Application and optimization of artificial intelligence techniques for small to medium sized manufacturing enterprises in an industry 4.0 environment

by

KAHIOMBA SONIA KIANGALA

submitted in accordance with the requirements

for the degree of

DOCTOR OF PHILOSOPHY

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: PROF Z WANG

JANUARY 2022

DECLARATION

Name: Kahiomba Sonia KIANGALA

Student number: 60988568

Degree: Doctor of Philosophy in Science, Engineering, and Technology (SET)

Thesis title: Application and optimization of artificial intelligence techniques for small to medium sized manufacturing enterprises in an Industry 4.0 environment

I declare that the above thesis is the outcome of my own work, research, and investigation. All sources that have contributed in this research or that I have utilized and quoted have been clearly indicated and acknowledged by using complete references.
I also declare that I have not previously submitted this work, or part of it, for examination at UNISA for another qualification or at any other higher education institution.

11.11.2021

SIGNATURE **DATE**

ABSTRACT

The ascent of the current industrial revolution known as Industry 4.0 (I40) or smart manufacturing is considerably impacting the manufacturing sector with the utilization of more intelligent and advanced practices such as Artificial Intelligence (AI) methods and Machine Learning (ML) within factories. The successful implementation of intelligent practices promises to significantly increase organization flexibility, improve operations, and increase production throughput. Without a doubt, to remain relevant in the competitive market and ensure their future growth, manufacturing companies should start actively considering and adopting advanced and intelligent I40 solutions for their businesses.

Manufacturing small and medium-scaled enterprises (SMEs) represent the backbone of several countries' economies, actively supporting jobs creations and contributing to Gross Domestic Product (GDP). However, SMEs have been the least appealing in embracing new and advanced technological trends within their production processes, mainly due to a lack of appropriate practical guidance or solutions customized to their environment and limited resources (financial and human) in their organizations. Few of the existing works intending to encourage manufacturing SMEs in adopting new technological trends are high-level frameworks, surveys, or only tackle the design of advanced technological solutions without incorporating the corresponