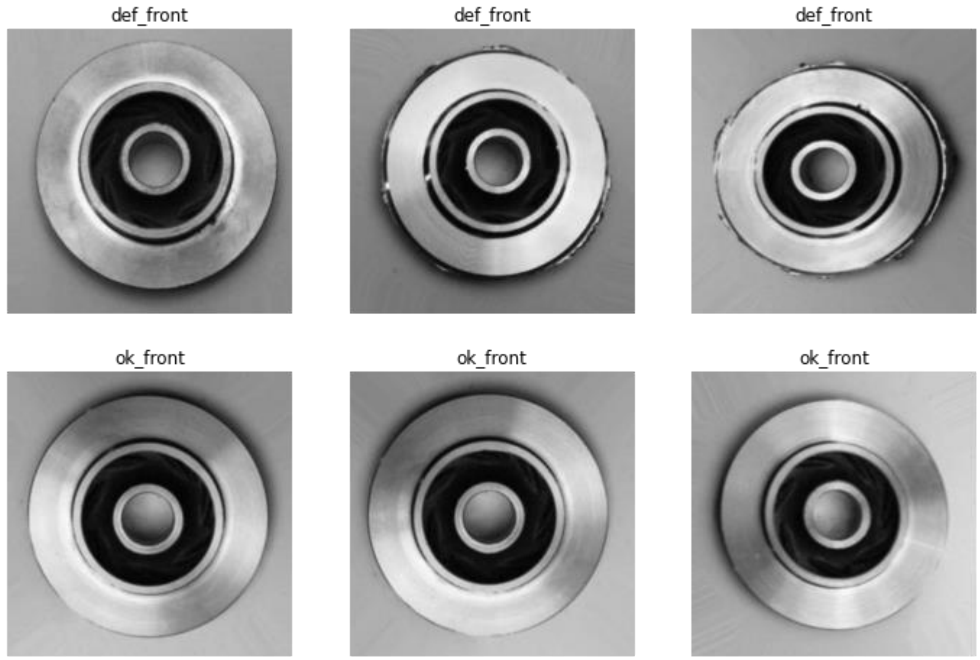

Figure 18. Sample images showing ‘defective’ and ‘okay’ stainless steel castings.

4.8 SQI – Modelling and Design

4.8.1 CNN Model

The AI model developed for the inspection task is a CNN with a custom architecture. The network is constructed using a set of Convolutional Layers (Conv2D), Max Pooling Layers (MaxPooling2D), Activation Functions, and Dense Layers. The convolution operation performed by the Conv2D layer is a dot product of the ‘kernel’ and the image. The MaxPooling2D layer creates a pooled feature map by reducing the parameters involved. The max pooling layer consists of a technique called zero-padding that involves adding zeros to the edges of the image before the next convolution operation. This preserves any features that are generated at the edges of the images. In the proposed model, we use a kernel of size 3×3 for the convolutional layer and zero-padding in the max pooling layer. For the dense layer, we use the Rectified Linear Unit (ReLU) activation function [201]. Equation (9) shows the ReLU function where x is the input to the neuron.

$$f(x) = \max(0, x) = \begin{cases} x, & \text{if} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

The model also uses the Adaptive Moment Estimation (Adam) optimizer [202] and the sparse categorical cross-entropy loss function from Keras [203]. A summary of the model generated by the Keras library is shown in Table 10 and a visualization of the model architecture is shown in Figure 19.

Model: “sequential”

Table 10. Keras model summary.

Layer (type)	**Output Shape**	**Param #**
rescaling (Rescaling)	(None, 300, 300, 3)	0
conv2d (Conv2D)	(None, 300, 300, 16)	448
max_pooling2d (MaxPooling2D)	(None, 150, 150, 16)	0
conv2d_1 (Conv2D)	(None, 150, 150, 32)	4640
max_pooling2d_1 (MaxPooling2D)	(None, 75, 75, 32)	0
conv2d_2 (Conv2D)	(None, 75, 75, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 37, 37, 64)	0
flatten (Flatten)	(None, 87616)	0
dense (Dense)	(None, 128)	11214976
dense_1 (Dense)	(None, 2)	258

Total params: 11,238,818

Trainable params: 11,238,818

Non-trainable params: 0

A brief explanation of the different layers of the model is provided below:

1. Input Layer: The input layer is the raw image either in grayscale or Red-Green-Blue (RGB) format with (300, 300) as its dimensions. This 300 × 300 image is an array of pixels, with 300 as width and 300 as height.

2. Rescaling Operation: Neural networks generally perform better when the inputs are normalized. The channel coefficients for images are in the [0, 255] range, which is high. Higher numeric values may be computationally more expensive and could

affect performance. For the casting data, we rescale the inputs to the [0, 1] range by using a 1/255 scaling factor.

3. Convolution Layer (conv2d): The first of the three convolution layers has 448 parameters with the data as (300, 300, 3) shaped array.
4. Max-Pooling (max_pooling2d): The pooling layer is useful in reducing the number of dimensions of the data. Pooling not only reduces the consumption of computing resources but also improves overall performance [204]. Max-pooling helps optimize the feature space by identifying the maximum value of elements from every pool, thereby achieving scale invariance [205].
5. Convolution Layer (conv2d_1): The second convolution layer has 4640 parameters
6. Max-Pooling (max_pooling2d_1): Like the max_pooling2d, this layer is aimed at optimizing the feature space from (150, 150, 32) to (75, 75, 32).
7. Convolution Layer (conv2d_2): The third convolution layer has 18496 parameters with an input shape of (75, 75, 64).
8. Max-Pooling (max_pooling2d_2): This max-pooling layer further reduces the dimensions of the feature map from (75, 75, 64) to (37, 37, 64) by selecting the maximum value of elements from every pool.
9. Flatten Layer (flatten): The pooled feature map is transformed from 3 dimensions to a 1-dimensional vector. This layer essentially collapses all the input into a single dimension.
10. Dense Layer (dense and dense_1): The dense and dense_1 layers from the model are geared towards the classification task. In general, a dense layer is a fully connected layer—every input and output neuron have a connection. The dense layer uses a ReLU [201] activation function and the dense_1 layer is designed with a number of output nodes equal to the number of classes.

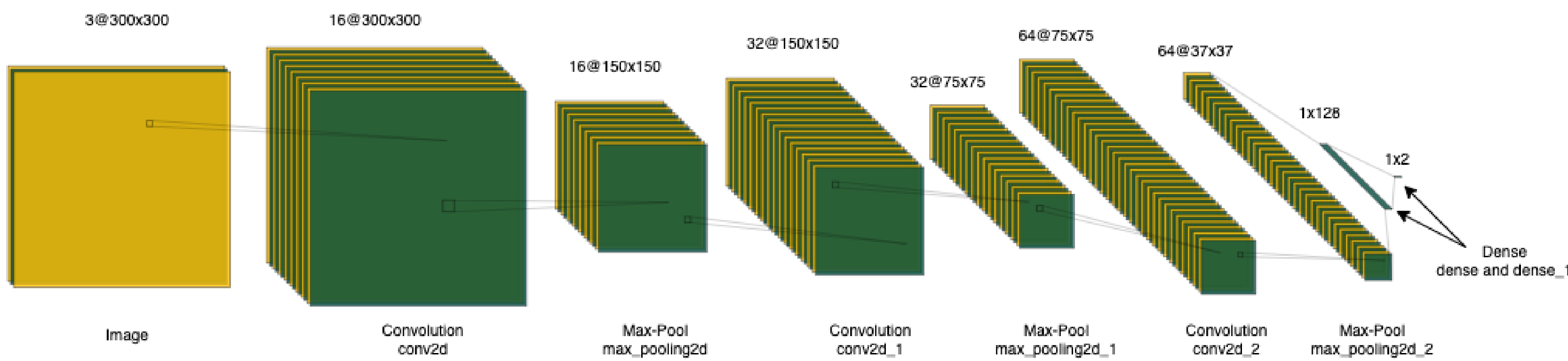

Figure 19. CNN architecture for Smart Quality Inspection.

4.8.2 Shopfloor Application

The aim of this application is to enable a hassle-free inspection process on the shop floor. Images of the product from the assembly line can be uploaded into the application, and the CNN model will inspect the product to determine whether it is defective or acceptable, thereby failing or passing inspection respectively. Additionally, the SQI application allows the operator to document the findings in the inspection log. Information such as product identifiers, machine identifiers, the result of the inspection, and additional remarks can be stored in the inspection log. Figure 20 shows the SQI application window.

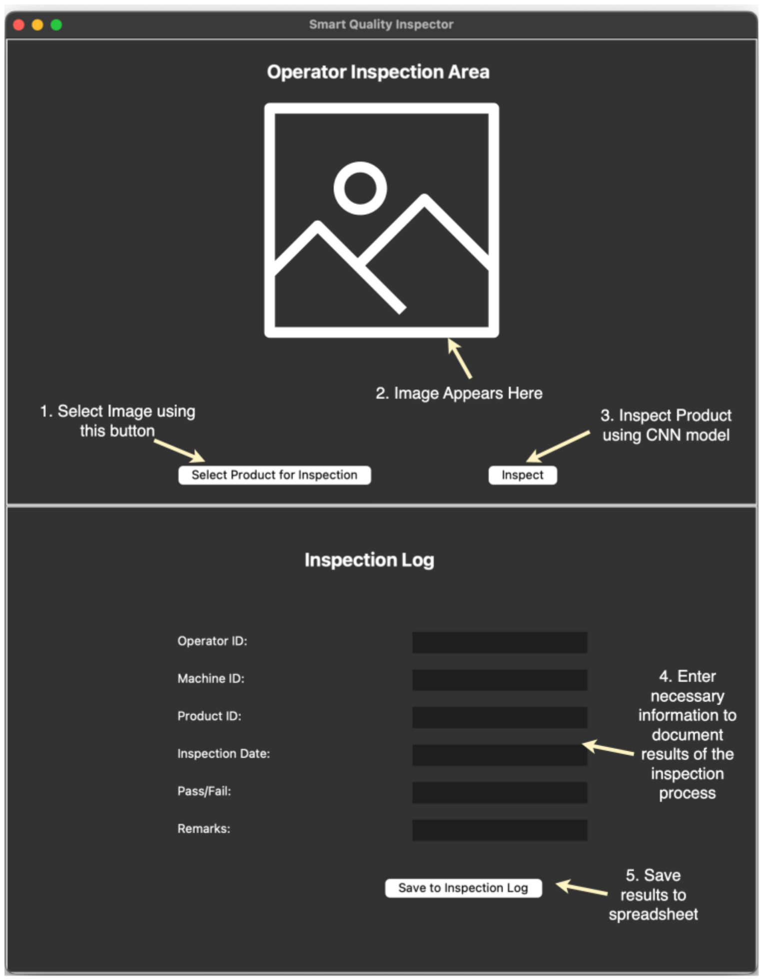

Figure 20. Shop floor application for Smart Quality Inspection.

4.9 Performance and Results

Stages 1–3 of the SQI method involve receiving the product, capturing product images, and preprocessing images. The casting data considered in our work consists of images that were captured under stable lighting and with a Canon EOS 1300D camera [200]. Additionally, some augmentations were already applied to the image data—shear, crop, contrast adjustment, etc. The only pre-processing necessary was rescaling the images before applying the CNN model.

4.9.1 Training and Performance Validation

The training dataset consists of 6633 images, out of which 5307 images were used purely for training the model and the rest 1326 files were used in the validation set to tune the model. To optimize the performance of the model and usage of computational resources, we use the 'Autotune' option from the Tensorflow library. As noted previously, the ReLU activation functions were used in the Conv2D and Dense layers, and the loss function considered was a sparse-categorical cross-entropy loss. The Adam optimizer was used to compile the model, but it is worth mentioning that other optimization methods were also experimented with. The Root Mean Squared Propagation (RMSProp) and Nesterov-accelerated Adaptive Moment Estimation (Nadam) were also explored but the Adam optimizer narrowly outperformed them on the casting data. The training phase was set to run for a maximum of 20 epochs but was completed in 13 epochs. This is due to the inclusion of the early stopping criteria as a safeguard against overfitting [206]. We use validation loss as the criteria for early stopping, and the execution of the model is interrupted when the validation loss does not improve. The training and validation accuracy along with the training and validation losses have been monitored at the end of each epoch. The plot in Figure 21 shows how the accuracies and the losses of the training and validation set change by epoch.

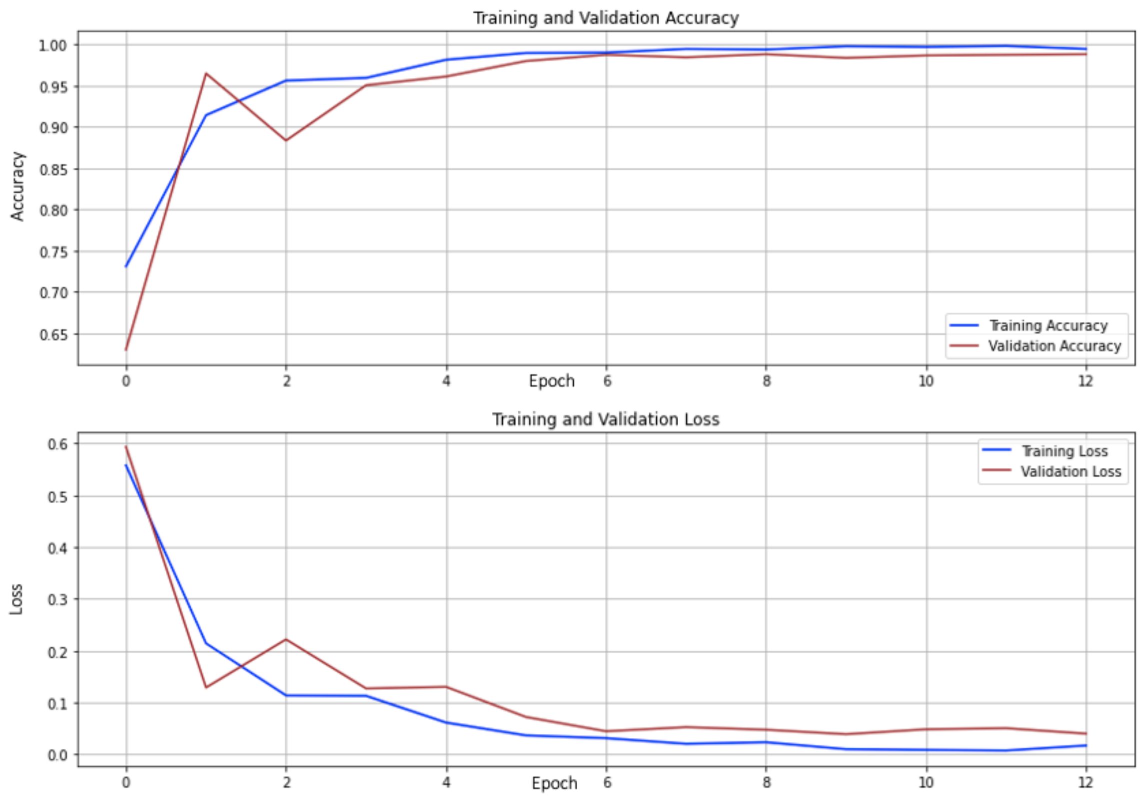

Figure 21. Monitoring the accuracy and loss of the training and validation set.

4.9.2 Testing Results

The performance of the model is evaluated on test data of 715 images. Overall, the model achieves an accuracy of 99.86% and outperforms all the other existing models from published works. Table 11 compares the performance metrics of SQI's proposed model with other models.

Table 11. Comparison of the performance of our proposed model with other models from published works.

Model	Precision	Recall	F1 Score	Accuracy
CNN with Densenet [207]	99.08%	100%	99.54%	99.42%
EfficientNetB0 [208]	97.11%	95.87%	-	96.88%
CNN-based Vision System [209]	-	-	-	99.7%
Transfer Learning with DenseNet [210]	97.96%	95.58%	-	95.94%
CNN model for Holonic Shop Floor [37]	-	-	-	99.82%
VGG-16 with CNN [211]	98.7%	94.1%	-	95.8%
Vision Transformer [212]	99.66%	99.33%	-	99.58%
Accelerated CNN [213]	99.24%	100 %	99.62%	99.72%
Proposed Smart Quality Inspection (SQI) Model	**99.62%**	**100%**	**99.81%**	**99.86%**

We can also note that there was only one product that erroneously failed inspection when it was actually an acceptable product. If we look at the confusion matrix in Figure 22, we can see the results of the inspection on 715 images of the test set. There are 261 images that are correctly labelled as 'OK' (acceptable) and 453 images that are correctly labelled as 'DEFECTIVE' (rejected). These are the True Positive and True Negative values, respectively. On the other hand, one image that has a true label of 'OK' has been incorrectly classified as 'DEFECTIVE', resulting in one False Positive value. No images of 'DEFECTIVE' products were incorrectly classified—meaning no False Negative values.

Figure 22. Confusion matrix showing the results of the inspection process.

In the case of quality inspection, False Positives are regarded as Producer's risk and False Negatives are regarded as Consumer's risk. Producer's risk is the error of rejecting a good-quality product, and Consumer's risk is the error of accepting a bad-quality product. In manufacturing, the aim is to reduce or minimize consumer's risk while producer's risk is acceptable to some extent. Based on the evaluation of 715 images, our proposed Smart Quality Inspection approach shows that there is no risk to the consumer, i.e., no defective products have been incorrectly accepted by the model.

4.9.3 Results from Shopfloor Application

Using the shop floor application for SQI, we can inspect the casting products. To demonstrate the application's functionalities, we inspect a defective product and an acceptable product. Figure 23a,b shows the results of the inspection of a defective product and an acceptable product, respectively. With the literal click of a button, the product is inspected. The operator is then able to document the inspection process by entering identifying information related to the product, machinery, etc. The information entered is saved in an inspection log in the form of a spreadsheet (see Figure 23c).

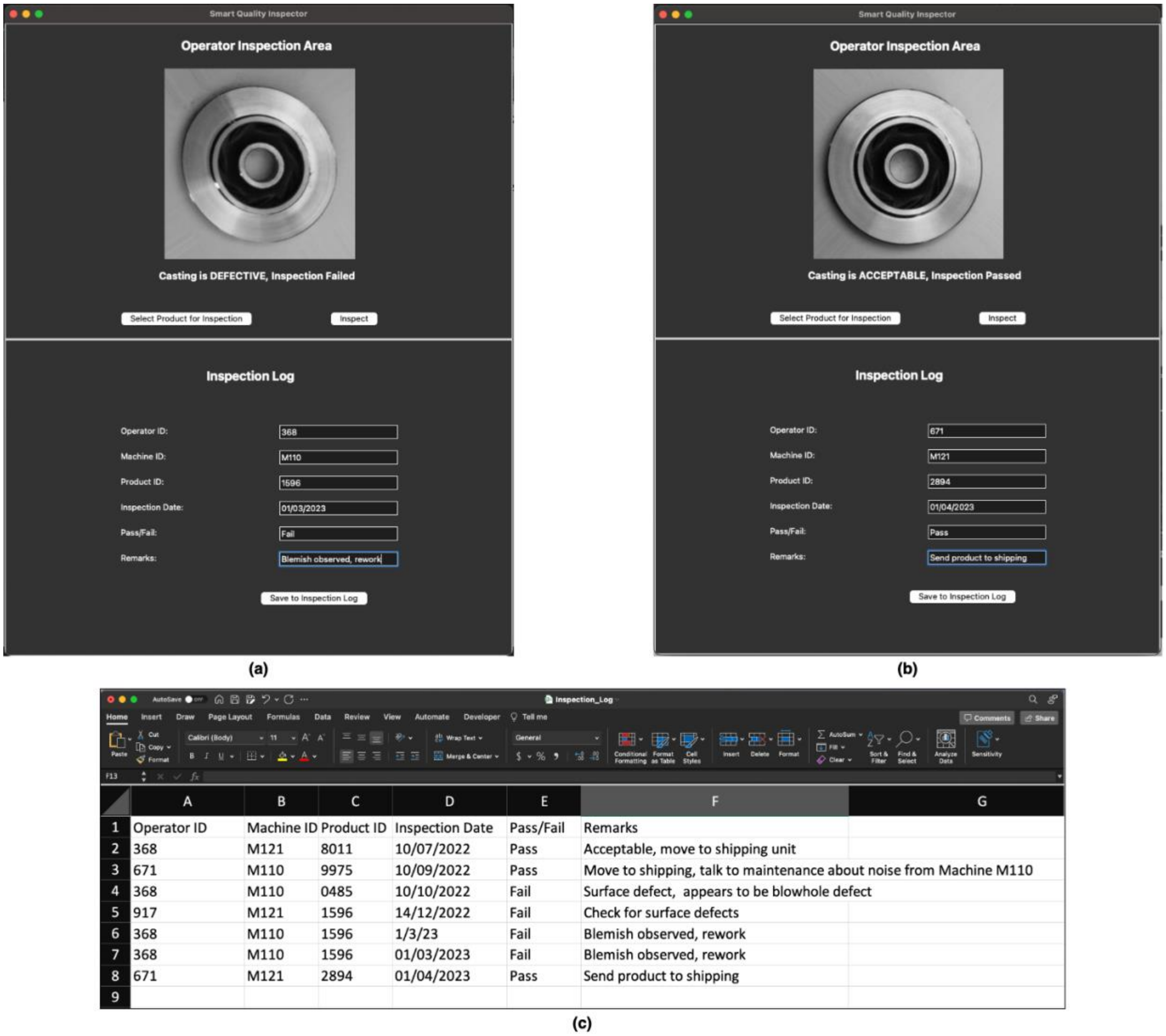

Figure 23. Shop floor application for Smart Quality Inspection. (a) shows the inspection of a defective product, (b) shows the inspection of an acceptable product, and (c) shows the results of the inspection documented in the inspection log (spreadsheet opened in Microsoft Excel).

4.10 Discussion

This chapter addresses the area of product health monitoring from the perspective of the quality inspection process. The monitoring of system health and product health in manufacturing are reviewed. The steps involved in the quality inspection and visual inspection process are discussed, and key factors that affect the visual inspection process are analyzed. The casting operation is reviewed and the process of visual inspection of casting products is examined. Based on the challenges involved in visual inspection, the

AI-based approach of Smart Quality Inspection (SQI) is proposed. A custom CNN model for SQI is designed and implemented on casting product images. The model achieves a high accuracy of 99.86% in inspecting casting products. The accuracy and F-1 score for the model are the highest compared to all the published works on the dataset. Additionally, a shop floor application is also developed to make the inspection process hassle-free. The goal of the application is to minimize the effect of as many factors affecting the inspection process as possible. The effects of many of the operator or individual factors, social factors, and organizational factors are minimized with AI-based inspection. Even some of the task factors and environmental factors are controlled. For instance, in an automated inspection system, environmental factors such as time of the day and shift duration would have no impact on the AI model's performance. The application also allows the quality inspector to document their findings from the inspection process and store it in an inspection log.

Chapter 5

TLP-based SPHM Framework to Support Manufacturing System Maintenance

The research presented in this work is aimed at highlighting how TLP can be transformative in PHM and improve the maintenance operation. First, we review the maintenance approaches and the role played by PHM. We then explain what non-technical and technical text data are, and how text from MWOs is unique. We demonstrate how traditional NLP fails on technical text, the why there is a need for TLP. We propose a TLP framework to implement in the PHM environment and suggest some potential areas of its use. To demonstrate the capabilities of the proposed framework, we identify two scenarios using MWOs of aircraft and apply customized text processing methods to it. We first use TLP to predict corrective actions for new maintenance problems using semantic and syntactic text similarity and evaluate it using the cosine similarity score; then we use Topic Modeling to extract the most dominant topics from the MWOs and use relevant keywords to link problems to standardized maintenance codes.

5.1 Introduction

Developments in Artificial Intelligence (AI) with Machine Learning (ML), Deep Learning (DL), and Natural Language Processing (NLP) are revolutionizing several industries and the way some tasks are performed. Self-driving cars, AI-based chatbots and real-time identification of objects on smart devices are just some of the real-world applications of these technologies. In the context of manufacturing and industrial operations, the area of Prognostics and Health Management (PHM) has incorporated AI-based methods to detect, predict and in some cases, diagnose failures. Maintenance strategies such as unplanned or reactive maintenance and planned or preventive maintenance are being supplemented by approaches such as predictive maintenance that use AI [214]. With the introduction of these state-of-the-art methods, maintenance approaches can reduce excess inventory, minimize system downtime, and improve overall performance. However, even with these developments, the modern maintenance operation faces several challenges such as 1) the lack of appropriate ontologies to integrate maintenance processes, 2) complexities involved in the implementation of specific maintenance strategies, 3) shortage of well-trained manpower for the maintenance of crucial systems, and 4) limitations in the analysis of multimodal and unstructured data from heterogeneous systems.

Industry 4.0 and Smart Manufacturing based architectures are promoting digitization of the industry and making manufacturing operations more interoperable [92]. These hierarchy-free models are enabling the planning and implementation of maintenance processes in a sustainable manner. Several reference ontology models have been proposed [215]–[217] to address the challenge of integration of different maintenance processes. The second and third challenges that face modern maintenance are closely related. The complexities and difficulties associated with implementation of some maintenance approaches is exacerbated by the scarcity of trained technicians and engineers [218]. These concerns can be mitigated to some extent by technologies such as Augmented Reality (AR) and Remote Maintenance (RM) [219], [220], and can enable faster training of the workforce and provide real-time assistance on the shopfloor. To analyze data from the shopfloor, data-driven PHM methods have successfully used sensor measurements to detect faults and predict the Remaining Useful Life (RUL) of systems. Sensor signals are often numerical or image-based, and AI models have been fine-tuned for such modes of data. While this addresses some of the requirements of the fourth challenge of modern maintenance, it does not tackle all modes of data.

One part of maintenance that contains troves of data but is yet to be exploited, is the technical text data stored in Maintenance Work Orders (MWOs). Text data from MWOs are sometimes considered as ‘black holes’, i.e., they are fed with so much data, but are seldom used to make data-driven decisions. Instead, they are often regarded as historical logs that are relied upon only when there is an absolute necessity. NLP can offer some hope with its ability to analyze text data and provide appropriate solutions where necessary. However, NLP's successes predominantly come from the analysis of text that are a part of non-technical sources. NLP can be used on different types of data using domain-adaptation or transfer-learning, but this assumes that to process data from-low resource domains, there are high-resource domains that are somewhat similar consisting of annotated data [221]. That is not the case when it comes to technical data, specifically data from MWOs. These documents are often unstructured or semi-structured, they adopt unique language dictionaries, and use colloquialisms and jargon that are domain specific. In order to process text data, the approach of Technical Language Processing (TLP) has been proposed [222]–[224]. TLP uses a human-in-the-loop strategy to address the challenges posed by technical text data with the help of customized NLP methods.

5.2 The Role of Prognostics and Health Management

The introduction of data-driven methods with ML and DL are bringing proactive maintenance approaches to the forefront [214]. Condition-monitoring of systems using Internet of Things (IoT) based sensors and field devices allow the measurement of system parameters in real-time that can be used to record large amount of data for analysis. This can be used along with historical data and domain knowledge to predict when failures would occur or how close they are to occurring. These methods have been used successfully in fault diagnosis of rotating machinery [225], defect identification of stainless steel welds [226], in monitoring and predicting the quality of solutions in electroplating [227], and in several other areas. While these methods provide excellent results in the real-world, they have limitations in terms of the types and formats of the data that can be used. Sensor signals and other measurements primarily take the form of numeric or image data. These methods are not equipped to handle text data, especially the technical text data from domain-specific maintenance operations.

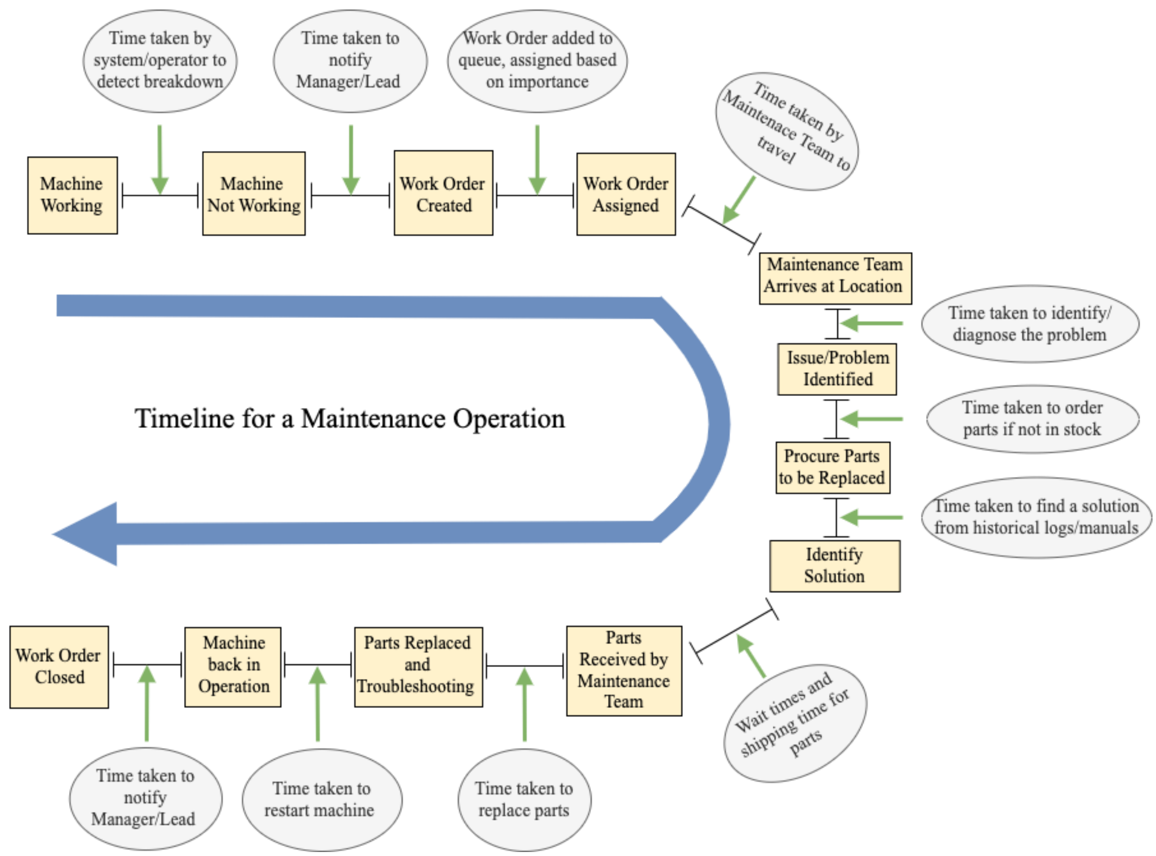

Figure 24. Timeline for a maintenance operation.

Prognostics and diagnostics of systems and components can be enhanced if the analysis of technical text data can be incorporated into existing approaches. Consider the timeline for a typical maintenance operation shown in Figure 24. There are many steps involved right from detecting a breakdown, creating an MWO, to fixing the problem, and closing the MWO. We can also observe how much time is spent in the entire process. In the current maintenance timeline, there are several opportunities for technical text processing that can help save time. In a reactive maintenance setting, once a breakdown is detected, corrective actions could be predicted based on the problem text. Similarly, in predictive maintenance, although time is saved by predicting when a failure would occur, time is still spent in diagnosing the type of failure and in repairing/replacing the necessary parts. In this case too, analyzing technical text can help speed up the process. There are a few more potential applications and use-cases for TLP in the context of the typical maintenance operations timeline that we will discuss.

5.3 Text Data

5.3.1 Non-Technical Text

We interact with non-technical text data presented to us on a daily basis via different forms of media. On our mobile devices and computers, text data usually takes the form of digital newspaper articles, webpages, blogs, Short Message Service (SMS) text, e-books, etc. Almost all of these text data are non-technical and based on spoken languages or are derivatives of spoken languages. Since these languages are a part of everyday interaction, there is a lot of scope to use the data from it, which is exactly what NLP methods are built for. One of the main requirements for NLP models to succeed is the availability of large amounts of text data, also referred to as corpora, for model training purposes. Since non-technical data is available in abundance, it is not a problem for the NLP pipeline. Such non-technical text data have been assembled into several corpora and have been made publicly available. Corpora such as Penn Treebank Corpus [228], British National Corpus [229] and WikiText-2 [230] are used for the generic training of NLP models, and each of them contain tens of thousands, if not millions of words. Some corpora are designed for specific tasks. Amazon Customer Reviews [231] and IMDb Movie Reviews [232] are commonly used to train models for Sentiment Analysis. The Reuters Corpus (RCV1) [233] is routinely used for text clustering. Similarly, there are corpora designed for Named-Entity Recognition (NER) and Parts-Of-Speech (POS) tagging tasks. Genre-specific corpora have also been compiled for formal text from academic articles, journals, and conferences; informal text from emails, blogs, and webpages; spoken language from language databases; etc. This brings to light how resource-rich some of the non-technical areas are, and how NLP models can be trained to achieve high levels of accuracy in their tasks.

5.3.2 Technical Text

The term ‘Technical text’ is not a clearly defined one and has been used to described text data from various fields [234]. In the engineering and scientific community, the term ‘technical’ can take a few different meanings. It could signify the complex mathematical formulae and equations, or the sophisticated terminology used in a specific domain. It could also be considered as a subset of a language used by experts that are familiar with the knowledge about a specific task or process. In the field of law, the question of whether law and the associated legal text can be considered to be technical has been studied in detail [235]. Arguments have been made that law is to be interpreted as a technical language when it is intended to be understood and used by legal professionals [236]–[238]. Medical terminologies can be considered to be technical text too [239], [240]. Similarly, there is technical text in chemistry [241], in physics [242], and in molecular biology [243], [244]. In manufacturing, technical data is observed in maintenance [245], and also in product inspection logs [246]. It is clear that technical text exists in science, engineering, medicine, law, and several other domains. Although these types of text data are comfortably understood by domain experts, it is not as easily understood by text processing models. Figure 25 shows examples of technical text data from different disciplines.

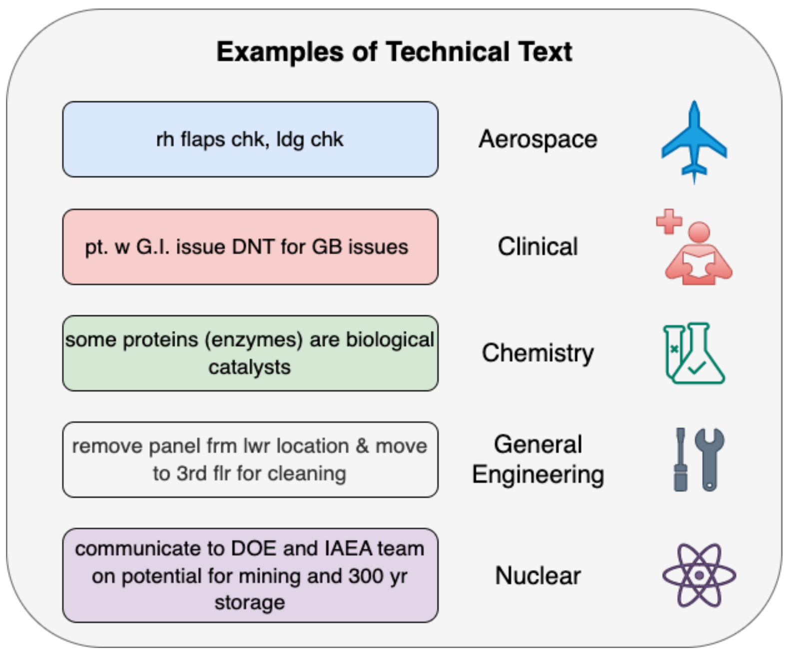

Figure 25. Examples of Technical Text data from various disciplines.

5.3.3 Text from Maintenance Work Orders

MWOs are critical in recording data from each step of a maintenance operation. These documents can provide an in-depth understanding of the system's performance over time, allowing the extraction of valuable knowledge. This type of technical data is unique due the fact that it is generated by a human source as opposed to being readings from a machine or instrument [247]. The records essentially consist of thoughts about a specific problem expressed by technicians in a language they are comfortable using. Many industrial operations, particularly Small and Medium-sized Enterprises (SMEs), use hand-written formats for MWOs. The digitization of the shopfloor has made it possible to store data in a more efficient manner. Modern MWOs use Computerized Maintenance Management Systems (CMMS) to record data in a semi-structured format [248]. The text used in MWOs differs significantly from other corpora, including a lot of technical ones, because engineers, operators, and technicians tend to use domain specific verbiage and jargon. MWOs often comprise of inconsistent language, incomplete entries, or sometimes no entries at all [249]. In some cases, different shorthand notations are used to refer to the same component, which can lead to discrepancies [250]. Text used in MWOs are also much shorter when compared to other text in the NLP corpora, even though they may contain the same number of records [223]. MWOs typically consist of a unique identifier, the problem, action taken, and timestamp of when it was created and closed [251] as seen in Table 12.

Table 12. Example of a typical MWO.

MWO Number	Asset Identifier	Problem	Action	Opening Time	Closing Time
1033982	14C – Conveyor	Roller & belt needs re-cal on both sides	Calibrated and ck'ed OK	22:00:00 hrs	23:50:00 hrs
1033982	191AS – Airstair	Rearmost Halobulb fixt. cracked under Airstair No 91	Fixt removed and replaced	22:48:00 hrs	23:25:00 hrs

5.4 Inability of Natural Language Processing to handle Technical Text

NLP has been successful at adapting to a few technical disciplines. Technical text from medicine has been processed with good results [39], [252]. In supply chain, semantic text matching has been used in the management of transportation assets [40]. NLP has also adapted to technical text from finance in the form of chatbots [41]. With these successful adaptations, one may wonder why it cannot be used for text data from MWOs. We use insights from previous works [222], [223] to understand the drawbacks of NLP for technical text from MWOs.

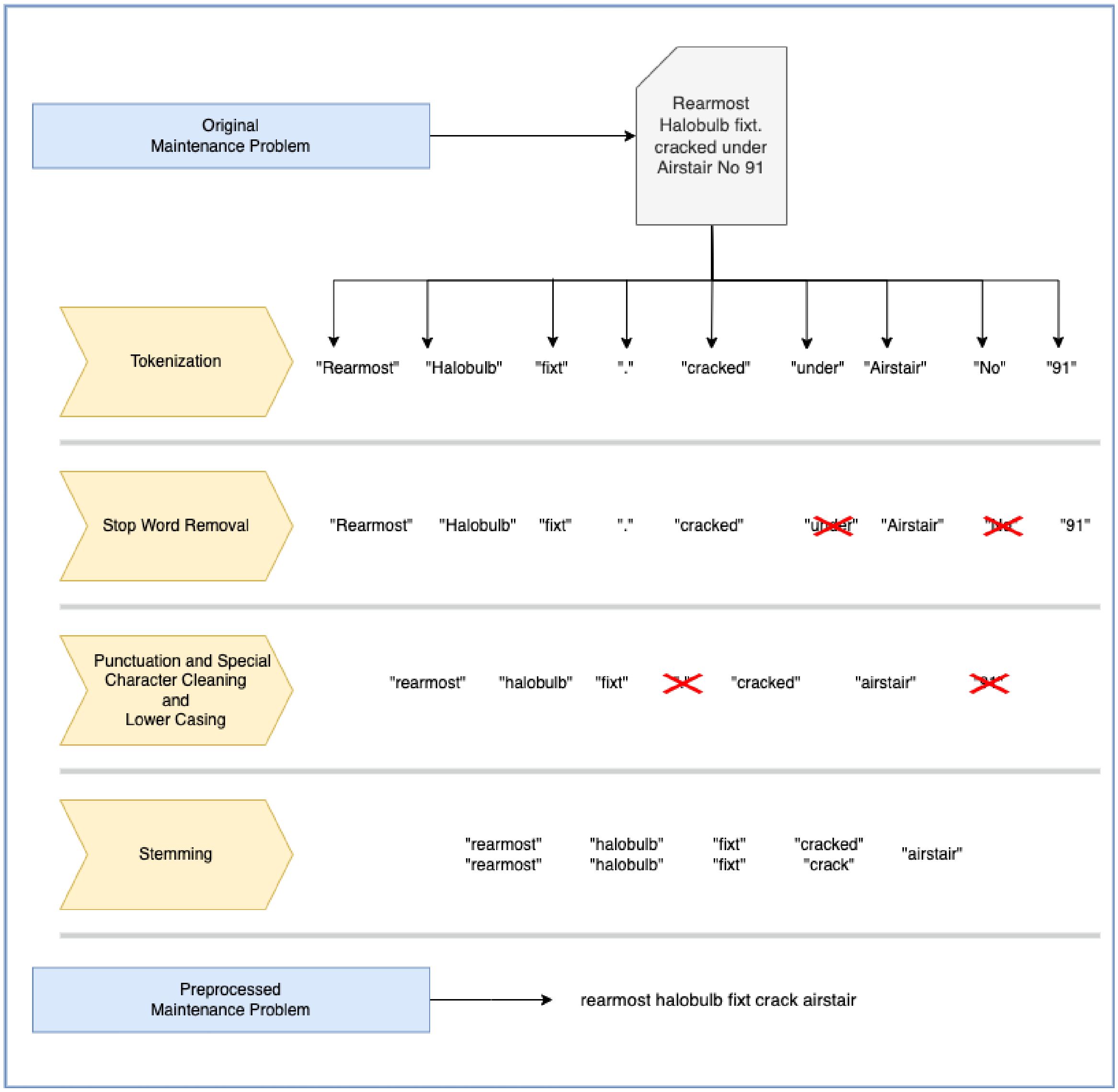

Figure 26. Traditional NLP's flaws when used on technical text from MWOs.

Consider the second maintenance problem from Table 12 that reads *Rearmost Halobulb fixt. cracked under Airstair No 91*. The maintenance jargon and short hand is not very easy to understand. Translated to spoken English, it means "the rearmost halogen bulb fixture located under airstair number 91 is cracked". We use a typical NLP pipeline for preprocessing the technical text data (See Figure 26). In the first step of Tokenization, algorithms segment the text data into words and phrases, and these tokens become the input for the NLP task [253]. The tokenizer converts the string of technical text into individual tokens, but the word *Halobulb* has not been tokenized correctly. We then use these individual tokens as input for the next step, stop word removal. Stop words are considered to be unimportant words that provide little information [254]. Removal of words such as *under* may be useful in a non-technical scenario but are very important in the context of MWOs. In this case, the word *under* is stating the location of the halogen bulb. Stop word removal also removes the word *No*, by assuming that it is used to convey negation, when in reality it is short hand for "number" in the context of this MWO. The next step of text cleaning performs some basic operations such as punctuation and special character removal, and lower casing of the text string. This step removes the punctuation mark used for period (full stop), and also removes the number *91* that is used to identify the airstair. Stemming, is the following step, and is used to find the stem words or the root words. The word *cracked* is reduced to its stem, "crack". Technical jargon like the word *halobulb* is not stemmed to the separate root words "halogen" and "bulb" due to its complexity. The term *fixt*, referring to the "fixture", is also not stemmed to its root word "fixtur". This concludes the preprocessing stage and results in the preprocessed text reading *rearmost halobulb fixt crack airstair*. Although this phrase consists of some of the keywords from the original technical text, it has lost its meaning. The location of the airstair has been lost during the preprocessing, and number identifying the airstair has also been discarded by the NLP pipeline. If this processed text string were to be used as input to a prediction model, the results could misrepresent the severity or seriousness of the actual maintenance problem, or in some case, even lead to hazardous consequences.

The complexities involved in processing technical text from MWOs are quite clear, and traditional NLP is not quite suited for this task. Domain adaptation from other technical domains is also not possible because the maintenance data is so unique. Transfer learning cannot be used either due to the absence of similarly structured technical data on which models can be trained. It is evident that domain-knowledge needs to be incorporated into the text processing pipeline considering the peculiarities of technical text used in MWOs.

To demonstrate how this can be made possible, we propose a framework to implement TLP to process technical text from MWOs that uses expert knowledge.

5.5 A Technical Language Processing Framework for Prognostics and Health Management

One of the major reasons why NLP is unsuccessful on MWOs is that the complexity of technical text makes it extremely difficult for a computerized model to understand the context in which certain words or terminology are used. To overcome this obstacle, expert knowledge must be incorporated during various stages of processing for decision-making. We propose TLP framework for PHM that incorporates domain knowledge at each step of processing as shown in Figure 27.

The first step in the TLP framework involves obtaining the MWOs that have been generated. These work orders can consist of a varying number of fields such as an identifier, the opening and closing times of the work order, the maintenance problem, the corrective action taken, the name and an identifier for the technician, the part replaced, the costs involved, etc. Since there is no universal format for MWOs, it is essential to understand the technical text that is presented in it. To decipher the short hand, technical jargon, abbreviations, and other colloquialisms used, expert knowledge needs to be utilized. Historical data and use-cases are used as references to analyze raw data that is presented [222]. Dictionaries with technical terminology, abbreviations, morphosyntactic information, stop words and other relevant data are collected by talking with technicians, operators, and maintenance engineers. This data, in addition to the data from sources such as maintenance manuals, CMMS, and other metadata are collectively referred to as "fortuitous" data [223], [255].

Once there is sufficient understanding of the text, appropriate preprocessing steps can be selected. Methods such as Tokenization, stop word removal and lemmatization can be used in a such a way that it preserves the meaning of the original text. Unnecessary stop words are screened, technical jargon is substituted with understandable verbiage, and any punctuations that do not alter the meaning of the data are removed. The preprocessed text is then examined to ensure that the quality of the data has not been lost. If it is deemed that the preprocessing is not satisfactory, it is revised.

The preprocessed technical text can now be used for a variety of tasks. There are several areas of PHM and maintenance that can benefit from TLP. With the help of domain experts and analysts, the areas where TLP will be effective can be identified, and appropriate computational resources can be procured. TLP can be used to predict corrective actions for new maintenance problems by comparing them with existing ones from historical MWOs. In the same vein, it can also be used to match the technical text to identify what parts might need to be replaced and refer the technicians to the necessary manuals/documentation. If there is some form of annotating to be performed on MWOs, a TLP model can assist performing that task too. Another potential area of where TLP can optimize the maintenance process is by incorporating it directly into a predictive maintenance framework. Such prescriptive maintenance systems that are able to handle multimodal data have been proposed [256], [257] but it remains to be seen how they can be incorporated across the industry. Nevertheless, it shows that there are many areas where TLP can be used with PHM to improve maintenance.

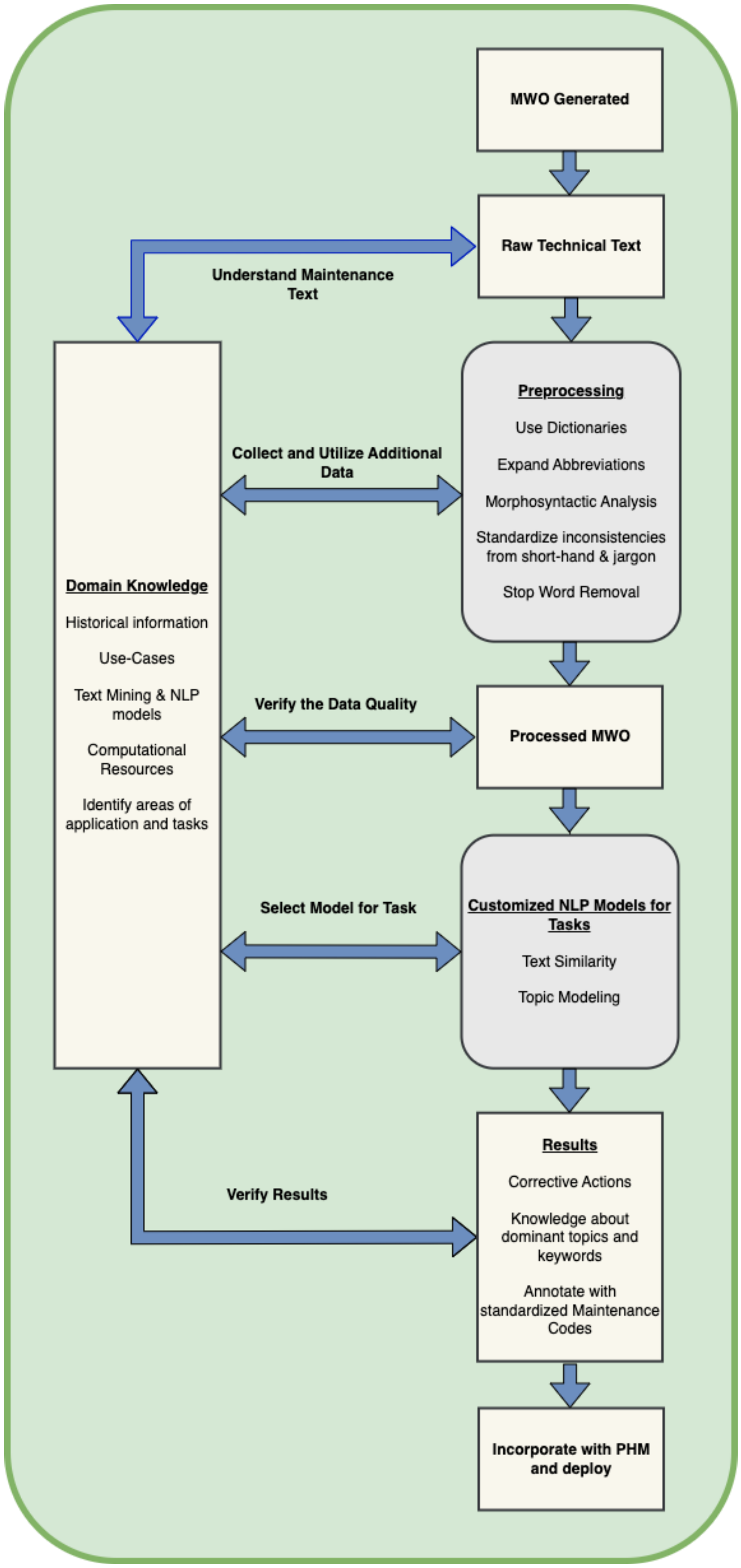

Figure 27. A framework for Technical Language Processing.

Once it is established where TLP would be effective in the maintenance timeline, text processing methods compatible with the tasks can be identified. Our understanding is that text similarity methods with semantic and syntactic similarity can be very useful for a variety of tasks – matching new maintenance problems to existing problems actions to predict corrective actions, using keywords from processes to identify replacement strategies for tools/components, and several other tasks. Another suitable method for data from MWOs is Topic Modeling. It identifies the most dominant topics and keywords from maintenance problems, helping to quantify what systems/components utilizes the most resources. Topic Modeling also assists in linking MWOs from dominant topics to standardized industry codes, making annotation of the data easier for future tasks. The results from all of the methods applied to the preprocessed MWOs are evaluated by experts in the field to verify its validity.

When the performance of the text processing methods is deemed to be satisfactory, i.e., a sufficient number of corrective actions have been successfully predicted for new maintenance problems, or new problems have been effectively linked to appropriate topics, the next steps are considered. These steps involve deploying it as a part of CMMS, or as a standalone tool for PHM. As noted previously, it can also be incorporated as a part of the predictive maintenance paradigm. There are many possible use-cases for such a framework but to demonstrate our work, we identify two scenarios where it can be used on aircraft MWOs.

5.6 Applying Technical Language Processing to Aircraft Maintenance Work Orders

There are a limited number of publicly available datasets with technical text. We use the Aviation Maintenance dataset that is originally from University of North Dakota's Aviation Program, made available by researchers [258] on the open-source web-interface called Maintnet. The data consists of three attributes out of which only two are of interest to us – the Problem which contains text about the maintenance problem/issue, and Action or which contains text that describes the action taken to address the problem. A set of supplemental data files that contain abbreviations dictionaries, morphosyntactic information, and domain term banks are also provided. Table 13 shows four randomly selected records from the MWO.

Table 13. Randomly selected records from Aviation maintenance data available at Maintnet.

IDENT	PROBLEM	ACTION
100111	L/H ENGINE #4 AFT BAFFLE CRACKED IN MULTIPLE PLACES.	INSTALLED REPAIRED BAFFLING.
100532	SCROLL TO PC FLITTER TUBE HAS EXCESSIVE FRETTING UNDER FLOAT	ORDERED LINE.
103139	ACCIDENTALLY OVERSPED ENGINE BEFORE PULLING THROTTLE BACK, E	EVALUATED DATA ON FDM CARD & FOUND ENGINE OVERSPEED TO
104077	CYL #2 & 4 ROCKER COVER GASKETS ARE LEAKING.	REMOVED & REPLACED ROCKER COVER GASKETS.

The creators of Maintnet have specified that the original data consisted of text that represented the maintenance problem, the action, the ATA chapter codes (maintenance codes as prescribed by the Air Transport Association), open and close dates for the MWOs, and a work order identifier [259]. However, due to the presence of sensitive information and privacy concerns, the data was de-identified and confidential information was removed from the publicly available version. The absence of annotations makes it challenging to apply any form of text processing methods especially since there are only two attributes. However, this opens up a window of opportunity for unsupervised methods to be used.

Given how the technical text from the aviation MWOs is unstructured, we devise two scenarios to demonstrate the capabilities of TLP. First, we use the problem and action attributes as historical data and apply syntactic and semantic text similarity models to predict corrective actions for new maintenance problems. In the second scenario, we apply

Latent Dirichlet Allocation (LDA) for Topic Modeling to identify the most important topics and salient terms and link them to the standardized ATA maintenance codes.

5.6.1 Preprocessing Maintenance Work Orders

The preprocessing steps we consider are broadly the same for both the scenarios, barring a few minor differences that will be highlighted. As we have seen in Table 13 the technical text from the MWOs of aircraft consist of abbreviations, short-hand, and some aviation maintenance specific jargon. To ensure that we capture all the important information from the data, we need to follow some preprocessing steps. Traditional NLP preprocessing steps such as Tokenization and English language stop word removal can significantly alter the meaning of the text from the MWOs. Consider terms such as *reswaged* and *resafetied* from the aviation MWOs that may not be tokenized correctly. Models trained on non-technical data will not be able to tokenize the technical slang. Words such as *On* and *Off* provide very important context in the MWO. If the English stop word list is used, they will be removed that can change what a phrase means potentially influencing important decisions.

The first step in preprocessing is converting abbreviations, short-hand, and colloquialisms into their full versions or expanded forms respectively. This will allow easier understanding of these phrases and will be more presentable to an audience that is not exclusively comprised of domain experts. We use the list of abbreviations provided with the aviation dataset. But upon further examination of the data, we realize there are many more abbreviations that needed to be expanded. Terms such as *r/h* and *l/h* meaning "right hand" and "left hand" respectively have been used extensively in the MWOs. Other short hand included *i/b* for "inboard", *o/b* for "outboard", *a/c* for "aircraft", *cht* for "cylinder head temperature", *egt* for "exhaust gas temperature", *c/w* for "complied with", *fdm* for "flight data monitoring", and many more. We tried to be as exhaustive as possible in the task of replacing abbreviations with the full words and phrases. In the next step, we focus on eliminating stop words. We know that stop words can either be a nuisance or of great value in technical text. So, we need to ensure that the stop words we choose to remove are appropriate in the context of the MWOs. We identified a list of stop words suitable for TLP [260]. However, words in this list were selected from various technical texts that were not domain specific. For the MWOs in consideration, this is not a complete list that is representative of the text. We customize the list of stop words so that they do not drastically change the meaning of the text. There are some words in the list such as *along*, *good*, *many*, *straight*, *forward*, and *upon* which are important in providing context to the maintenance problem and actions, so they are retained.

5.6.2 Scenario 1: Predicting Maintenance Actions Using Text Similarity

The value of saving time by improving maintenance processes is often ignored or overlooked [16]. In situations where delays in repairs could be very expensive, improving maintenance times becomes a priority. In the aviation industry, devising corrective actions for new maintenance problems is a time sensitive process in a highly stressful environment [261]. Any errors made in such situations could be catastrophic. To assist in the decision making process in such a highly challenging environment, TLP could be put to use.

In this scenario, we use the text data from the MWOs to predict what the corrective action should be for a new maintenance problem. We use keywords present in the new maintenance problems and compare them with ones from historical problems in the MWOs. Two approaches are considered: syntactic text similarity and semantic text similarity. There are research works that apply these methods to the electric power industry operations [262], biomedical texts [263], medical records [264], railway safety [265], production line failures [266], and also to aviation maintenance [267]. Our aim is to place a greater emphasis on how these methods can be applied to technical data and how effective they are in the context of improving the maintenance operation.

5.6.2.1 Syntactic Textual Similarity

When a new maintenance problem arises, it is first preprocessed to convert all the technical abbreviations to their expanded forms. Each word in the new maintenance problem is recorded in order to allow syntactic and semantic matching. To determine syntactically similar matches of the new maintenance problem, it is compared with all of the historical problems that have already been recorded in the MWOs. One might consider only exclusively selecting the historical problems that contain any of the keywords present in the new maintenance problem, but there is a potential issue with that approach. A stop word or a word that is extensively used across all maintenance problems might be present in the text. Figure 28 shows the raw text from new maintenance problem, the text after preprocessing, the extracted keywords, and how problems in the historical MWOs are matched with the keywords. In this instance, we see how the word *had* appears in many maintenance problems. If we use this approach to determine how close the existing maintenance problem is to the new problem, we could end up getting many records that are irrelevant due to the presence of an unrelated keyword, a stop word that might have slipped through, or even a technical word that is not pertinent to the meaning of the text. To address this, we use the Term Frequency-Inverse Document Frequency (TF-IDF) method to find the most common terms. The TF-IDF implementation in Python involves transforming all the text from the maintenance problem attribute into a feature matrix. First, the frequencies of all words in the MWOs are recorded. TF-IDF then incorporates a concept called inverse document frequency [268]. This measures the amount of information provided by a word by looking at how frequently it is present in the corpus.

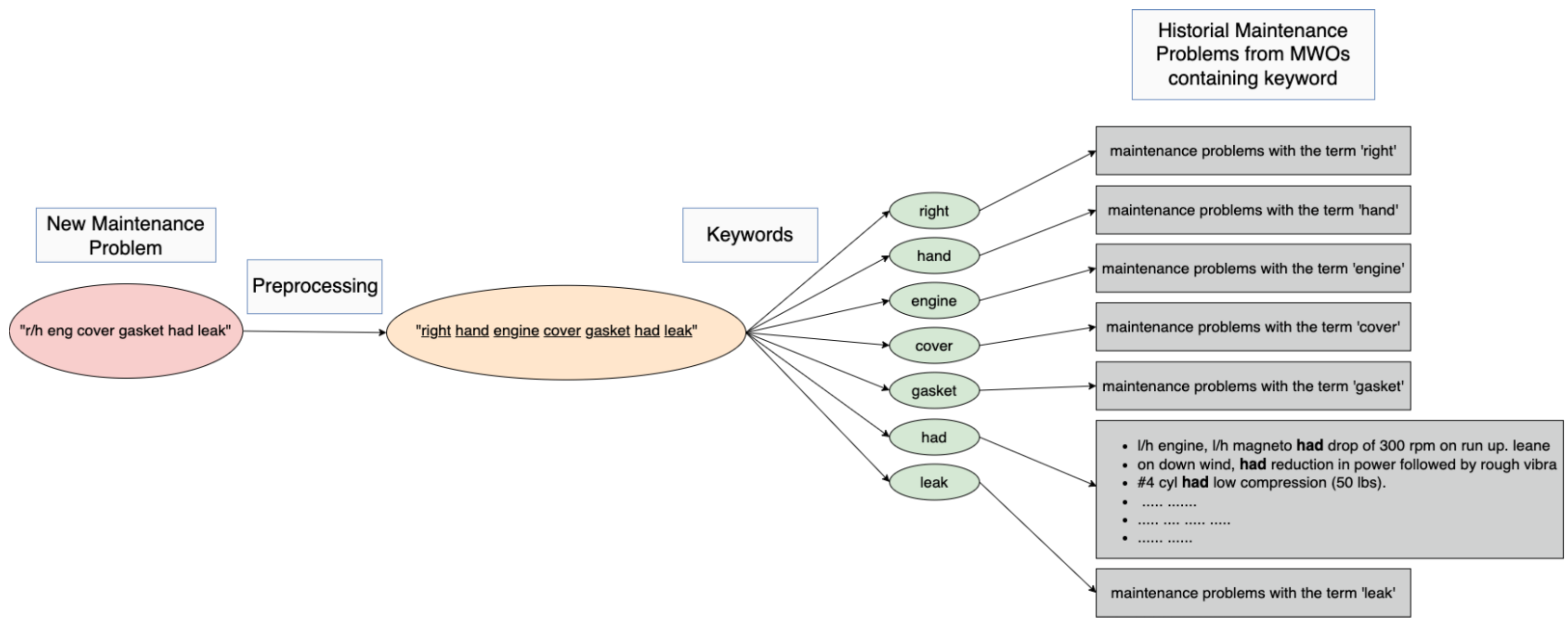

Figure 28. Keyword based matching.

We use the Scikit-learn library's [269] TF-IDF implementation in Python, that can be mathematically defined as shown in Equation (10):

$$idf(t, D) = log\left(\frac{(N+1)}{df(t)+1}\right) + 1 \tag{10}$$

where t is each term or word, N is the number of documents, and $df(t)$ is the number of documents that contain the words t. This method uses smoothing, i.e., the constant of "1" is added to both the numerator and denominator to prevent divisions by zero. The TF-IDF transformation is also applied to the new problem text, in addition to being applied to all the problems from the MWOs. To identify how similar the vectorized versions of the new problem are to the existing problems, we use the cosine similarity metric, which is the angle between the two vectors calculating by using an inner product. Given two vectors A and B, the cosine similarity between them can be calculated as shown in Equation (11),

$$cosine\ similarity\ (A, B) = \frac{A.B}{\|A\|\ \|B\|} \tag{11}$$

where, $A.B$ is the dot product between vectors A and B; and $\| A \|$ and $\| B \|$ are the L2 norm of vectors A and B respectively.

5.6.2.2 Semantic Textual Similarity

Implementing semantic text similarity is a little more complex as compared to syntactic text similarity. Methods such as Bag of Words (BoW) and TF-IDF cannot effectively capture the similarity between different words that convey the same meaning or idea [270]. It is even more difficult to detect different words that could convey the same concept when using technical vocabulary. According to research, semantic similarity can be classified into corpus-based similarity and knowledge-based similarity [271]. Due to the lack of availability of a publicly available corpus for technical text data, and specifically maintenance related data, we consider a pre-trained sentence transformer model. Transformer models are basically encoder-decoder models. The task of the encoder is to take an input of raw text and map it to a numerical sequence. The decoder then uses the output of the encoder along with other contextual information to generate a meaningful output [272]. One of the state-of-the-art models is the Bidirectional Encoder

Representations from Transforms or BERT [273]. For tasks such as semantic textual similarity, BERT has been modified with Siamese networks and is called Sentence-BERT or SBERT [274]. The sentence transformers library from Huggingface [275] provides many different methods that we can use for our task of semantic textual similarity. Considering this to be an unsupervised learning task, we choose to use the pre-trained model 'all-mpnet-base-v2'[2] model which is trained on more than 1 billion records for our task. When we feed the model with a new maintenance problem, the sentence transformer will consider all the historical maintenance problems and match them based on semantic similarity. We then use the cosine similarity metric to identify how semantically similar the new problem is to existing problems.

5.6.2.3 Results of Textual Similarity Methods

To demonstrate how syntactic and semantic textual similarity can be used, we generate three new instances of maintenance problems. The first new problem is almost identical to a consistently appearing existing maintenance problem in the MWOs. The second maintenance problem is also identical, but only semantically. The third maintenance problem shows how close semantic and syntactic matches can be obtained if similar words appear in both pieces of text. Table 14 shows the new maintenance problems; their preprocessed versions after stop words removal and abbreviation expansion; the matching syntactically similar problem and semantically similar problem with their matching corrective action predictions and cosine similarity scores respectively. We can see that in the first instance, the syntactic match and semantic match both show existing problems that are very similar to the new problem, and this logical similarity is further justified by their cosine similarity scores. In the second instance, the new maintenance problem consists of the word *disintegrating*, which does not appear anywhere in the existing problems from the MWOs. However, using the sentence transformer model, a semantically similar textual match is found in the word *worn*. In the same instance, the syntactically similar match does not appear to be a good fit. For the third new maintenance problem, the words *not clean* are semantically matched with the word *dirty* to find a match. For a syntactic match of the third problem, an existing maintenance problem where engine cleaning is needed is matched.

Table 14. New maintenance problems matched with syntactically similar problems and semantically similar problems along with their corresponding predictive actions and cosine similarity scores respectively

New Maintenance Problem	Preprocessed New Maintenance Problem	Syntactically similar problem using TF-IDF	Corrective Action using Syntactic Match	Cosine Similarity	Semantically similar problem using a Sentence Transformer	Corrective Action using Semantic Match	Cosine Similarity
l/h and r/h rocker cover gasket leak	left hand and right hand rocker cover gasket leak	right hand engine 2 and 3 rocker box cover gasket leaking	replaced gaskets	0.93	2 rocker cover gasket leaking, right hand eng	removed and replaced rocker cover gasket	0.93
baffle plugs disintegrating	baffle plugs disintegrating	both baffle plugs missing too many tabs	installed baffle plugs	0.66	baffle plugs worn	replaced both baffle plugs	0.81
eng is not clean	engine is not clean	both lower engine cowls need cleaning	cleaned both cowls	0.81	engine is dirty	washed engine	0.82

Using the syntactically and semantically matched maintenance problems, we can easily identify the corresponding corrective actions in the MWOs. The corrective actions seem appropriate in almost all the cases. This, however, is to be verified using expert-knowledge before implementation the action. The importance of using both syntactic textual similarity and semantic textual similarity is quite clear upon observing the results.

5.6.3 Scenario 2: Topic Modeling

The original MWOs consisted of a few more attributes such as the Air Transport Association of America chapter codes, commonly referred to as ATA chapter/code. ATA codes are a standardized numbering system used by pilots, engineers, and maintenance technicians across the industry. Over the years, the ATA codes have been modified to what is now the Joint Aircraft System/Component (JASC) codes, colloquially referred to as JASC/ATA codes [276]. Understandably, these codes are a very critical part of the record-keeping process of different airlines, private aircraft operators, and manufacturers. However, it is not an easy task to link the maintenance problems to these codes [277], and this is an industry-wide issue due to the complexities of maintenance problems. In a lot of cases, MWOs are manually linked to the ATA codes which can be a time-consuming process. Since many of the maintenance issues reported in the MWOs are similar, we can apply TLP to process the data and predict the matching ATA codes. A technique known as Topic Modeling helps in identifying and extracting keywords to determine the most dominant topics prevalent in the data. Topic Modeling has been applied in several domains such as aviation safety [278], software maintenance tasks [279], and railway fault diagnosis [280]. We use Topic Modeling to extract the most dominant keywords and identify suitable topics that can be linked with the standardized ATA codes.

5.6.3.1 Latent Dirichlet Allocation

The algorithm we choose for Topic Modeling is Latent Dirichlet Allocation (LDA), which is an unsupervised probabilistic model introduced for machine learning [281]. It characterizes documents as a mixture of topics where each topic is represented as a distribution over words [282]. A few assumptions made for LDA are: 1) each document is represented as a probabilistic distribution over topics, 2) the topic distributions in all the documents have a common Dirichlet prior, 3) each topic is also represented as a probabilistic distribution over words, and 4) word distributions in all the topics have a common Dirichlet prior as well. To understand the math behind LDA, let's define a few terms and equations.

Consider a corpus D with documents M having a vocabulary of size N. The documents d consists of N_d words with $d \in \{1, \dots, M\}$. The generative process for each document W in corpus D assumes:

- Choose a multinomial distribution ϕ_t for a topic t with $t \in \{1, \dots, T\}$ from a Dirichlet distribution with a parameter β.

- Choose another multinomial distribution θ_d for document d with $d \in \{1, \dots, M\}$ from a Dirichlet distribution with parameter α.

- For a word w_n with $n \in \{1, \dots, N_d\}$ in a document d,

  a) Select a topic z_n from θ_d,

  b) Select a word w_n from ϕ_{zn}.

For the above process, the observed variables are the words, the latent variables are ϕ and θ, and the hyperparameters are β and α. The probability of a corpus is shown in Equation (12):

$$p(D|\alpha,\beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d)\, p(w_{dn}|z_{dn},\beta)\right) d\theta_d \quad (12)$$

To make this task as close to what it would be like in the industry, we only consider the maintenance problem and do not use the corrective action data. The same pre-processing steps outlined in Section 5.6.1 are applied to the MWOs with some minor changes. First, some more stop words are removed since words like *right hand*, *need*, *have*, *and*, etc. have little importance when we are trying to identify a topic to match with ATA codes. Next, we perform word tokenization on the data, which means that we convert words into numeric representations. Since we have preprocessed the data pretty thoroughly, word tokenization should be effective. LDA then identifies the most dominant topics in the MWO along with the associated keywords that are considered to be salient terms. To determine how all the words are associated to each topic is determined by making two important considerations. First, we consider a ranking measure called *Lift* [283], which is the ratio of a word's probability within a topic to its marginal probability across the entire corpus. Then, we consider a parameter called *Relevance* [284], a method for ranking words within topics explained in the following manner.

ϕ_{kw} denotes the probability of a word $w \in \{1, \dots, V\}$ for a topic $k \in \{1, \dots, K\}$ where V is the number of words in the vocabulary and p_w is the marginal probability of a word w in the corpus. The value of ϕ is estimated using LDA and p_w is determined from the empirical distribution of the data.

Relevance of word w to topic k given weight parameter λ, where $0 \leq \lambda \leq 1$, is shown in Equation (13):

$$r(w, k \mid \lambda) = \lambda \log(\phi_{kw}) + (1 - \lambda) \log\left(\frac{\phi_{kw}}{p_w}\right) \tag{13}$$

where λ is the weight assigned to the probability of the word w under topic k relative to its *Lift*.

5.6.3.2 Results of Topic Modeling with LDA

For LDA model, we experimented with different learning rates and a different number of topics to obtain the most effective representation of the data in the MWOs. We find that the learning rate of 0.7 using the batch learning is most suited for our task and the model is run for 20 iterations. We identify that the optimal number of topics is between two and three, so we use three topics to characterize the data. We create a visualization in Python using the LDAvis method [284] to generate a global perspective of the topics (see Figure 29). The areas of the circles are representative of the relative prevalence of each topic in the MWO corpus. The inter-topic distances used in this model are computed using the Jensen-Shannon divergence [285] and scaled with principal components as the axes using Principal Component Analysis [286]. We observe large, non-overlapping topics conveying that the topics are distinct. Here, Topics 1, 2, and 3 contain 39.9%, 33%, and 27.1% of all the tokens in the corpus respectively.

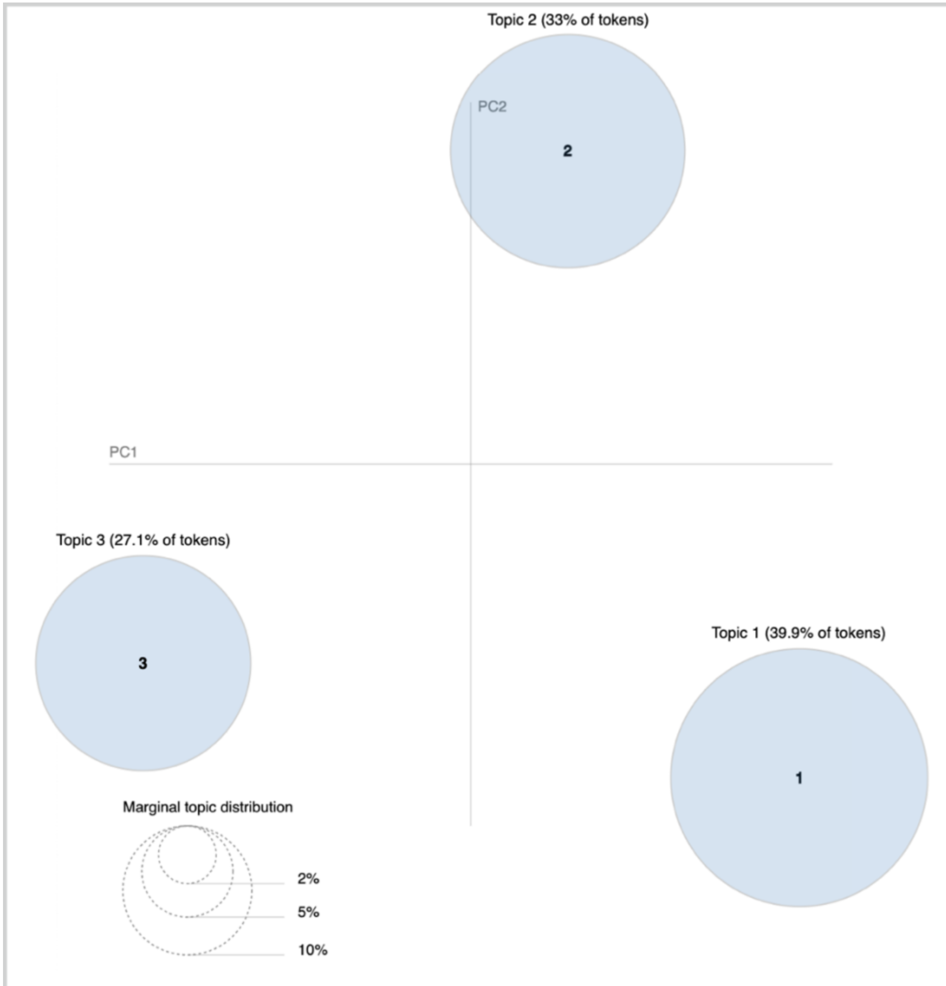

Figure 29. Inter-topic distance map via multidimensional scaling.

To understand what *relevance* and *lift* mean in the context of the MWOs, we can look at Figure 30. The visualization shows six bar graphs, with two graphs for each topic. For each topic, we evaluate the words associated with it by setting the value of the weight parameter as $\lambda = 1$ and $\lambda = 0$. The blue bar for any word represents the frequency of that word in the overall model and the red bar represents the frequency of that within a specific topic. The slider on the top of each graph controls the *relevance* metric. A word's association with a topic can be considered to be high if the frequency of its occurrence in that topic is high. This is achieved by setting $\lambda = 1.$ In this case, words are sorted by the frequency of their occurrence in the topic, represented by the length of the red bars. A word can also be considered to be highly associated with a topic if its *lift* is high, i.e., how much the frequency of a word in a topic stands out above its overall frequency in the model. In other words, the ratio between the red and blue bars. For Topic 1, almost the same words are shown in both graphs, making it clear that those words are representative of the model. For Topic 2, we observe that the most frequently occurring terms are *baffle*, *engine*, *oil*, *cylinder*, *seal*, etc. We also note that with $\lambda = 0$, for words such as *oil*, *cracked*, *seal*, *aft*, *forward*, *screw*, *side*, etc., stand out compared to their overall frequency in the dataset of MWOs. This gives us some more information about the topic. In the case of Topic 3, the most relevant terms by the frequency of their occurrence are *engine*, *cylinder*, *baffle*, *intakes*, etc. The words that stand out the most are *intakes*, *compression*, *low*, *rpm*, *plug*, etc.

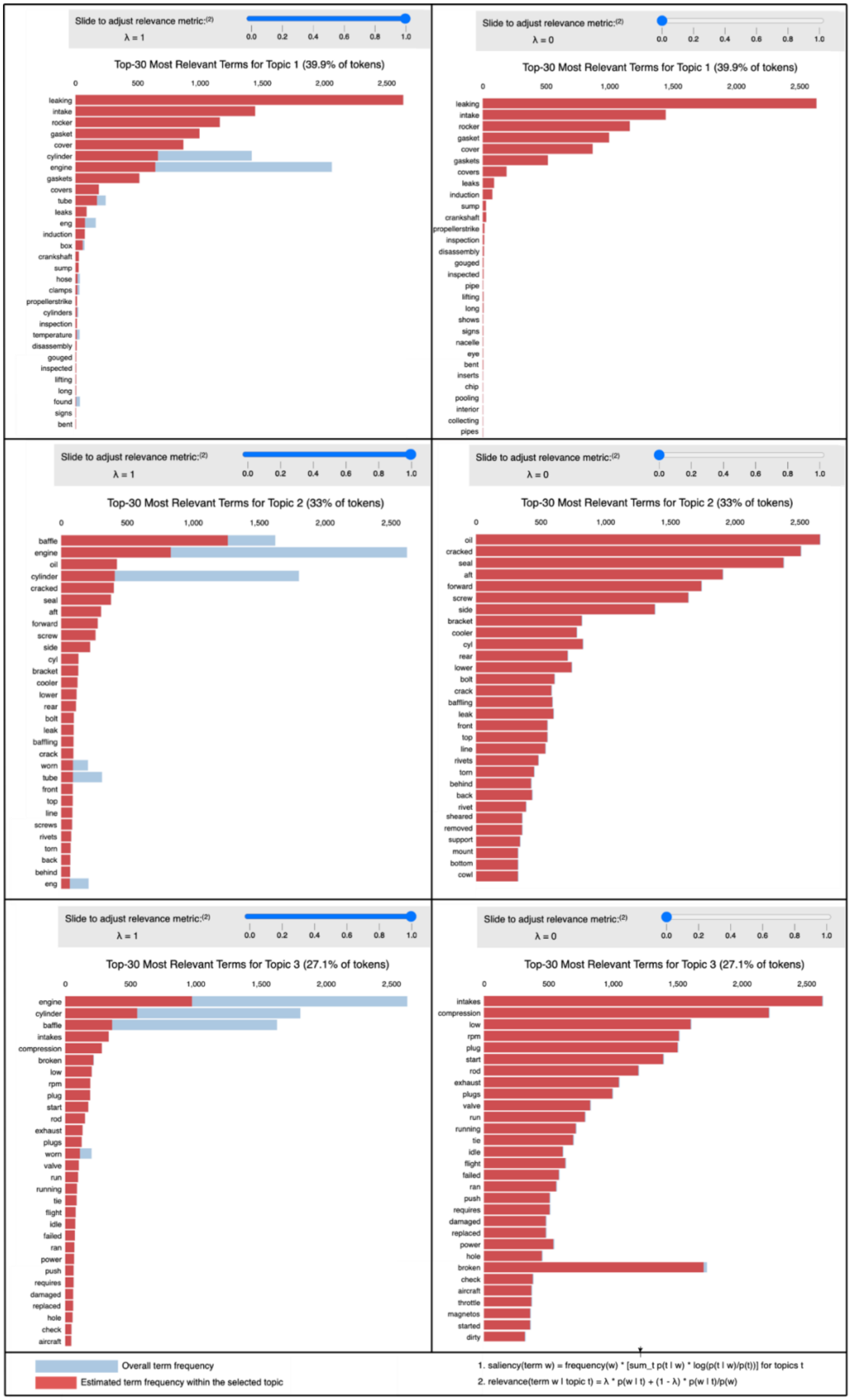

Figure 30. Most relevant terms in Topics 1, 2, and 3 for different values of the relevance metric.

We use this information learned from Topic Modeling to identify what JASC/ATA codes [276] these topics can be linked to. The JASC code of 7160 is related to all maintenance done with regards to the Engine Air Intake System. Topic 1 seems to have the keywords that fit into this maintenance category. The terms *leaking* and *intake* are dominant, and records from the MWOs that belong to Topic 1 could fall be annotated with code 7160. For Topic 2, the dominant terms are *baffle*, *engine*, and *oil*. The JASC code of 8550 pertains to Reciprocating Engine Oil System, so Topic 2 could be related to this code. Codes 7261 – Turbine Engine Oil System, and 7900 – Engine Oil System could also be appropriate annotations for the maintenance problems from this Topic 2. For the Topic 3, the terms *intakes*, *compression*, *seal*, *engine*, and *baffle* are dominant. The JASC Code of 8530 pertaining to Reciprocating Engine Cylinder Section seems to fit the description given the keywords of Topic 3. With the limited information available in the MWOs, we can see why linking standardized maintenance codes to maintenance issues can be a tedious and difficult task. Even after preprocessing with TLP, we cannot provide a 100% linking of maintenance problems to the codes because we do not have any context of the issue other than the short phrases in the MWOs. Yet, our method with Topic Modeling is capable of identifying dominant topics and getting approximate matches to the maintenance codes with the closest descriptions.

5.7 Challenges Faced

NLP has been very effective at analyzing text data from various domains. It has also been adapted to technical text from medicine, supply chain, and finance. Technical text data from MWOs though, is quite distinctive. It is generated by a human source, so the language used by operators, technicians, etc. is not easy to understand due to the highly specific colloquialisms, technical jargon, short hand, abbreviations and the domain-specific terminology. Off-the-shelf NLP methods perform poorly when provided with such data. TLP can be used to process highly complex technical text by incorporating domain knowledge.

Our research shows that TLP can be effective in tackling the complexities and heterogeneity of the technical text data. For the aircraft MWOs, we found that the data only has two columns that convey meaningful information. We also discovered that it takes an extensive study of the domain knowledge to identify what the common acronyms, abbreviations, short hand, and technical jargon used in the industry are. Stop word removal on this data was an extremely challenging task. It took several iterations to come up with the best list of stop words for our tasks. Upon examining the data, we observed that most of the maintenance problems in the MWOs are related to the power plant or very closely related to the engine and its associated components. The MWOs could have been for a smaller aircraft or a training aircraft, in which case the most important concerns would have been highlighted. For a much larger dataset, we would have probably encountered work orders from other areas of aircraft maintenance. However, the proposed framework can be extended to larger data from different domains, as long as it integrates appropriate domain-knowledge.

5.8 Discussion

In this work, we demonstrate how TLP can be applied to the PHM paradigm. We review the current maintenance strategies and the different types of text data. The notable distinctiveness of technical text data from MWOs is emphasized by examining some sample maintenance records. While NLP is successful in processing non-technical data and has been adapted successfully to certain technical text, we demonstrate how it underperforms on technical text from MWOs. The MWOs considered are highly unstructured, with a large number of technical abbreviations, short hand and maintenance related colloquialisms. To overcome these challenges, we propose a TLP framework for PHM that and provide an outline of all the steps involved. The framework uses knowledge from domain experts in a human-in-the-loop format. We also identify the potential areas of application for TLP using the framework. To demonstrate its practical applications, we apply it to technical text data from MWOs of aircraft and identify two relevant scenarios of application.

We first use the existing data from the MWOs to help in predicting corrective actions for new maintenance problems. We identify existing maintenance problems using both syntactic and semantic textual similarity techniques and predict the appropriate corrective actions. To determine syntactic and semantic similarity, we use TF-IDF and a BERT transformer respectively. Our predictions show that the recommended corrective actions are in fact appropriate for the new problems presented, and our results are reaffirmed with cosine similarity scores. Semantic textual similarity method results in cosine scores of above 0.8 for all three instances of new problems. For syntactic matches, we get cosine scores of above 0.8 for two instances of new problems, and a score of 0.66 for a new problem that contains a word that does not exist in any of the existing historical cases. This confirms that there is a need for both syntactic and semantic textual similarity methods when assessing complex maintenance data. In the second scenario, we apply Topic Modeling to the technical text from MWOs. To replicate what this scenario might be like in the real-world, we decide to use only the data from maintenance problems, and do not use the existing maintenance actions to model the topics. We use LDA to extract the most dominant topics and salient terms from the problems. We identify three dominant topics and use metrics such as *lift* and *relevance* to select the most appropriate keywords for each topic. We use these keywords and the description of the JASC/ATA codes to match the maintenance problems from each of the topics to an appropriate JASC/ATA code. This helps us to identify what code a new maintenance problems corresponds to and can be annotated with. We find that the data considered consists of maintenance problems that are predominantly related to the aircraft's engine and its nearby systems. Our results are consistent with the text descriptions provided with the standardized codes. In summary, given this research opportunity, we highlight how TLP can be used as a disruptive strategy to advance PHM and maintenance by:

1) Proposing a TLP framework for PHM that uses expert-knowledge in a human-in-the-loop format.

2) Applying the framework to MWOs from aircraft to:

    a) Predict corrective actions for new maintenance problems by using syntactic and semantic text similarity methods,

    b) Identify dominant topics and keywords from maintenance problems, matching them with standardized maintenance codes, and annotating all the records using those codes.

Chapter 6
Conclusions

6.1 Research Contributions

This dissertation applies Artificial Intelligence (AI) to health monitoring of manufacturing systems and products. The chapters presented in this work focus on four major areas: the interoperability requirements for smart manufacturing, prognostics and health management of a milling machine operation, deep learning based quality inspection of casting products, and the use of technical language processing for maintenance. The major contributions of the dissertation that address the research problems can be outlined as follows:

1) *Interoperability in Smart Manufacturing:* The need for interoperability in smart manufacturing is reviewed, and different types of interoperability are explored – syntactic interoperability, semantic interoperability, factory interoperability for horizontal integration, and cloud-based interoperability for vertical integration. Key research challenges involving data transfer, software compatibility, terminology used, and document standards to implement interoperability on the shopfloor are identified. Frameworks such as RAMI 4.0, IIRA, IBM's Industry 4.0, and NIST's service oriented architecture, and their role in different phases of the manufacturing process are discussed.

2) *Prognostics and Health Management:* A framework for Smart Prognostics and Health Management is proposed in 3 phases – setup of the equipment and data acquisition, data preparation and analysis, and modelling and evaluation. Key steps in setting up machines and sensors are specified. The consistency required in data formatting is highlighted, and the importance of understanding raw experimental parameters and sensor measurements are discussed. Methods to clean and preprocess data such as missing value treatment, and identification of noisy

observations and outliers are specified. The examination and processing of signal measurements with de-noising, amplification and smoothing are laid out. Once the signals are preprocessed, the extraction of important features using time domain, frequency domain and time-frequency domains are demonstrated along with the use of domain-specific knowledge. Feature selection approaches using correlation, multicollinearity and wrapper methods are highlighted to establish the final feature set. The use of data-driven models for prognostics and defect detection purposes is discussed. The capabilities of the SPHM framework are demonstrated by applying it to a milling machine operation. Using Machine Learning models, two tasks for classification and regression are identified. The classification task involves detection of anomalous instances, indicating that the milling machine's tool tip has degraded, and the regression task estimates the Remaining Useful Life (RUL) in terms of the number of 'runs' remaining. ML classification models demonstrate that although the accuracy of detecting anomalies is high, the recall is just over 50%. This is due to the small number of instances in the dataset. Regression models, on the other hand show high explained variance scores in estimating the RUL, but this performance too is subject to the size of the data.

3) *Quality Inspection using Deep Learning:* Visual inspection of products usually involves an operator that manually examines the product to identify defects. Task, environment, organization, social, and individual factors that affect the inspection process are highlighted. In the casting process, failures can be caused by surface defects, cracks, tears, molding flaws, scabs, blowholes, runouts, adhesions, and several other conditions. There are a lot of responsibilities placed on the inspector, thus increasing the potential for errors to be made in the process. There is a need for a visual inspection system that performs well at detecting defects, minimizes the factors affecting the inspection process, and allows documentation of all decisions made. A Convolutional Neural Network (CNN) based Smart Quality Inspection methodology is proposed with a shopfloor computer application. The application allows instantaneous defect detection of casting products, and shows an accuracy of 99.86% with high precision, recall, and F1 scores. Out of 715 images on the test set, only a single non-defective product is incorrectly identified as

defective, highlighting that it does not present any consumer's risk. The application also allows the inspector to document the inspection process in a log that is stored in a spreadsheet format.

4) *Technical Language Processing of Maintenance Work Orders:* This work explores the area of Technical Language Processing (TLP) and applies it to technical text from MWOs. It identifies the uniqueness of technical text data and highlights the complex terminologies, technical jargon, idiosyncrasies, and some of the other distinctive characteristics. The inabilities of NLP in processing technical text are corroborated by applying it to a sample technical text based instance from MWOs. We then propose a TLP framework for technical text processing that is capable of 1) understanding its intricacies, 2) processing it without misinterpreting its true meaning, and 3) utilizing it for health monitoring purposes. The framework and its capabilities are highlighted by applying it to MWOs of aircraft. Two major scenarios for its application are identified – predicting maintenance actions for new maintenance problems and matching new maintenance problems to standardized codes using topic modeling. Prediction of maintenance actions is done using semantic and syntactic text similarity methods with TF-IDF and BERT transformer respectively. Cosine similarity scores show that using semantic and syntactic text similarity to predict maintenance action is necessary due to the sophisticated nature of technical text data. The second scenario pertaining to the matching of new maintenance problems to standardized industry prescribed ATA codes involves the use of Topic Modeling to identify the most dominant topics and influential keywords. The use of the *relevance* and *lift* metrics help us to understand the importance of words that are not only frequently occurring in the model, but also how much the frequency of a word in a topic stands out above its overall frequency in the model. We then use the new maintenance problems from each topic and match them using keywords from the ATA code descriptions. This proves that AI has a significant role in solving one of the airline industry's most important problems [287].

In summary, this dissertation addresses some of the concerns of health monitoring of manufacturing systems for modern manufacturing systems. Interestingly, we note that the needs for consistent data formats and communication protocols (syntax), and standardized terminologies and vocabularies (semantics) are capable of being addressed by SPHM and TLP respectively. We hope that this work provides insight into how AI is transformative in the monitoring of systems, products, and in improving the maintenance operation.

6.2 Future Research Directions

With new systems and manufacturing architectures being proposed and introduced, standardization is gaining importance. The need for standardization and integration in making interoperability possible at every level is essential. The type of interoperability essentially depends on the systems or components of the architecture that one is trying to make interoperable. The most common type of interoperability can be observed within the manufacturing system, i.e., the physical level (Levels 2-0 of ISA-95). Other types of interoperability that the reference architecture models specifically address are between and across business units of the enterprise and factory locations. The reference architectures described in this dissertation are still at a conceptual phase and implementation has not been fully accomplished at the operational level due to the relative absence of standardization. Although syntactic interoperability has become increasingly common amongst ICT systems, semantic interoperability has yet to be developed at all levels because of the complexity in the logic applied. The future scope of research in this direction should focus on the specific interoperability challenges at each level of the network, both vertical and horizontal. Developing communication protocols built upon OPC-UA to enable an interoperable smart manufacturing environment should also be considered.

In the health monitoring of manufacturing systems, our work focused on the different steps required to successfully implement PHM. Due to limitations in the number of publicly available datasets, this work selected a dataset based on how closely it can represent a real-word manufacturing setup. This resulted in a smaller-sized dataset, necessitating the need for cross-validation techniques to limit any bias that may be present within the data. Future works in this area should attempt to use data from experimental shopfloor settings that allow the collection of large amounts of data. This in turn would allow more failure data to be collected, and enable models to provide more accurate estimates of RUL.

For AI-based product health monitoring, there are several areas for potential future research. The defects from the casting products could be classified into different types: blowholes, surface blemishes, cracks, adhesions, etc. A formal categorization of defects could be undertaken before classifying them. Localized detection of defects is another feature that can be incorporated into SQI. Localized defect detection identifies the exact location of defects and will return a probability of a defect occurring at that location. This feature was considered, but the results were not satisfactory. Figure 31 shows an example of localized fault detection on casting product images that was attempted.

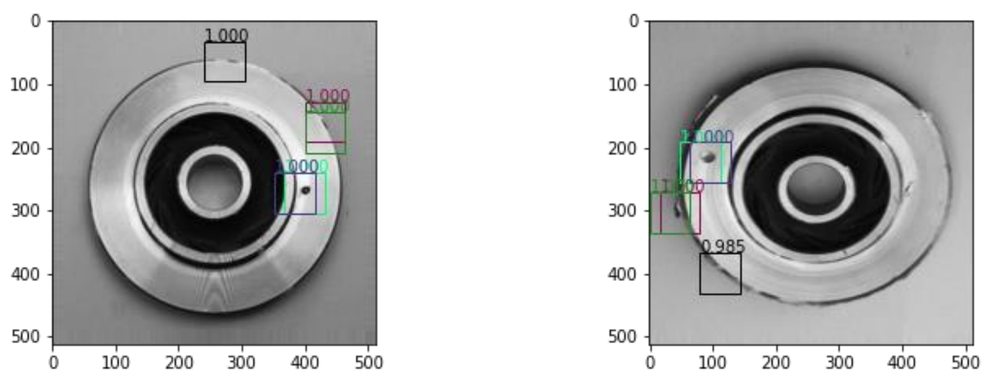

Figure 31. Localized defect detection with detection probability on two 512 × 512 sized images.

So far, the proposed SQI method has been tested on images of casting products. Some of the environmental conditions during the data collection were not in our control. For example, we had no control over the lighting setup to capture the images nor were we able to select the camera equipment used. This is the case with most publicly available datasets. On the other hand, there is limited real-time accessibility to factories where one can perform experiments. In an ideal scenario, this system can be set up directly on the assembly line by automatically taking a feed of product images. This would automate the inspection process even further and reduce any remaining factors that might affect an inspector's performance. Data from the inspection logs of the SQI tool could also be studied using Technical Language Processing (TLP) techniques.

We suggest that future works in the area of TLP for PHM be focused on incorporating technical text processing in conjunction with predictive maintenance and quality inspection techniques to form a prescriptive approach. We also propose collaboration among organizations and industries to standardize maintenance terminology and jargon, which would enable the construction of large corpora for maintenance text. TLP can also be extended to other areas of manufacturing with the abundance of text data. Quality inspection logs generate a lot of text data on which TLP can be used to help determine whether a product needs to be reworked/scrapped. The results of our research reemphasize that TLP is here to stay and our outlook for its widespread incorporation in the industry is optimistic.

Considering health monitoring as a whole, the financial implications of deploying PHM to the shopfloor would be valuable for any organization. The use of metrics such as cost of poor quality (COPQ), return on investment (ROI), and payback would justify the use of AI for PHM to manufacturing executives. Therefore, incorporating the economic impact of PHM into health monitoring frameworks should be considered as an area for future work. In summary, there are several potential areas of research in health monitoring of systems and products, and the developments in AI in the coming years will hopefully promote such works.

Appendix

Case	Run	VB	Time	DOC	Feed	Material	smcAC	smcDC	vib_table	vib_spindle	AE_table	AE_spindle
1	1	0	2	1.5	0.5	1	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim
1	2	NaN	4	1.5	0.5	1	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim
1	3	NaN	6	1.5	0.5	1	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim
1	4	0.11	7	1.5	0.5	1	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim
1	5	NaN	11	1.5	0.5	1	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim	9000 × 1 dim

Table 15. Snapshot of the milling machine dataset.

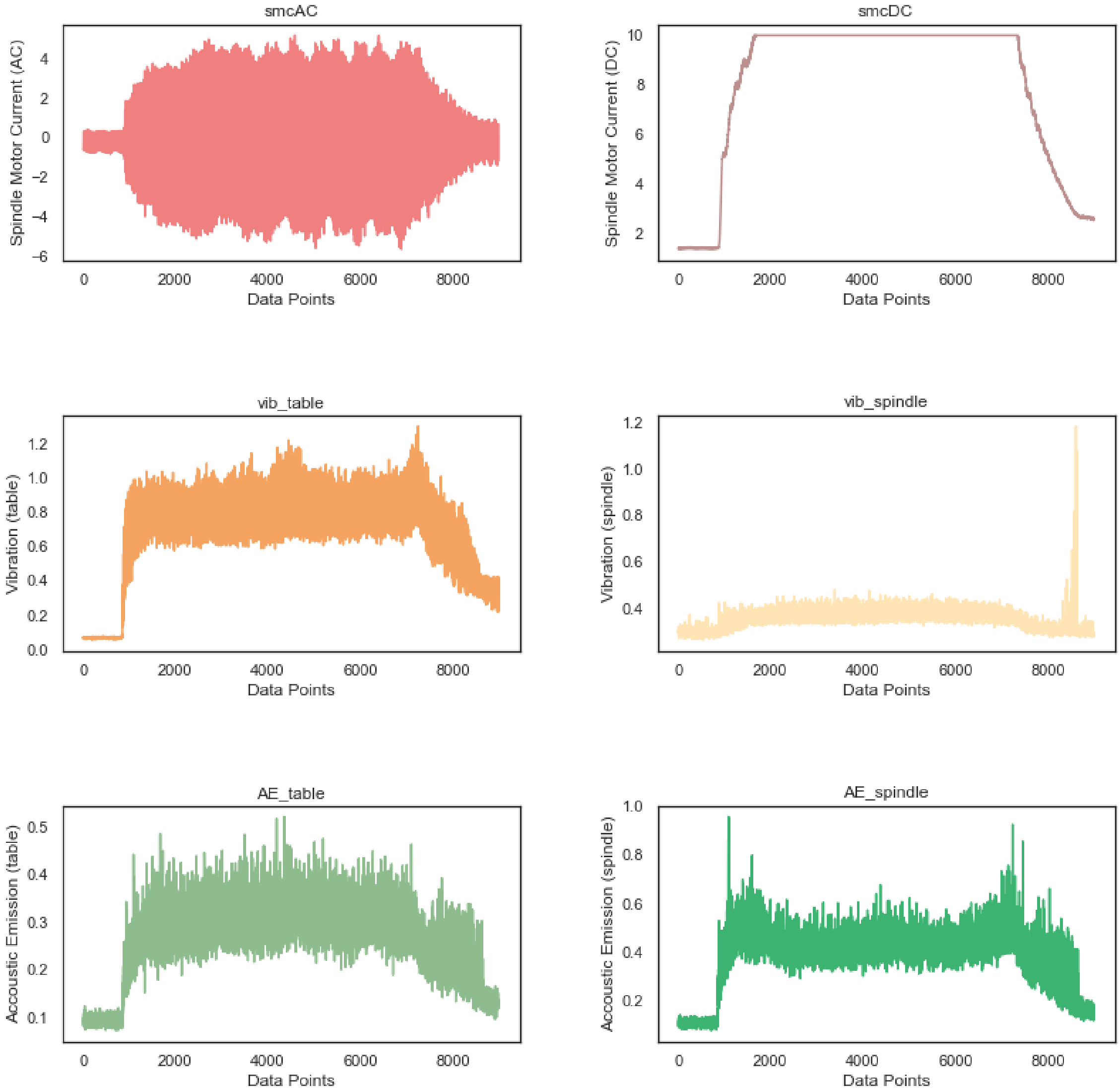

Figure 32. Signatures from the six sensor signals for case 1, run 11.

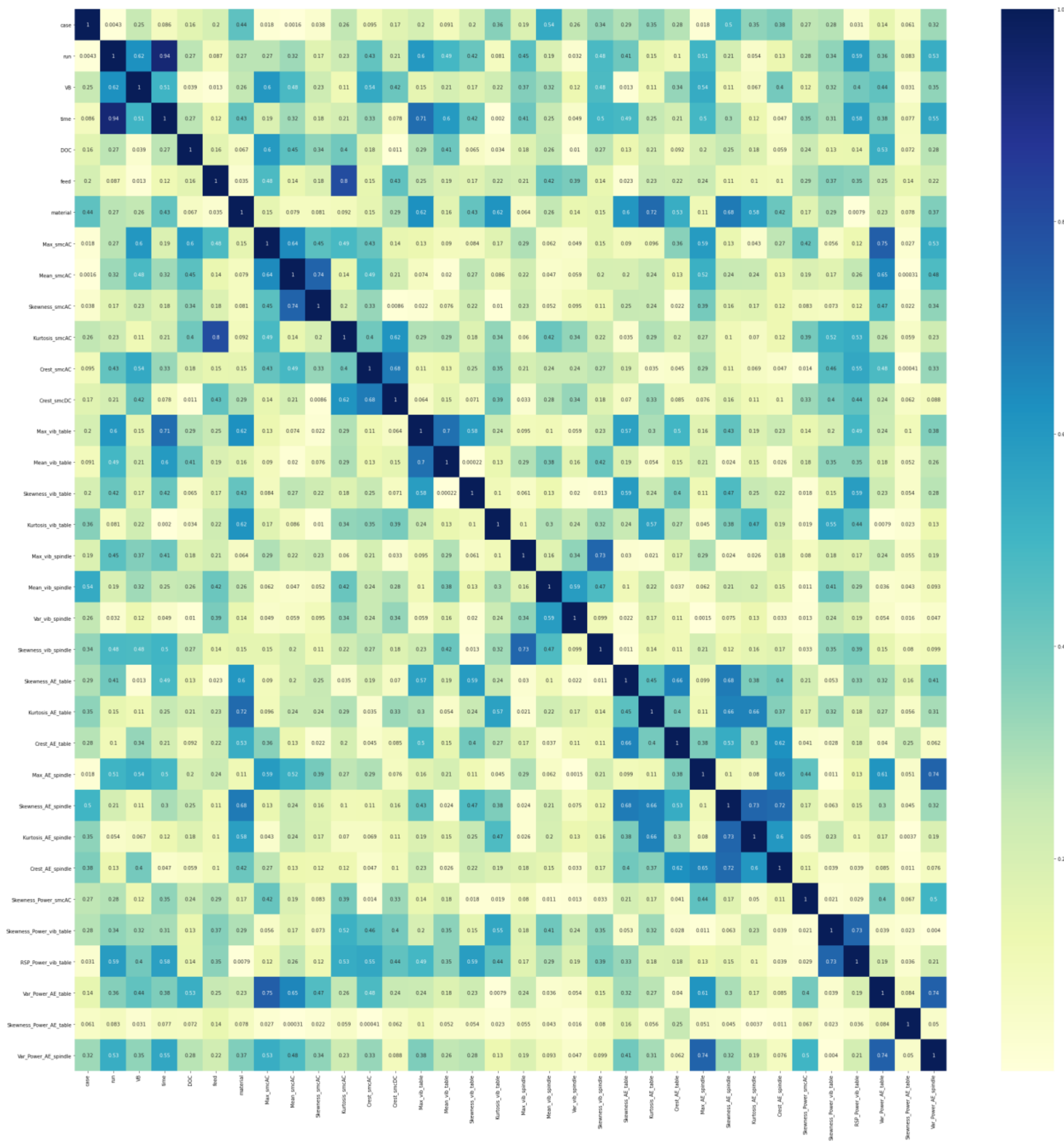

Figure 33. Correlation matrix of final set of features.

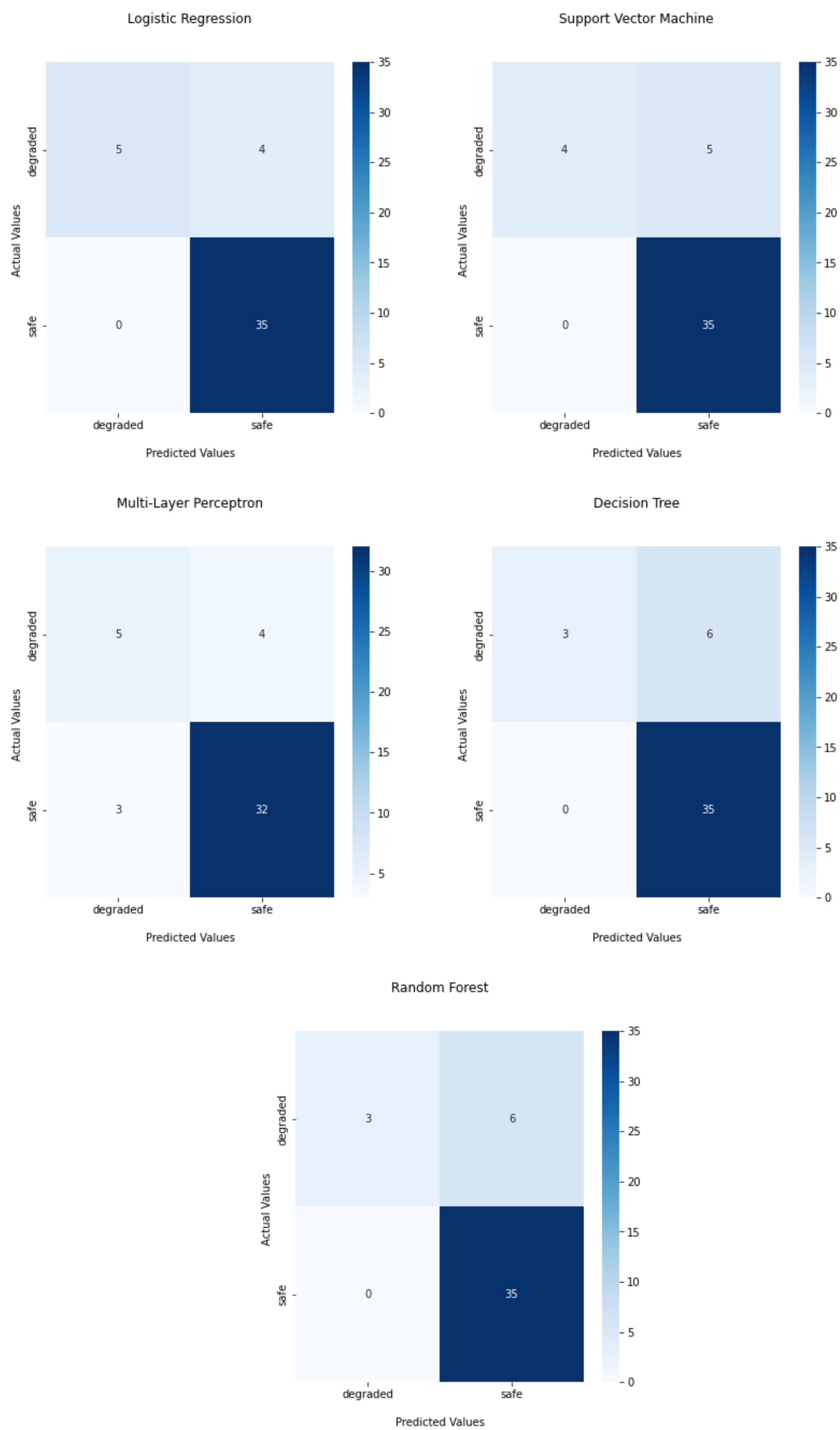

Figure 34. Confusion matrix of classification models.

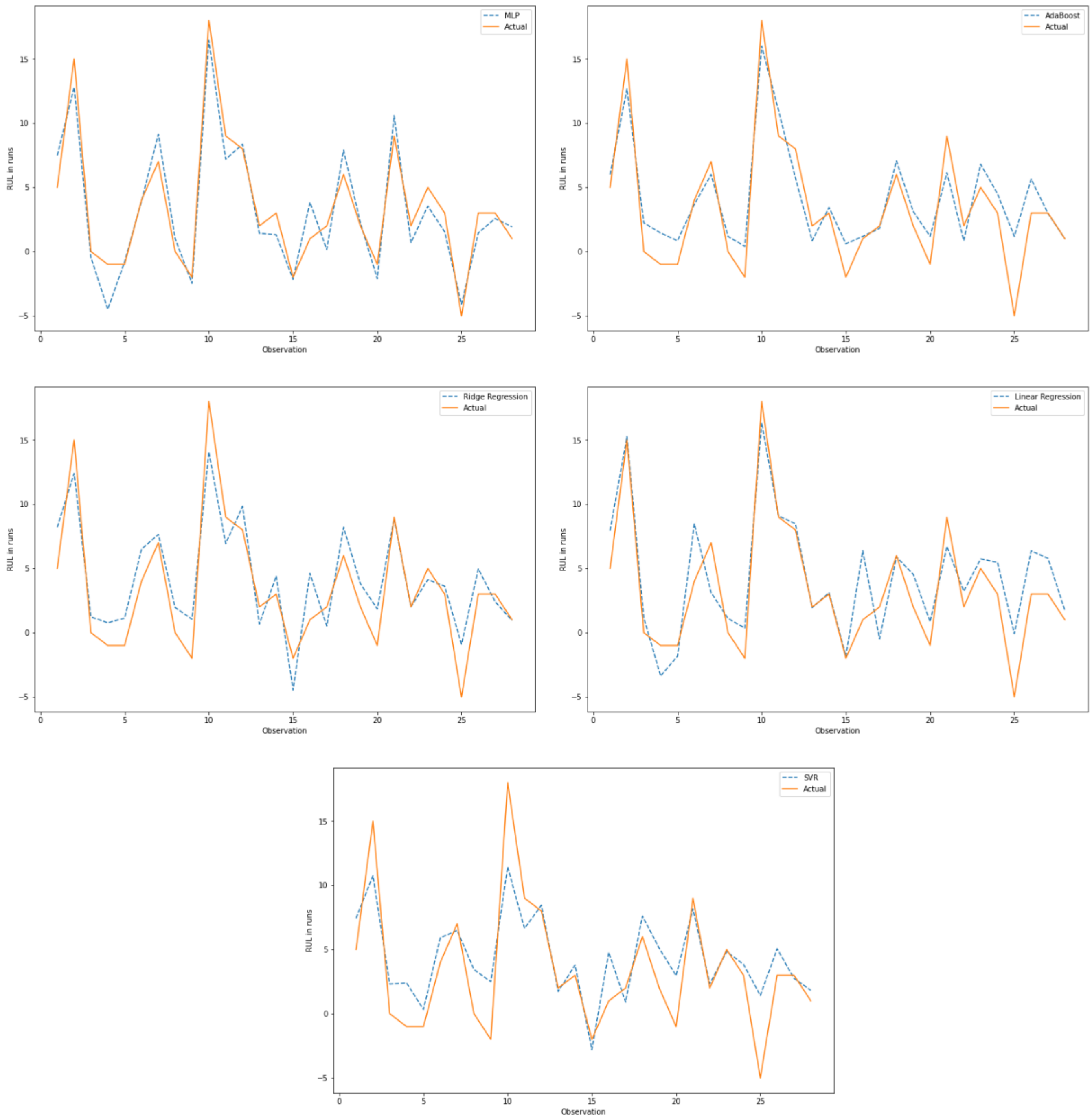

Figure 35. Actual and Predicted RUL values from regression task.

References

[1] Federal Ministry of Economic Affairs and Federal Ministry of Education and Research, “Platform Industry 4.0 - DIN SPEC 91345: 2016-04 - Reference architecture model Industry 4.0 (RAMI4.0),” 2016. https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/din-spec-rami40.html (accessed Jul. 16, 2020).

[2] “Made in China 2025 | Center for Strategic and International Studies.” https://www.csis.org/analysis/made-china-2025 (accessed May 18, 2019).

[3] J. F. Sargent, “Manufacturing USA,” in *National Network for Manufacturing Innovation: Background and Developments*, 2014, pp. 1–34. Accessed: May 18, 2019. [Online]. Available: https://digitalcommons.ilr.cornell.edu/key_workplace

[4] P. M. Krolikowski and K. Naggert, “Semiconductor shortages and vehicle production and prices,” *Economic Commentary*, no. 2021–17, 2021, doi: https://doi.org/10.26509/frbc-ec-202117.

[5] A. Garg and S. G. Deshmukh, “Maintenance management: literature review and directions,” *Journal of Quality in Maintenance Engineering*, vol. 12, no. 3, pp. 205–238, Jan. 2006, doi: 10.1108/13552510610685075.

[6] D. S. Thomas, “The Costs and Benefits of Advanced Maintenance in Manufacturing,” *National Institute of Standards and Technology*, no. NIST AMS 100-18, pp. 1–45, 2018, doi: 10.6028/NIST.AMS.100-18.

[7] D. S. Thomas and B. Weiss, “Maintenance Costs and Advanced Maintenance Techniques: Survey and Analysis,” *International Journal of Prognostics and Health Management*, vol. 12, no. 1, Apr. 2021, doi: 10.36001/ijphm.2021.v12i1.2883.

[8] V. Venkatasubramanian, “Prognostic and diagnostic monitoring of complex systems for product lifecycle management: Challenges and opportunities,” in *Computers and Chemical Engineering*, Elsevier Ltd, May 2005, pp. 1253–1263. doi: 10.1016/j.compchemeng.2005.02.026.

[9] L. M. Pintelon and L. F. Gelders, “Maintenance management decision making,” *European Journal of Operational Research*, vol. 58, no. 3, pp. 301–317, May 1992, doi: 10.1016/0377-2217(92)90062-E.

[10] B. G. Mwanza and C. Mbohwa, “Design of a total productive maintenance model for effective implementation: Case study of a chemical manufacturing company,” *Procedia Manufacturing*, vol. 4, pp. 461–470, 2015, doi: https://doi.org/10.1016/j.promfg.2015.11.063.

[11] A. Gunasekaran, N. Subramanian, and W. T. E. Ngai, “Quality management in the 21st century enterprises: Research pathway towards Industry 4.0,” *International Journal of Production Economics*, vol. 207, pp. 125–129, Jan. 2019, doi: 10.1016/j.ijpe.2018.09.005.

[12] A. Srinivasan and B. Kurey, “Creating a culture of quality,” *Harvard business review*, vol. 92, no. 4, pp. 23–25, 2014.

[13] O. Nir, “The Evolution Of Quality Assurance,” *Quality*, vol. 60, no. 13, p. 42, Dec. 2021, Accessed: Jul. 27, 2023. [Online]. Available: https://www.proquest.com/docview/2625352253/abstract/390E6B57FAA047F5PQ/1
[14] E. I. Basri, I. H. A. Razak, H. Ab-Samat, and S. Kamaruddin, “Preventive maintenance (PM) planning: a review,” *Journal of Quality in Maintenance Engineering*, vol. 23, no. 2, pp. 114–143, 2017, [Online]. Available: https://doi.org/10.1108/JQME-04-2016-0014
[15] Y. Kimura, “Maintenance tribology: its significance and activity in Japan,” *Wear*, vol. 207, no. 1–2, pp. 63–66, 1997, [Online]. Available: https://doi.org/10.1016/S0043-1648(96)07472-8
[16] R. K. Mobley, *An introduction to predictive maintenance*. Elsevier, 2002. [Online]. Available: https://doi.org/10.1016/B978-0-7506-7531-4.X5000-3
[17] W. Winchell, *Inspection and Measurement in Manufacturing: Keys to Process Planning and Improvement*. Society of Manufacturing Engineers, 1996.
[18] D. H. Harris, “The Nature of Industrial Inspection,” *Hum Factors*, vol. 11, no. 2, pp. 139–148, Apr. 1969, doi: 10.1177/001872086901100207.
[19] “Visual Inspection Reliability for Precision Manufactured Parts - Judi E. See, 2015.” https://journals.sagepub.com/doi/10.1177/0018720815602389 (accessed Jan. 10, 2023).
[20] General Electric, “The Impact of Digital on Unplanned Downtime,” Oct. 2016. https://www.ge.com/digital/sites/default/files/download_assets/ge-the-impact-of-digital-on-unplanned-downtime.pdf (accessed Jan. 10, 2022).
[21] Offices of Industries and Economics, “Small and medium-sized enterprises: Characteristics and performance,” Washington, DC, Investigation No. 332-510, 2010. [Online]. Available: https://www.usitc.gov/publications/332/pub4189.pdf
[22] X. Jin, B. A. Weiss, D. Siegel, and J. Lee, “Present status and future growth of advanced maintenance technology and strategy in US manufacturing,” *International journal of prognostics and health management*, vol. 7, no. Spec Iss on Smart Manufacturing PHM, 2016, doi: https://doi.org/10.36001/ijphm.2016.v7i3.2409.
[23] W. H. Murphy, “Small and mid-sized enterprises (SMEs) quality management (QM) research (1990–2014): a revealing look at QM’s vital role in making SMEs stronger,” *Journal of Small Business & Entrepreneurship*, vol. 28, no. 5, pp. 345–360, Sep. 2016, doi: 10.1080/08276331.2016.1166554.
[24] A. Ghobadian and D. Gallear, “TQM and organization size,” *International Journal of Operations & Production Management*, vol. 17, no. 2, pp. 121–163, Jan. 1997, doi: 10.1108/01443579710158023.
[25] B. El-Jawhari, S. Halbe, M. Whyte, K. Cobbaert, and A. Odenkirchen, “An introduction to implementing AI in manufacturing - PwC,” Global Manufacturing and Industrialisation Summit, 2020. [Online]. Available: https://www.pwc.com/gx/en/industrial-manufacturing/pdf/intro-implementing-ai-manufacturing.pdf
[26] C. Schmitz, A. Tschiesner, C. Jansen, S. Hallerstede, and F. Garms, “Industry 4.0: Capturing value at scale in discrete manufacturing,” McKinsey & Company, 2019.
[27] D. Küpper and K.-F. Seitz, “BCG-WEF Project: AI-Powered Industrial Operations.” Accessed: Jul. 27, 2023. [Online]. Available:

https://www.bcg.com/about/partner-ecosystem/world-economic-forum/ai-project-survey

[28] L. Monostori *et al.*, “Cyber-physical systems in manufacturing,” *Cirp Annals*, vol. 65, no. 2, pp. 621–641, 2016.

[29] S. Kemmerer J., “Manufacturing Interoperability Program, a Synopsis NISTIR 7533,” 2009, [Online]. Available: http://www.webcitation.org/746PeCi7k

[30] E. Traini, G. Bruno, G. D’Antonio, and F. Lombardi, “Machine learning framework for predictive maintenance in milling,” *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 177–182, Sep. 2019, doi: 10.1016/j.ifacol.2019.11.172.

[31] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, “Machinery health prognostics: A systematic review from data acquisition to RUL prediction,” *Mechanical Systems and Signal Processing*, vol. 104, pp. 799–834, May 2018, doi: 10.1016/j.ymssp.2017.11.016.

[32] T. Mohanraj, S. Shankar, R. Rajasekar, N. R. Sakthivel, and A. Pramanik, “Tool condition monitoring techniques in milling process-a review,” *Journal of Materials Research and Technology*, vol. 9, no. 1, pp. 1032–1042, Jan. 2020, doi: 10.1016/j.jmrt.2019.10.031.

[33] I. Shin *et al.*, “A Framework for Prognostics and Health Management Applications toward Smart Manufacturing Systems,” *International Journal of Precision Engineering and Manufacturing - Green Technology*, vol. 5, no. 4, pp. 535–554, Aug. 2018, doi: 10.1007/s40684-018-0055-0.

[34] Z. He and Q. Liu, “Deep Regression Neural Network for Industrial Surface Defect Detection,” *IEEE Access*, vol. 8, pp. 35583–35591, 2020, doi: 10.1109/ACCESS.2020.2975030.

[35] J. P. Yun, W. C. Shin, G. Koo, M. S. Kim, C. Lee, and S. J. Lee, “Automated defect inspection system for metal surfaces based on deep learning and data augmentation,” *Journal of Manufacturing Systems*, vol. 55, pp. 317–324, Apr. 2020, doi: 10.1016/j.jmsy.2020.03.009.

[36] P. Sassi, P. Tripicchio, and C. A. Avizzano, “A Smart Monitoring System for Automatic Welding Defect Detection,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9641–9650, Dec. 2019, doi: 10.1109/TIE.2019.2896165.

[37] P. Oborski and P. Wysocki, “Intelligent Visual Quality Control System Based on Convolutional Neural Networks for Holonic Shop Floor Control of Industry 4.0 Manufacturing Systems,” *Adv. Sci. Technol. Res. J.*, vol. 16, no. 2, pp. 89–98, Apr. 2022, doi: 10.12913/22998624/145503.

[38] K. J. Lee, J. W. Kwon, S. Min, and J. Yoon, “Deploying an Artificial Intelligence-Based Defect Finder for Manufacturing Quality Management,” *AI Magazine*, vol. 42, no. 2, Art. no. 2, Oct. 2021, doi: 10.1609/aimag.v42i2.15094.

[39] X. Chen, H. Xie, F. L. Wang, Z. Liu, J. Xu, and T. Hao, “A bibliometric analysis of natural language processing in medical research,” *BMC medical informatics and decision making*, vol. 18, no. 1, pp. 1–14, 2018, doi: https://doi.org/10.1186/s12911-018-0594-x.

[40] T. Le and H. David Jeong, “NLP-based approach to semantic classification of heterogeneous transportation asset data terminology,” *Journal of Computing in Civil Engineering*, vol. 31, no. 6, p. 04017057, 2017, doi: https://doi.org/10.1061/(ASCE)CP.1943-5487.0000701.

[41] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural language processing: State of the art, current trends and challenges,” *Multimedia tools and applications*, vol. 82, no. 3, pp. 3713–3744, 2023, doi: https://doi.org/10.1007/s11042-022-13428-4.
[42] Y. Lu, K. C. Morris, and S. Frechette, “Current standards landscape for smart manufacturing systems,” *National Institute of Standards and Technology, NISTIR*, vol. 8107, no. 3, 2016.
[43] X. Xu, “From cloud computing to cloud manufacturing,” *Robotics and computer-integrated manufacturing*, vol. 28, no. 1, pp. 75–86, 2012.
[44] S. R. Ray and A. T. Jones, “Manufacturing interoperability,” *Journal of intelligent Manufacturing*, vol. 17, pp. 681–688, 2006.
[45] M. Moghaddam, M. N. Cadavid, C. R. Kenley, and A. V. Deshmukh, “Reference architectures for smart manufacturing: A critical review,” *Journal of manufacturing systems*, vol. 49, pp. 215–225, 2018.
[46] E. A. Lee, “Cyber Physical Systems: Design Challenges,” in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, May 2008, pp. 363–369. doi: 10.1109/ISORC.2008.25.
[47] National Science Foundation, “Cyber-Physical Systems (CPS),” Arlington, VA, Program Solicitation NSF 10-515, 2010. Accessed: Jul. 13, 2023. [Online]. Available: https://www.nsf.gov/pubs/2010/nsf10515/nsf10515.htm
[48] J. Nolin and N. Olson, “The Internet of Things and convenience,” *Internet Research*, vol. 26, no. 2, pp. 360–376, 2016.
[49] F. Tao, Y. Cheng, L. D. Xu, L. Zhang, and B. H. Li, “CCIoT-CMfg: Cloud Computing and Internet of Things-Based Cloud Manufacturing Service System,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1435–1442, 2014, doi: 10.1109/TII.2014.2306383.
[50] ISO/IEC, “ISO/IEC 18384-1:2016 Information technology — Reference Architecture for Service Oriented Architecture (SOA RA) — Part 1: Terminology and concepts for SOA,” *ISO*, 2016. https://www.iso.org/standard/63104.html (accessed Jul. 13, 2023).
[51] N. Ivezic *et al.*, “2018 NISTOAGi workshop:: enabling composable service-oriented manufacturing systems,” National Institute of Standards and Technology, Gaithersburg, MD, NIST AMS 100-21, Apr. 2019. doi: 10.6028/NIST.AMS.100-21.
[52] T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, “Machine learning in manufacturing: advantages, challenges, and applications,” *Journal*, vol. 4, no. 1, pp. 23–45, Jan. 2016, doi: 10.1080/21693277.2016.1192517.
[53] J. A. Harding, M. Shahbaz, Srinivas, and A. Kusiak, “Data Mining in Manufacturing: A Review,” *Journal of Manufacturing Science and Engineering*, vol. 128, no. 4, pp. 969–976, 2005, doi: 10.1115/1.2194554.
[54] A. Radziwon, A. Bilberg, M. Bogers, and E. S. Madsen, “The smart factory: exploring adaptive and flexible manufacturing solutions,” *Procedia engineering*, vol. 69, pp. 1184–1190, 2014.
[55] L. Wang, M. Törngren, and M. Onori, “Current status and advancement of cyber-physical systems in manufacturing,” *Journal of manufacturing systems*, vol. 37, pp. 517–527, 2015.

[56] J. Lee, B. Bagheri, and C. Jin, "Introduction to cyber manufacturing," *Manufacturing Letters*, vol. 8, pp. 11–15, 2016.

[57] E. Hofmann and M. Rüsch, "Industry 4.0 and the current status as well as future prospects on logistics," *Computers in industry*, vol. 89, pp. 23–34, 2017.

[58] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, "Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation," *CAD Computer Aided Design*, vol. 59, pp. 1–14, Feb. 2015, doi: 10.1016/j.cad.2014.07.006.

[59] P. Jiang, K. Ding, and J. Leng, "Towards a cyber-physical-social-connected and service-oriented manufacturing paradigm: Social Manufacturing," *Manufacturing Letters*, vol. 7, pp. 15–21, 2016.

[60] ISO, "ISO 16100-6:2011 Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 6: Interface services and protocols for matching profiles based on multiple capability class structures," *ISO*, 2011. https://www.iso.org/standard/55213.html (accessed Jul. 13, 2023).

[61] "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries," *IEEE Std 610*, pp. 1–217, Jan. 1991, doi: 10.1109/IEEESTD.1991.106963.

[62] M. Kasunic, "Measuring systems interoperability: Challenges and opportunities," Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst, 2001.

[63] T. C. Ford, J. M. Colombi, S. R. Graham, and D. R. Jacques, "A survey on interoperability measurement," *Gateways*, vol. 2, p. 3, 2007.

[64] S. Heiler, "Semantic interoperability," *ACM Computing Surveys (CSUR)*, vol. 27, no. 2, pp. 271–273, 1995.

[65] R. J. Curts and D. E. Campbell, "Architecture: the road to interoperability," presented at the Command & Control Research & Technology Symposium (CCRTS), US Naval War College, Newport, RI, 1999.

[66] A.-J. Berre *et al.*, "State-of-the art for Interoperability architecture approaches," *InterOP Network of Excellence-Contract no.: IST-508*, vol. 11, 2004.

[67] C.-M. Chituc, C. Toscano, and A. Azevedo, "Interoperability in Collaborative Networks: Independent and industry-specific initiatives–The case of the footwear industry," *Computers in Industry*, vol. 59, no. 7, pp. 741–757, 2008.

[68] E. Morris, L. Levine, C. Meyers, P. Place, and D. Plakosh, "System of systems interoperability (sosi)," *Carnegie Mellon Univ., Software Engineering Institute*, 2004.

[69] D. Chen and F. B. Vernadat, "Enterprise Interoperability: A Standardisation View," in *Enterprise Inter- and Intra-Organizational Integration: Building International Consensus*, K. Kosanke, R. Jochem, J. G. Nell, and A. O. Bas, Eds., in IFIP — The International Federation for Information Processing. Boston, MA: Springer US, 2003, pp. 273–282. doi: 10.1007/978-0-387-35621-1_28.

[70] M. Mosley, M. H. Brackett, S. Earley, and D. Henderson, *DAMA guide to the data management body of knowledge (DAMA-DMBOK guide)*, 1st ed. Bradley Beach, N.J.: Technics Publications, 2009. Accessed: Jul. 13, 2023. [Online]. Available: https://proquest.safaribooksonline.com/9781935504009

[71] H. van der Veer and A. Wiles, "Achieving technical interoperability," *European telecommunications standards institute*, 2008.

[72] A. Shukla, S. Harris, and J. Davies, “Semantic interoperability in practice,” in *2010 43rd Hawaii International Conference on System Sciences*, IEEE, 2010, pp. 1–10.
[73] S. Decker *et al.*, “The Semantic Web: the roles of XML and RDF,” *IEEE Internet Computing*, vol. 4, no. 5, pp. 63–73, 2000, doi: 10.1109/4236.877487.
[74] F. Lelli, “Interoperability of the Time of Industry 4.0 and the Internet of Things,” *Future Internet*, vol. 11, no. 2, Art. no. 2, Feb. 2019, doi: 10.3390/fi11020036.
[75] S. Brunnermeier and S. A. Martin, *Interoperability Cost Analysis of the U. S. Automotive: Supply Chain: Final Report*. DIANE Publishing, 2000.
[76] D. Chen, G. Doumeingts, and F. Vernadat, “Architectures for enterprise integration and interoperability: Past, present and future,” *Computers in Industry*, vol. 59, no. 7, pp. 647–659, Sep. 2008, doi: 10.1016/j.compind.2007.12.016.
[77] ISO/IEC, “ISO/IEC 19941:2017 Information technology — Cloud computing — Interoperability and portability,” *ISO*, 2017. https://www.iso.org/standard/66639.html (accessed Jul. 13, 2023).
[78] R. T. Fielding, “Architectural styles and the design of network -based software architectures,” Ph.D., University of California, Irvine, California, 2000. Accessed: Jul. 13, 2023. [Online]. Available: https://www.proquest.com/docview/304591392/abstract/27294AFFEE0B4966PQ/1
[79] ISO/IEC, “ISO/IEC 20922:2016 Information technology — Message Queuing Telemetry Transport (MQTT) v3.1.1,” *ISO*, 2016. https://www.iso.org/standard/69466.html (accessed Jul. 13, 2023).
[80] M. Mourad, A. Nassehi, and D. Schaefer, “Interoperability as a Key Enabler for Manufacturing in the Cloud,” *Procedia CIRP*, vol. 52, pp. 30–34, Jan. 2016, doi: 10.1016/j.procir.2016.07.051.
[81] X. V. Wang and X. W. Xu, “ICMS: A Cloud-Based Manufacturing System,” in *Cloud Manufacturing: Distributed Computing Technologies for Global and Sustainable Manufacturing*, W. Li and J. Mehnen, Eds., in Springer Series in Advanced Manufacturing. London: Springer, 2013, pp. 1–22. doi: 10.1007/978-1-4471-4935-4_1.
[82] Y. Lu, X. Xu, and J. Xu, “Development of a Hybrid Manufacturing Cloud,” *Journal of Manufacturing Systems*, vol. 33, no. 4, pp. 551–566, Oct. 2014, doi: 10.1016/j.jmsy.2014.05.003.
[83] B. Li, X. Chai, B. Hou, L. Zhang, J. Zhou, and Y. Liu, “New Generation Artificial Intelligence-Driven Intelligent Manufacturing (NGAIIM),” in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Oct. 2018, pp. 1864–1869. doi: 10.1109/SmartWorld.2018.00313.
[84] X. Chai, B. Hou, P. Zou, J. Zeng, and jiehan Zhou, “INDICS: An Industrial Internet Platform,” in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Oct. 2018, pp. 1824–1828. doi: 10.1109/SmartWorld.2018.00307.

[85] A. P. Sage and W. B. Rouse, *Handbook of Systems Engineering and Management, 2nd Edition | Wiley*. 2014. Accessed: Jul. 13, 2023. [Online]. Available: https://www.wiley.com/en-us/Handbook+of+Systems+Engineering+and+Management%2C+2nd+Edition-p-9780470083536
[86] S.-W. Lin *et al.*, "Architecture Alignment and Interoperability - An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper," IIC:WHT:IN3:V1.0:PB:20171205, 2017. [Online]. Available: https://www.iiconsortium.org/pdf/JTG2_Whitepaper_final_20171205.pdf
[87] IEC, "IEC TR 62541-2:2016 OPC Unified Architecture - Part 2: Security Model," 2016. Accessed: Jul. 13, 2023. [Online]. Available: https://webstore.iec.ch/publication/25996
[88] IBM, "Announcing the IoT Industrie 4.0 reference architecture," 2017. https://www.ibm.com/cloud/blog/announcements/iot-industrie-40-reference-architecture (accessed Jul. 13, 2023).
[89] Y. Lu, F. Riddick, and N. Ivezic, "The Paradigm Shift in Smart Manufacturing System Architecture," in *Advances in Production Management Systems. Initiatives for a Sustainable World*, I. Nääs, O. Vendrametto, J. Mendes Reis, R. F. Gonçalves, M. T. Silva, G. von Cieminski, and D. Kiritsis, Eds., in IFIP Advances in Information and Communication Technology. Cham: Springer International Publishing, 2016, pp. 767–776. doi: 10.1007/978-3-319-51133-7_90.
[90] A. Lüder, M. Schleipen, N. Schmidt, J. Pfrommer, and R. Henßen, "One step towards an industry 4.0 component," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, Aug. 2017, pp. 1268–1273. doi: 10.1109/COASE.2017.8256275.
[91] R. Drath, A. Luder, J. Peschke, and L. Hundt, "AutomationML - the glue for seamless automation engineering," in *2008 IEEE International Conference on Emerging Technologies and Factory Automation*, Sep. 2008, pp. 616–623. doi: 10.1109/ETFA.2008.4638461.
[92] A. Zeid, S. Sundaram, M. Moghaddam, S. Kamarthi, and T. Marion, "Interoperability in Smart Manufacturing: Research Challenges," *Machines*, vol. 7, no. 2, Art. no. 2, Jun. 2019, doi: 10.3390/machines7020021.
[93] S. K. Pinjala, L. Pintelon, and A. Vereecke, "An empirical investigation on the relationship between business and maintenance strategies," *International Journal of Production Economics*, vol. 104, no. 1, pp. 214–229, Nov. 2006, doi: 10.1016/j.ijpe.2004.12.024.
[94] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel, "Prognostics and health management design for rotary machinery systems - Reviews, methodology and applications," *Mechanical Systems and Signal Processing*, vol. 42, no. 1–2, pp. 314–334, Jan. 2014, doi: 10.1016/j.ymssp.2013.06.004.
[95] V. Atamuradov, K. Medjaher, P. Dersin, B. Lamoureux, and N. Zerhouni, *Prognostics and Health Management for Maintenance Practitioners - Review, Implementation and Tools Evaluation*, vol. 8. 2017. Accessed: Jun. 10, 2021. [Online]. Available: https://oatao.univ-toulouse.fr/19521/
[96] G. W. Vogl, B. A. Weiss, and M. Helu, "A review of diagnostic and prognostic capabilities and best practices for manufacturing," *Journal of Intelligent*

Manufacturing, vol. 30, no. 1, pp. 79–95, Jan. 2019, doi: 10.1007/s10845-016-1228-8.
[97] H. Meng and Y. F. Li, "A review on prognostics and health management (PHM) methods of lithium-ion batteries," *Renewable and Sustainable Energy Reviews*, vol. 116, p. 109405, Dec. 2019, doi: 10.1016/j.rser.2019.109405.
[98] O. F. Eker, F. Camci, and I. K. Jennions, "Major Challenges in Prognostics: Study on Benchmarking Prognostics Datasets," PHM Society, Dec. 2012. Accessed: Jan. 03, 2021. [Online]. Available: http://dspace.lib.cranfield.ac.uk/handle/1826/9994
[99] H. M. Elattar, H. K. Elminir, and A. M. Riad, "Prognostics: a literature review," *Complex & Intelligent Systems*, vol. 2, no. 2, pp. 125–154, Jun. 2016, doi: 10.1007/s40747-016-0019-3.
[100] H. Sarih, A. P. Tchangani, K. Medjaher, and E. Pere, "Data preparation and preprocessing for broadcast systems monitoring in PHM framework," in *2019 6th International Conference on Control, Decision and Information Technologies, CoDIT 2019*, Institute of Electrical and Electronics Engineers Inc., Apr. 2019, pp. 1444–1449. doi: 10.1109/CoDIT.2019.8820370.
[101] A. Cubillo, S. Perinpanayagam, and M. Esperon-Miguez, "A review of physics-based models in prognostics: Application to gears and bearings of rotating machinery," *Advances in Mechanical Engineering*, vol. 8, no. 8, pp. 1–21, Aug. 2016, doi: 10.1177/1687814016664660.
[102] M. Pecht and Jie Gu, "Physics-of-failure-based prognostics for electronic products," *Transactions of the Institute of Measurement and Control*, vol. 31, no. 3–4, pp. 309–322, Jun. 2009, doi: 10.1177/0142331208092031.
[103] Y. H. Lui *et al.*, "Physics-based prognostics of implantable-grade lithium-ion battery for remaining useful life prediction," *Journal of Power Sources*, vol. 485, p. 229327, Feb. 2021, doi: 10.1016/j.jpowsour.2020.229327.
[104] D. Bradley, C. Ortega-Sanchez, and A. Tyrrell, "Embryonics + immunotronics: A bio-inspired approach to fault tolerance," in *Proceedings - NASA/DoD Conference on Evolvable Hardware, EH*, Institute of Electrical and Electronics Engineers Inc., 2000, pp. 215–223. doi: 10.1109/EH.2000.869359.
[105] H. Dong, X. Yang, A. Li, Z. Xie, and Y. Zuo, "Bio-inspired PHM model for diagnostics of faults in power transformers using dissolved gas-in-oil data," *Sensors (Switzerland)*, vol. 19, no. 4, p. 845, Feb. 2019, doi: 10.3390/s19040845.
[106] A. Soualhi, H. Razik, G. Clerc, and D. D. Doan, "Prognosis of bearing failures using hidden markov models and the adaptive neuro-fuzzy inference system," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 6, pp. 2864–2874, 2014, doi: 10.1109/TIE.2013.2274415.
[107] M. Moghaddam, Q. Chen, and A. V. Deshmukh, "A neuro-inspired computational model for adaptive fault diagnosis," *Expert Systems with Applications*, vol. 140, p. 112879, Feb. 2020, doi: 10.1016/j.eswa.2019.112879.
[108] B. Huang, Y. Di, C. Jin, and J. Lee, "REVIEW OF DATA-DRIVEN PROGNOSTICS AND HEALTH MANAGEMENT TECHNIQUES: LESSIONS LEARNED FROM PHM DATA CHALLENGE COMPETITIONS," 2017.
[109] X. Jia, B. Huang, J. Feng, H. Cai, and J. Lee, "Review of PHM Data Competitions from 2008 to 2017," *Annual Conference of the PHM Society*, vol. 10, no. 1, Sep. 2018, doi: 10.36001/phmconf.2018.v10i1.462.

[110] H. Z. Huang, H. K. Wang, Y. F. Li, L. Zhang, and Z. Liu, "Support vector machine based estimation of remaining useful life: current research status and future trends," *Journal of Mechanical Science and Technology*, vol. 29, no. 1, pp. 151–163, Jan. 2015, doi: 10.1007/s12206-014-1222-z.

[111] V. Mathew, T. Toby, V. Singh, B. M. Rao, and M. G. Kumar, "Prediction of Remaining Useful Lifetime (RUL) of turbofan engine using machine learning," in *IEEE International Conference on Circuits and Systems, ICCS 2017*, Institute of Electrical and Electronics Engineers Inc., Mar. 2018, pp. 306–311. doi: 10.1109/ICCS1.2017.8326010.

[112] M. Elforjani and S. Shanbr, "Prognosis of Bearing Acoustic Emission Signals Using Supervised Machine Learning," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5864–5871, Jul. 2018, doi: 10.1109/TIE.2017.2767551.

[113] S. S. Mansouri, P. Karvelis, G. Georgoulas, and G. Nikolakopoulos, "Remaining Useful Battery Life Prediction for UAVs based on Machine Learning," in *IFAC-PapersOnLine*, Elsevier B.V., Jul. 2017, pp. 4727–4732. doi: 10.1016/j.ifacol.2017.08.863.

[114] S. Cho, S. Asfour, A. Onar, and N. Kaundinya, "Tool breakage detection using support vector machine learning in a milling process," *International Journal of Machine Tools and Manufacture*, vol. 45, no. 3, pp. 241–249, Mar. 2005, doi: 10.1016/j.ijmachtools.2004.08.016.

[115] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mechanical Systems and Signal Processing*, vol. 138, p. 106587, Apr. 2020, doi: 10.1016/j.ymssp.2019.106587.

[116] T. S. Abdelgayed, W. G. Morsi, and T. S. Sidhu, "Fault detection and classification based on co-training of semisupervised machine learning," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1595–1605, Jul. 2017, doi: 10.1109/TIE.2017.2726961.

[117] X. Wang, H. Feng, and Y. Fan, "Fault detection and classification for complex processes using semi-supervised learning algorithm," *Chemometrics and Intelligent Laboratory Systems*, vol. 149, pp. 24–32, Dec. 2015, doi: 10.1016/j.chemolab.2015.10.019.

[118] K. Yan, C. Zhong, Z. Ji, and J. Huang, "Semi-supervised learning for early detection and diagnosis of various air handling unit faults," *Energy and Buildings*, vol. 181, pp. 75–83, Dec. 2018, doi: 10.1016/j.enbuild.2018.10.016.

[119] A. Malhi, R. Yan, and R. X. Gao, "Prognosis of defect propagation based on recurrent neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 703–711, Mar. 2011, doi: 10.1109/TIM.2010.2078296.

[120] F. O. Heimes, "Recurrent neural networks for remaining useful life estimation," in *2008 International Conference on Prognostics and Health Management, PHM 2008*, 2008. doi: 10.1109/PHM.2008.4711422.

[121] A. S. Palau, K. Bakliwal, M. H. Dhada, T. Pearce, and A. K. Parlikad, "Recurrent Neural Networks for real-time distributed collaborative prognostics," in *2018 IEEE International Conference on Prognostics and Health Management, ICPHM 2018*, Institute of Electrical and Electronics Engineers Inc., Aug. 2018. doi: 10.1109/ICPHM.2018.8448622.

[122] N. Gugulothu, V. TV, P. Malhotra, L. Vig, P. Agarwal, and G. Shroff, “Predicting Remaining Useful Life using Time Series Embeddings based on Recurrent Neural Networks,” *International Journal of Prognostics and Health Management*, vol. 9, no. 1, Sep. 2017, Accessed: Jun. 11, 2021. [Online]. Available: http://arxiv.org/abs/1709.01073

[123] J. Deutsch and D. He, “Using deep learning-based approach to predict remaining useful life of rotating components,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 1, pp. 11–20, Jan. 2018, doi: 10.1109/TSMC.2017.2697842.

[124] G. Zhao, G. Zhang, Y. Liu, B. Zhang, and C. Hu, “Lithium-ion battery remaining useful life prediction with Deep Belief Network and Relevance Vector Machine,” in *2017 IEEE International Conference on Prognostics and Health Management, ICPHM 2017*, Institute of Electrical and Electronics Engineers Inc., Jul. 2017, pp. 7–13. doi: 10.1109/ICPHM.2017.7998298.

[125] C. Zhang, P. Lim, A. K. Qin, and K. C. Tan, “Multiobjective Deep Belief Networks Ensemble for Remaining Useful Life Estimation in Prognostics,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306–2318, Oct. 2017, doi: 10.1109/TNNLS.2016.2582798.

[126] L. Liao, W. Jin, and R. Pavel, “Enhanced Restricted Boltzmann Machine with Prognosability Regularization for Prognostics and Health Assessment,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7076–7083, Nov. 2016, doi: 10.1109/TIE.2016.2586442.

[127] W. Sun, R. Zhao, R. Yan, S. Shao, and X. Chen, “Convolutional Discriminative Feature Learning for Induction Motor Fault Diagnosis,” *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1350–1359, Jun. 2017, doi: 10.1109/TII.2017.2672988.

[128] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, “A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals,” *Sensors (Switzerland)*, vol. 17, no. 2, p. 425, Feb. 2017, doi: 10.3390/s17020425.

[129] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, “Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks,” *Journal of Sound and Vibration*, vol. 388, pp. 154–170, Feb. 2017, doi: 10.1016/j.jsv.2016.10.043.

[130] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, “Real-Time Motor Fault Detection by 1-D Convolutional Neural Networks,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 11, pp. 7067–7075, Nov. 2016, doi: 10.1109/TIE.2016.2582729.

[131] R. Liu, G. Meng, B. Yang, C. Sun, and X. Chen, “Dislocated Time Series Convolutional Neural Architecture: An Intelligent Fault Diagnosis Approach for Electric Machine,” *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1310–1320, Jun. 2017, doi: 10.1109/TII.2016.2645238.

[132] O. Janssens *et al.*, “Convolutional Neural Network Based Fault Detection for Rotating Machinery,” *Journal of Sound and Vibration*, vol. 377, pp. 331–345, Sep. 2016, doi: 10.1016/j.jsv.2016.05.027.

[133] G. S. Babu, P. Zhao, and X. L. Li, “Deep convolutional neural network based regression approach for estimation of remaining useful life,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2016, pp. 214–228. doi: 10.1007/978-3-319-32025-0_14.

[134] J. Wang, J. Zhuang, L. Duan, and W. Cheng, “A multi-scale convolution neural network for featureless fault diagnosis,” in *International Symposium on Flexible Automation, ISFA 2016*, Institute of Electrical and Electronics Engineers Inc., Dec. 2016, pp. 65–70. doi: 10.1109/ISFA.2016.7790137.

[135] W. You, C. Shen, X. Guo, X. Jiang, J. Shi, and Z. Zhu, “A hybrid technique based on convolutional neural network and support vector regression for intelligent diagnosis of rotating machinery,” *Advances in Mechanical Engineering*, vol. 9, no. 6, p. 2017, Jun. 2017, doi: 10.1177/1687814017704146.

[136] D. Weimer, B. Scholz-Reiter, and M. Shpitalni, “Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection,” *CIRP Annals - Manufacturing Technology*, vol. 65, no. 1, pp. 417–420, Jan. 2016, doi: 10.1016/j.cirp.2016.04.072.

[137] S. Khan and T. Yairi, “A review on the application of deep learning in system health management,” *Mechanical Systems and Signal Processing*, vol. 107, pp. 241–265, Jul. 2018, doi: 10.1016/j.ymssp.2017.11.024.

[138] Y. Liu, X. Hu, and W. Zhang, “Remaining useful life prediction based on health index similarity,” *Reliability Engineering and System Safety*, vol. 185, pp. 502–510, May 2019, doi: 10.1016/j.ress.2019.02.002.

[139] D. Liu, H. Wang, Y. Peng, W. Xie, and H. Liao, “Satellite lithium-ion battery remaining cycle life prediction with novel indirect health indicator extraction,” *Energies*, vol. 6, no. 8, pp. 3654–3668, Jul. 2013, doi: 10.3390/en6083654.

[140] F. Yang, M. S. Habibullah, T. Zhang, Z. Xu, P. Lim, and S. Nadarajan, “Health index-based prognostics for remaining useful life predictions in electrical machines,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 4, pp. 2633–2644, Apr. 2016, doi: 10.1109/TIE.2016.2515054.

[141] T. Xia, Y. Dong, L. Xiao, S. Du, E. Pan, and L. Xi, “Recent advances in prognostics and health management for advanced manufacturing paradigms,” *Reliability Engineering & System Safety*, vol. 178, pp. 255–268, Oct. 2018, doi: 10.1016/J.RESS.2018.06.021.

[142] T. Blecker and G. Friedrich, “Guest editorial: Mass customization manufacturing systems,” *IEEE Transactions on Engineering Management*, vol. 54, no. 1, pp. 4–11, Feb. 2007, doi: 10.1109/TEM.2006.889063.

[143] X. Jin and J. Ni, “Joint Production and Preventive Maintenance Strategy for Manufacturing Systems With Stochastic Demand,” *Journal of Manufacturing Science and Engineering*, vol. 135, no. 3, Jun. 2013, doi: 10.1115/1.4024042.

[144] M. C. Fitouhi and M. Nourelfath, “Integrating noncyclical preventive maintenance scheduling and production planning for multi-state systems,” *Reliability Engineering & System Safety*, vol. 121, pp. 175–186, Jan. 2014, doi: 10.1016/J.RESS.2013.07.009.

[145] Y. Koren *et al.*, “Reconfigurable Manufacturing Systems,” *CIRP Annals*, vol. 48, no. 2, pp. 527–540, Jan. 1999, doi: 10.1016/S0007-8506(07)63232-6.

[146] T. Xia, L. Xi, E. Pan, and J. Ni, "Reconfiguration-oriented opportunistic maintenance policy for reconfigurable manufacturing systems," *Reliability Engineering & System Safety*, vol. 166, pp. 87–98, Oct. 2017, doi: 10.1016/J.RESS.2016.09.001.

[147] J. Zhou, D. Djurdjanovic, J. Ivy, and J. Ni, "Integrated reconfiguration and age-based preventive maintenance decision making," *http://dx.doi.org/10.1080/07408170701291779*, vol. 39, no. 12, pp. 1085–1102, Dec. 2007, doi: 10.1080/07408170701291779.

[148] Y. Koren, X. Gu, F. Badurdeen, and I. S. Jawahir, "Sustainable Living Factories for Next Generation Manufacturing," *Procedia Manufacturing*, vol. 21, pp. 26–36, Jan. 2018, doi: 10.1016/J.PROMFG.2018.02.091.

[149] J. Gao, Y. Yao, V. C. Y. Zhu, L. Sun, and L. Lin, "Service-oriented manufacturing: a new product pattern and manufacturing paradigm," *Journal of Intelligent Manufacturing 2009 22:3*, vol. 22, no. 3, pp. 435–446, Jul. 2009, doi: 10.1007/S10845-009-0301-Y.

[150] D. Ning, J. Huang, J. Shen, and D. Di, "A cloud based framework of prognostics and health management for manufacturing industry," *2016 IEEE International Conference on Prognostics and Health Management, ICPHM 2016*, Aug. 2016, doi: 10.1109/ICPHM.2016.7542871.

[151] R. Wirth, "CRISP-DM: Towards a standard process model for data mining," in *Proceedings of The Fourth International Conference On The Practical Application of Knowledge Discovery and Data Mining*, 2000, pp. 29--39. Accessed: May 03, 2021. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.198.5133

[152] S. van Buuren, *Flexible Imputation of Missing Data, Second Edition*. Chapman and Hall/CRC, 2018. doi: 10.1201/9780429492259.

[153] B. Efron, "Missing data, imputation, and the bootstrap," *Journal of the American Statistical Association*, vol. 89, no. 426, pp. 463–475, 1994, doi: 10.1080/01621459.1994.10476768.

[154] F. Miao, R. Zhao, and X. Wang, "A New Method of Denoising of Vibration Signal and Its Application," *Shock and Vibration*, vol. 2020, 2020, doi: 10.1155/2020/7587840.

[155] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, vol. 37, no. 4. USA: Prentice-Hall, Inc., 1993. doi: 10.2307/1269750.

[156] A. Siddhpura and R. Paurobally, "A review of flank wear prediction methods for tool condition monitoring in a turning process," *International Journal of Advanced Manufacturing Technology*, vol. 65, no. 1–4, pp. 371–393, Mar. 2013, doi: 10.1007/s00170-012-4177-1.

[157] C. Zhang, X. Yao, J. Zhang, and H. Jin, "Tool condition monitoring and remaining useful life prognostic based on awireless sensor in dry milling operations," *Sensors (Switzerland)*, vol. 16, no. 6, Jun. 2016, doi: 10.3390/s16060795.

[158] W. Caesarendra and T. Tjahjowidodo, "A Review of Feature Extraction Methods in Vibration-Based Condition Monitoring and Its Application for Degradation Trend Estimation of Low-Speed Slew Bearing," *Machines*, vol. 5, no. 4, p. 21, Sep. 2017, doi: 10.3390/machines5040021.

[159] J. O. Smith, *Mathematics of the Discrete Fourier Transform (DFT)*. Online Book, 2002. Accessed: Jun. 20, 2021. [Online]. Available: http://ccrma.stanford.edu/~jos/mdft/Fourier_Theorems_DFT.html

[160] J. O. Smith, *"Periodogram" Spectral Audio Signal Processing*. Online Book, 2011. [Online]. Available: https://ccrma.stanford.edu/~jos/sasp/Periodogram.html

[161] Q. Zhu, Y. Wang, and G. Shen, "Comparison and application of time-frequency analysis methods for nonstationary signal processing," in *Communications in Computer and Information Science*, Springer, Berlin, Heidelberg, 2011, pp. 286–291. doi: 10.1007/978-3-642-21783-8_47.

[162] A. Bommert, X. Sun, B. Bischl, J. Rahnenführer, and M. Lang, "Benchmark for filter methods for feature selection in high-dimensional classification data," *Computational Statistics and Data Analysis*, vol. 143, p. 106839, Mar. 2020, doi: 10.1016/j.csda.2019.106839.

[163] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[164] K. Goebel and A. Agogino, "Documentation for Mill Data Set," *BEST Lab, UC Berkeley*, 2007, [Online]. Available: http://ti.arc.nasa.gov/project/prognostic-data-repository

[165] K. F. Goebel, "Management of uncertainty in sensor validation, sensor fusion, and diagnosis of mechanical systems using soft computing techniques," University of California, Berkeley, 1996. [Online]. Available: https://www.proquest.com/docview/304224063?pq-origsite=gscholar&fromopenview=true#

[166] MATLAB, "MATLAB 2020," *The MathWorks Inc.* Natick, Massachusetts, p. 1, 2021. Accessed: Jun. 08, 2021. [Online]. Available: https://nl.mathworks.com/products/matlab.html%0Ahttp://www.mathworks.com/products/matlab/

[167] ISO, "ISO 3685:1993 Tool-life testing with single-point turning tools." 1993. Accessed: Jul. 20, 2023. [Online]. Available: https://www.iso.org/standard/9151.html

[168] D. Krstajic, L. J. Buturovic, D. E. Leahy, and S. Thomas, "Cross-validation pitfalls when selecting and assessing regression and classification models," *Journal of Cheminformatics*, vol. 6, no. 1, p. 10, Mar. 2014, doi: 10.1186/1758-2946-6-10.

[169] B. Tang, J. Kong, X. Wang, and L. Chen, "Surface Inspection System of Steel Strip Based on Machine Vision," in *2009 First International Workshop on Database Technology and Applications*, Apr. 2009, pp. 359–362. doi: 10.1109/DBTA.2009.133.

[170] J. Jia, "A Machine Vision Application for Industrial Assembly Inspection," in *2009 Second International Conference on Machine Vision*, Dec. 2009, pp. 172–176. doi: 10.1109/ICMV.2009.51.

[171] A. R. Rao, "Future directions in industrial machine vision: a case study of semiconductor manufacturing applications," *Image and Vision Computing*, vol. 14, no. 1, pp. 3–19, Feb. 1996, doi: 10.1016/0262-8856(95)01035-1.

[172] N. N.J. and A. Kadan, "Role of Computer Vision in Automatic Inspection Systems," *International Journal of Computer Applications*, vol. 123, pp. 28–31, Aug. 2015, doi: 10.5120/ijca2015905603.

[173] S. Sundaram and A. Zeid, “Smart Prognostics and Health Management (SPHM) in Smart Manufacturing: An Interoperable Framework,” *Sensors*, vol. 21, no. 18, Art. no. 18, Jan. 2021, doi: 10.3390/s21185994.

[174] O. Fink, Q. Wang, M. Svensén, P. Dersin, W.-J. Lee, and M. Ducoffe, “Potential, challenges and future directions for deep learning in prognostics and health management applications,” *Engineering Applications of Artificial Intelligence*, vol. 92, p. 103678, Jun. 2020, doi: 10.1016/j.engappai.2020.103678.

[175] H. Brüggemann and P. Bremer, *Grundlagen Qualitätsmanagement: Von den Werkzeugen über Methoden zum TQM*. Wiesbaden: Springer Fachmedien, 2020. doi: 10.1007/978-3-658-28780-1.

[176] S. Borkowski and K. Knop, “Challenges Faced in Modern Quality Inspection,” *Management and Production Engineering Review*, vol. 7, no. 3, pp. 11–22, 2016, doi: 10.1515/mper-2016-0022.

[177] C. G. Drury, “Human Factors and Automation in Test and Inspection,” in *Handbook of Industrial Engineering*, John Wiley & Sons, Ltd, 2001, pp. 1887–1920. doi: 10.1002/9780470172339.ch71.

[178] J. A. Pesante-Santana and J. Woldstad, “Quality Inspection Task in Modern Manufacturing,” *Industrial and Management Systems Engineering Faculty Publications*, Jan. 2000, [Online]. Available: https://digitalcommons.unl.edu/imsefacpub/11

[179] M. Sinclair, “Ergonomics of quality control,” presented at the Workshop document, International Conference on Occupational Ergonomics, Toronto, 1984.

[180] M. J. Wang and C. G. Drury, “A method of evaluating inspector’s performance differences and job requirements,” *Applied Ergonomics*, vol. 20, no. 3, pp. 181–190, Sep. 1989, doi: 10.1016/0003-6870(89)90075-6.

[181] T. J. GALLWEY, “Selection tests for visual inspection on a multiple fault type task,” *Ergonomics*, vol. 25, no. 11, pp. 1077–1092, Nov. 1982, doi: 10.1080/00140138208925066.

[182] F. Peters, R. Stone, K. Watts, P. Zhong, and A. Clemons, “Visual Inspection of Casting Surfaces,” Apr. 2013. Accessed: Jan. 12, 2023. [Online]. Available: https://www.semanticscholar.org/paper/Visual-Inspection-of-Casting-Surfaces-Peters-Stone/063619079d27b4620605c713dc08087eaf2174e9

[183] J. E. See, C. G. Drury, A. Speed, A. Williams, and N. Khalandi, “The Role of Visual Inspection in the 21st Century,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 61, no. 1, pp. 262–266, Sep. 2017, doi: 10.1177/1541931213601548.

[184] H. N. Gupta, *Manufacturing Process*, 2nd ed. Daryaganj, New Delhi: New Age International Ltd., 2009.

[185] I. Rajkumar and N. Rajini, “Influence of parameters on the smart productivity of modern metal casting process: An overview,” *Materials Today: Proceedings*, vol. 47, pp. 2694–2700, Jan. 2021, doi: 10.1016/j.matpr.2021.03.003.

[186] M. Sahoo and S. “Sam” Sahu, “Principles of Metal Casting,” *AccessEngineering | McGraw-Hill Education - Access Engineering*, 2014. https://www.accessengineeringlibrary.com/content/book/9780071789752 (accessed Jan. 10, 2023).

[187] A. Alagarsamy, "Casting defect analysis procedure and a case history," presented at the PT Keith Millis Symposium on Ductile Iron Castings, Citation Corporation, Birmingham, Alabama, October, 2003, pp. 20–23.
[188] P. Gamage and S. Q. Xie, "A real-time vision system for defect inspection in cast extrusion manufacturing process," *Int J Adv Manuf Technol*, vol. 40, no. 1, pp. 144–156, Jan. 2009, doi: 10.1007/s00170-007-1326-z.
[189] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *Journal of Manufacturing Systems*, vol. 48, pp. 144–156, Jul. 2018, doi: 10.1016/j.jmsy.2018.01.003.
[190] J. Yang, S. Li, Z. Wang, H. Dong, J. Wang, and S. Tang, "Using Deep Learning to Detect Defects in Manufacturing: A Comprehensive Survey and Current Challenges," *Materials*, vol. 13, no. 24, Art. no. 24, Jan. 2020, doi: 10.3390/ma13245755.
[191] F. Chang, M. Liu, M. Dong, and Y. Duan, "A mobile vision inspection system for tiny defect detection on smooth car-body surfaces based on deep ensemble learning," *Meas. Sci. Technol.*, vol. 30, no. 12, p. 125905, Sep. 2019, doi: 10.1088/1361-6501/ab1467.
[192] T. Wang, Y. Chen, M. Qiao, and H. Snoussi, "A fast and robust convolutional neural network-based defect detection model in product quality control," *Int J Adv Manuf Technol*, vol. 94, no. 9, pp. 3465–3471, Feb. 2018, doi: 10.1007/s00170-017-0882-0.
[193] K. Imoto, T. Nakai, T. Ike, K. Haruki, and Y. Sato, "A CNN-Based Transfer Learning Method for Defect Classification in Semiconductor Manufacturing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 32, no. 4, pp. 455–459, Nov. 2019, doi: 10.1109/TSM.2019.2941752.
[194] K. B. Lee, S. Cheon, and C. O. Kim, "A Convolutional Neural Network for Fault Classification and Diagnosis in Semiconductor Manufacturing Processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 30, no. 2, pp. 135–142, May 2017, doi: 10.1109/TSM.2017.2676245.
[195] S. S. Kumar, D. M. Abraham, M. R. Jahanshahi, T. Iseley, and J. Starr, "Automated defect classification in sewer closed circuit television inspections using deep convolutional neural networks," *Automation in Construction*, vol. 91, pp. 273–283, Jul. 2018, doi: 10.1016/j.autcon.2018.03.028.
[196] Y. Yang *et al.*, "A High-Performance Deep Learning Algorithm for the Automated Optical Inspection of Laser Welding," *Applied Sciences*, vol. 10, no. 3, Art. no. 3, Jan. 2020, doi: 10.3390/app10030933.
[197] I. Ullah, R. U. Khan, F. Yang, and L. Wuttisittikulkij, "Deep Learning Image-Based Defect Detection in High Voltage Electrical Equipment," *Energies*, vol. 13, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/en13020392.
[198] O. S. Amosov, S. G. Amosova, and I. O. Iochkov, "Deep Neural Network Recognition of Rivet Joint Defects in Aircraft Products," *Sensors*, vol. 22, no. 9, Art. no. 9, Jan. 2022, doi: 10.3390/s22093417.
[199] S.-H. Park, K.-H. Lee, J.-S. Park, and Y.-S. Shin, "Deep Learning-Based Defect Detection for Sustainable Smart Manufacturing," *Sustainability*, vol. 14, no. 5, Art. no. 5, Jan. 2022, doi: 10.3390/su14052697.

[200] Dabhi, Ravirajsinh, “Casting Product Image Data for Quality Inspection.” https://www.kaggle.com/datasets/ravirajsinh45/real-life-industrial-dataset-of-casting-product (accessed Jan. 10, 2023).

[201] V. Nair and G. E. Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines”.

[202] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization.” arXiv, Jan. 29, 2017. doi: 10.48550/arXiv.1412.6980.

[203] Tensorflow by Google, “tf.keras.losses.SparseCategoricalCrossentropy | TensorFlow v2.11.0,” *TensorFlow*. https://www.tensorflow.org/api_docs/python/tf/keras/losses/SparseCategoricalCrossentropy (accessed Jan. 17, 2023).

[204] A. Zafar *et al.*, “A Comparison of Pooling Methods for Convolutional Neural Networks,” *Applied Sciences*, vol. 12, no. 17, Art. no. 17, Jan. 2022, doi: 10.3390/app12178643.

[205] Y.-L. Boureau, J. Ponce, and Y. LeCun, “A theoretical analysis of feature pooling in visual recognition,” in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, in ICML’10. Madison, WI, USA: Omnipress, Jun. 2010, pp. 111–118.

[206] R. Caruana, S. Lawrence, and C. Giles, “Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping,” *Advances in neural information processing systems*, vol. 13, 2000.

[207] D. Ekambaram and V. Kumar, “Identification of Defects in Casting Products by using a Convolutional Neural Network,” *IEIE Transactions on Smart Processing and Computing*, vol. 11, pp. 149–155, Jun. 2022, doi: 10.5573/IEIESPC.2022.11.3.149.

[208] T. Benbarrad, M. Salhaoui, S. B. Kenitar, and M. Arioua, “Intelligent Machine Vision Model for Defective Product Inspection Based on Machine Learning,” *Journal of Sensor and Actuator Networks*, vol. 10, no. 1, Art. no. 1, Mar. 2021, doi: 10.3390/jsan10010007.

[209] S. Taweelerd, C. C. Chang, and G. Y. Tzou, “Vision system based on deep learning for product inspection in casting manufacturing: pump impeller images,” *J. Phys.: Conf. Ser.*, vol. 2020, no. 1, p. 012046, Sep. 2021, doi: 10.1088/1742-6596/2020/1/012046.

[210] H. T. Nguyen, N. Shin, G.-H. Yu, G.-J. Kwon, W.-Y. Kwak, and J.-Y. Kim, “Deep learning-based defective product classification system for smart factory,” in *The 9th International Conference on Smart Media and Applications*, in SMA 2020. New York, NY, USA: Association for Computing Machinery, Nov. 2021, pp. 80–85. doi: 10.1145/3426020.3426039.

[211] U. K. Lilhore, S. Simaiya, J. K. Sandhu, N. K. Trivedi, A. Garg, and A. Moudgil, “Deep Learning-Based Predictive Model for Defect Detection and Classification in Industry 4.0,” in *2022 International Conference on Emerging Smart Computing and Informatics (ESCI)*, Mar. 2022, pp. 1–5. doi: 10.1109/ESCI53509.2022.9758280.

[212] M. Elion, S. Tabti, and J. Budynek, “INTERPRETABILITY OF DEEP LEARNING MODELS FOR VISUAL DEFECT DETECTION: A PRELIMINARY STUDY,” presented at the ConfianceAI Days, Paris, France, Oct. 05, 2022. [Online]. Available: https://sonia.wp.imt.fr/files/2022/08/Confiance_ai22_Fieldbox_final.pdf

[213] D. Tsiktsiris, T. Sanida, A. Sideris, and M. Dasygenis, “Accelerated Defective Product Inspection on the Edge Using Deep Learning,” in *Recent Advances in Manufacturing Modelling and Optimization*, S. Kumar, J. Ramkumar, and P. Kyratsis, Eds., in Lecture Notes in Mechanical Engineering. Singapore: Springer Nature, 2022, pp. 185–191. doi: 10.1007/978-981-16-9952-8_18.

[214] S. Sundaram and A. Zeid, “Smart Prognostics and Health Management (SPHM) in Smart Manufacturing: An Interoperable Framework,” *Sensors*, vol. 21, no. 18, Art. no. 18, Jan. 2021, doi: 10.3390/s21185994.

[215] M. H. Karray, B. Chebel-Morello, and N. Zerhouni, “A formal ontology for industrial maintenance,” *Applied ontology*, vol. 7, no. 3, pp. 269–310, 2012.

[216] M. H. Karray, F. Ameri, M. Hodkiewicz, and T. Louge, “ROMAIN: Towards a BFO compliant reference ontology for industrial maintenance,” *Applied Ontology*, vol. 14, no. 2, pp. 155–177, 2019, doi: 10.3233/AO-190208.

[217] J. J. Montero Jiménez, R. Vingerhoeds, B. Grabot, and S. Schwartz, “An ontology model for maintenance strategy selection and assessment,” *Journal of Intelligent Manufacturing*, vol. 34, no. 3, pp. 1369–1387, 2023, doi: https://doi.org/10.1007/s10845-021-01855-3.

[218] Federal Energy Management Program, “Operations and Maintenance Challenges and Solutions,” *Energy.gov*, 2021. https://www.energy.gov/femp/operations-and-maintenance-challenges-and-solutions (accessed Jun. 07, 2023).

[219] R. Masoni *et al.*, “Supporting remote maintenance in industry 4.0 through augmented reality,” *Procedia manufacturing*, vol. 11, pp. 1296–1302, 2017, doi: https://doi.org/10.1016/j.promfg.2017.07.257.

[220] D. Mourtzis, V. Zogopoulos, and E. Vlachou, “Augmented reality application to support remote maintenance as a service in the robotics industry,” *Procedia Cirp*, vol. 63, pp. 46–51, 2017, doi: https://doi.org/10.1016/j.procir.2017.03.154.

[221] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, “A theory of learning from different domains,” *Machine learning*, vol. 79, pp. 151–175, 2010, doi: https://doi.org/10.1007/s10994-009-5152-4.

[222] M. P. Brundage, T. Sexton, M. Hodkiewicz, A. Dima, and S. Lukens, “Technical language processing: Unlocking maintenance knowledge,” *Manufacturing Letters*, vol. 27, pp. 42–46, Jan. 2021, doi: 10.1016/j.mfglet.2020.11.001.

[223] A. Dima, S. Lukens, M. Hodkiewicz, T. Sexton, and M. P. Brundage, “Adapting natural language processing for technical text,” *Applied AI Letters*, vol. 2, no. 3, p. e33, 2021, doi: 10.1002/ail2.33.

[224] J. C. Linhares and A. Dias, “A linguistic approach proposal for mechanical design using natural language processing,” presented at the Computational Processing of the Portuguese Language: 6th International Workshop, PROPOR 2003 Faro, Portugal, June 26–27, 2003 Proceedings 6, Springer, 2003, pp. 171–174. doi: https://doi.org/10.1007/3-540-45011-4_25.

[225] C. Jiang, H. Chen, Q. Xu, and X. Wang, “Few-shot fault diagnosis of rotating machinery with two-branch prototypical networks,” *J Intell Manuf*, vol. 34, no. 4, pp. 1667–1681, Apr. 2023, doi: 10.1007/s10845-021-01904-x.

[226] R. Zhang, N. Zhao, L. Fu, X. Bai, and J. Cai, “Recognizing defects in stainless steel welds based on multi-domain feature expression and self-optimization,” *J Intell Manuf*, vol. 34, no. 3, pp. 1293–1309, Mar. 2023, doi: 10.1007/s10845-021-01849-1.

[227] G. E. Granados, L. Lacroix, and K. Medjaher, "Condition monitoring and prediction of solution quality during a copper electroplating process," *J Intell Manuf*, vol. 31, no. 2, pp. 285–300, Feb. 2020, doi: 10.1007/s10845-018-1445-4.
[228] M. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of English: The Penn Treebank," 1993, [Online]. Available: https://repository.upenn.edu/cis_reports/237/
[229] BNC Consortium, "British national corpus," *Oxford Text Archive Core Collection*, 2007, [Online]. Available: http://www.natcorp.ox.ac.uk/
[230] S. Merity, C. Xiong, J. Bradbury, and R. Socher, "Pointer sentinel mixture models," *arXiv preprint arXiv:1609.07843*, 2016, [Online]. Available: https://doi.org/10.48550/arXiv.1609.07843
[231] Amazon, "Amazon Customer Reviews Dataset," 2015. https://s3.amazonaws.com/amazon-reviews-pds/readme.html (accessed Jun. 08, 2023).
[232] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," presented at the Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, 2011, pp. 142–150. [Online]. Available: https://aclanthology.org/P11-1015.pdf
[233] T. Russell-Rose, M. Stevenson, and M. Whitehead, "The Reuters Corpus Volume 1-from Yesterday's News to Tomorrow's Language Resources.," 2002, [Online]. Available: https://research.gold.ac.uk/id/eprint/29760/1/80.pdf
[234] T. Copeck, K. Barker, S. Delisle, S. Szpakowicz, and J.-F. Delannoy, "What is technical text?," *Language Sciences*, vol. 19, no. 4, pp. 391–423, 1997, doi: https://doi.org/10.1016/S0388-0001(97)00003-X.
[235] F. Schauer, "Is law a technical language," *San Diego L. Rev.*, vol. 52, p. 501, 2015, [Online]. Available: https://digital.sandiego.edu/sdlr/vol52/iss3/2/
[236] L. L. Fuller, "Positivism and fidelity to law--A reply to Professor Hart," *Harv. L. Rev.*, vol. 71, p. 630, 1958, doi: https://doi.org/10.2307/1338226.
[237] O. W. Holmes, "The path of the law," *Harvard Law Review*, vol. 110, no. 5, pp. 991–1009, 1997, doi: https://doi.org/10.2307/1342108.
[238] B. Horvath, "Jurisprudence, Men and Ideas of the Law," *The American Journal of Comparative Law*, vol. 3, no. 3, pp. 448–451, Jul. 1954, doi: 10.2307/837969.
[239] T. M. Chung and P. Nation, "Identifying technical vocabulary," *System*, vol. 32, no. 2, pp. 251–263, Jun. 2004, doi: 10.1016/j.system.2003.11.008.
[240] F. Schironi, "Technical languages: Science and medicine," *A companion to the Ancient Greek language*, pp. 338–53, 2010, doi: https://doi.org/10.1002/9781444317398.ch23.
[241] V. P. Meshalkin, E. A. Panina, and R. S. Bykov, "Principles of developing an interactive system for the semantic processing of scientific and technical texts on chemical technology of reagents and ultrapure substances," *Theoretical Foundations of Chemical Engineering*, vol. 49, pp. 422–426, 2015, doi: https://doi.org/10.1134/S0040579515040314.
[242] P. A. Alexander and J. M. Kulikowich, "Learning from physics text: A synthesis of recent research," *Journal of research in science teaching*, vol. 31, no. 9, pp. 895–911, 1994, [Online]. Available: https://doi.org/10.1002/tea.3660310906

[243] M. Krallinger and A. Valencia, “Text-mining and information-retrieval services for molecular biology,” *Genome biology*, vol. 6, no. 7, pp. 1–8, 2005, doi: https://doi.org/10.1186/gb-2005-6-7-224.

[244] W. J. Wilbur and Y. Yang, “An analysis of statistical term strength and its use in the indexing and retrieval of molecular biology texts,” *Computers in Biology and Medicine*, vol. 26, no. 3, pp. 209–222, 1996, doi: https://doi.org/10.1016/0010-4825(95)00055-0.

[245] C. Stenström, M. Al-Jumaili, and A. Parida, “Natural language processing of maintenance records data,” *International journal of condition monitoring and diagnostic engineering management*, vol. 18, no. 2, pp. 33–37, 2015, Accessed: May 09, 2023. [Online]. Available: https://www.scopus.com/record/display.uri?eid=2-s2.0-84959309369&origin=inward&txGid=d8d4c40955e8b463ea2991b7ab882d32

[246] S. Sundaram and A. Zeid, “Artificial intelligence-based smart quality inspection for manufacturing,” *Micromachines*, vol. 14, no. 3, p. 570, 2023, doi: https://doi.org/10.3390/mi14030570.

[247] H. Bokinsky *et al.*, “Application of natural language processing techniques to marine V-22 maintenance data for populating a CBM-oriented database,” presented at the AHS Airworthiness, CBM, and HUMS Specialists’ Meeting, Huntsville, AL, 2013. [Online]. Available: https://www.proceedings.com/19340.html

[248] C. Woods, M. Hodkiewicz, and T. French, “Requirements for Adaptive User Interfaces for Industrial Maintenance Procedures: A discussion of context, requirements and research opportunities,” in *Proceedings of the 31st Australian Conference on Human-Computer-Interaction*, in OzCHI ’19. New York, NY, USA: Association for Computing Machinery, Jan. 2020, pp. 322–326. doi: 10.1145/3369457.3369487.

[249] C. D. O’Donoghue and J. G. Prendergast, “Implementation and benefits of introducing a computerised maintenance management system into a textile manufacturing company,” *Journal of Materials Processing Technology*, vol. 153–154, pp. 226–232, Nov. 2004, doi: 10.1016/j.jmatprotec.2004.04.022.

[250] D. Rajpathak and R. Chougule, “A generic ontology development framework for data integration and decision support in a distributed environment,” *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 2, pp. 154–170, 2011, doi: 10.1080/0951192X.2010.531291.

[251] M. Navinchandran, M. E. Sharp, M. P. Brundage, and T. B. Sexton, “Discovering critical KPI factors from natural language in maintenance work orders,” *Journal of Intelligent Manufacturing*, pp. 1–19, 2021, doi: https://doi.org/10.1007/s10845-021-01772-5.

[252] L. Zhou and G. Hripcsak, “Temporal reasoning with medical data—a review with emphasis on medical natural language processing,” *Journal of biomedical informatics*, vol. 40, no. 2, pp. 183–202, 2007, doi: https://doi.org/10.1016/j.jbi.2006.12.009.

[253] J. J. Webster and C. Kit, “Tokenization as the initial phase in NLP,” presented at the COLING 1992 volume 4: The 14th international conference on computational linguistics, 1992. [Online]. Available: https://aclanthology.org/C92-4173.pdf

[254] C. Manning and H. Schutze, *Foundations of statistical natural language processing*. MIT press, 1999. [Online]. Available: https://mitpress.mit.edu/9780262133609/

[255] B. Plank, "What to do about non-standard (or non-canonical) language in NLP," *arXiv preprint arXiv:1608.07836*, 2016, doi: https://doi.org/10.48550/arXiv.1608.07836.

[256] F. Ansari, R. Glawar, and T. Nemeth, "PriMa: a prescriptive maintenance model for cyber-physical production systems," *International Journal of Computer Integrated Manufacturing*, vol. 32, no. 4–5, pp. 482–503, May 2019, doi: 10.1080/0951192X.2019.1571236.

[257] F. Ansari, R. Glawar, and W. Sihn, "Prescriptive Maintenance of CPPS by Integrating Multimodal Data with Dynamic Bayesian Networks," in *Machine Learning for Cyber Physical Systems*, J. Beyerer, A. Maier, and O. Niggemann, Eds., in Technologien für die intelligente Automation. Berlin, Heidelberg: Springer, 2020, pp. 1–8. doi: 10.1007/978-3-662-59084-3_1.

[258] F. Akhbardeh, T. Desell, and M. Zampieri, "Maintnet: A collaborative open-source library for predictive maintenance language resources," *arXiv preprint arXiv:2005.12443*, 2020, [Online]. Available: https://doi.org/10.48550/arXiv.2005.12443

[259] F. Akhbardeh, T. Desell, and M. Zampieri, "NLP tools for predictive maintenance records in MaintNet," presented at the Proceedings of the 1st conference of the asia-pacific chapter of the association for computational linguistics and the 10th international joint conference on natural language processing: System demonstrations, 2020, pp. 26–32. [Online]. Available: https://aclanthology.org/2020.aacl-demo.5/

[260] S. Sarica and J. Luo, "Stopwords in technical language processing," *Plos one*, vol. 16, no. 8, p. e0254937, 2021, doi: https://doi.org/10.1371/journal.pone.0254937.

[261] K. A. Latorella and P. V. Prabhu, "A review of human error in aviation maintenance and inspection," *International Journal of industrial ergonomics*, vol. 26, no. 2, pp. 133–161, 2000, doi: https://doi.org/10.1016/S0169-8141(99)00063-3.

[262] W. Wan, Y. Liu, X. Han, and H. Wang, "Evaluation Model of Power Operation and Maintenance Based on Text Emotion Analysis," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–8, 2021, doi: https://doi.org/10.1155/2021/2824689.

[263] M. C. Phan, A. Sun, and Y. Tay, "Robust representation learning of biomedical names," presented at the Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 3275–3285. doi: 10.18653/v1/P19-1317.

[264] P. Warnekar and J. Carter, "HIV terms coverage by a commercial nomenclature," presented at the AMIA Annual Symposium Proceedings, American Medical Informatics Association, 2003, p. 1046. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1480038/

[265] A. W. Qurashi, V. Holmes, and A. P. Johnson, "Document processing: Methods for semantic text similarity analysis," presented at the 2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), IEEE, 2020, pp. 1–6. doi: 10.1109/INISTA49547.2020.9194665.

[266] H. Tekgöz, S. İ. Omurca, K. Y. Koç, U. Topçu, and O. Çelik, "Semantic Similarity Comparison Between Production Line Failures for Predictive Maintenance,"

Advances in Artificial Intelligence Research, vol. 3, no. 1, pp. 1–11, 2023, doi: https://doi.org/10.54569/aair.1142568.
[267] S. M. R. Naqvi, M. Ghufran, S. Meraghni, C. Varnier, J.-M. Nicod, and N. Zerhouni, "Cbr-based decision support system for maintenance text using nlp for an aviation case study," presented at the 2022 Prognostics and Health Management Conference (PHM-2022 London), IEEE, 2022, pp. 344–349. doi: 10.1109/PHM2022-London52454.2022.00067.
[268] K. Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of documentation*, vol. 28, no. 1, pp. 11–21, 1972, doi: https://doi.org/10.1108/eb026526.
[269] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011, [Online]. Available: https://zbmath.org/1280.68189
[270] D. Chandrasekaran and V. Mago, "Evolution of semantic similarity—a survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–37, 2021, doi: https://doi.org/10.1145/3440755.
[271] W. H. Gomaa and A. A. Fahmy, "A survey of text similarity approaches," *international journal of Computer Applications*, vol. 68, no. 13, pp. 13–18, 2013, doi: 10.5120/11638-7118.
[272] A. Vaswani *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017, [Online]. Available: https://www.proceedings.com/39083.html
[273] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018, doi: https://doi.org/10.48550/arXiv.1810.04805.
[274] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019. doi: 10.18653/v1/D19-1410.
[275] T. Wolf *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," *arXiv preprint arXiv:1910.03771*, 2019, doi: https://doi.org/10.48550/arXiv.1910.03771.
[276] FAA Flight Standards Service, "FEDERAL AVIATION ADMINISTRATION JOINT AIRCRAFT SYSTEM/COMPONENT CODE TABLE AND DEFINITIONS." 2008. [Online]. Available: https://av-info.faa.gov/sdrx/documents/JASC_Code.pdf
[277] U.S. Department of Transportation and Federal Aviation Administration, "Survey of Service Difficulty Reports for In-Service Damage in Transport Category Aircraft," Tech Report DOT/FAA/TC-TN17/70, 2017. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/57873
[278] R. L. Rose, T. G. Puranik, D. N. Mavris, and A. H. Rao, "Application of structural topic modeling to aviation safety data," *Reliability Engineering & System Safety*, vol. 224, p. 108522, 2022, doi: https://doi.org/10.1016/j.ress.2022.108522.
[279] X. Sun, B. Li, Y. Li, and Y. Chen, "What information in software historical repositories do we need to support software maintenance tasks? an approach based on

topic model," *Computer and Information Science*, pp. 27–37, 2015, doi: https://doi.org/10.1007/978-3-319-10509-3_3.
[280] S. Wu, "Short text mining for fault diagnosis of railway system based on multi-granularity topic model," presented at the 2018 8th International Conference on Logistics, Informatics and Service Sciences (LISS), IEEE, 2018, pp. 1–6. doi: 10.1109/LISS.2018.8593228.
[281] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *J Mach Learn Res*, vol. 3, no. Jan, pp. 993–1022, 2003, [Online]. Available: https://zbmath.org/1112.68379
[282] H. Jelodar *et al.*, "Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey," *Multimedia Tools and Applications*, vol. 78, pp. 15169–15211, 2019, doi: https://doi.org/10.1007/s11042-018-6894-4.
[283] M. Taddy, "On estimation and selection for topic models," presented at the Artificial intelligence and statistics, PMLR, 2012, pp. 1184–1193. [Online]. Available: https://proceedings.mlr.press/v22/taddy12.html
[284] C. Sievert and K. Shirley, "LDAvis: A method for visualizing and interpreting topics," presented at the Proceedings of the workshop on interactive language learning, visualization, and interfaces, 2014, pp. 63–70. [Online]. Available: https://aclanthology.org/W14-3110.pdf
[285] B. Fuglede and F. Topsoe, "Jensen-Shannon divergence and Hilbert space embedding," presented at the International symposium onInformation theory, 2004. ISIT 2004. Proceedings., IEEE, 2004, p. 31. doi: 10.1109/ISIT.2004.1365067.
[286] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1–3, pp. 37–52, 1987, doi: https://doi.org/10.1016/0169-7439(87)80084-9.
[287] International Air Transport Association, "Aviation Data - Data Science Hype or Ripe for Aviation," White Paper, 2019.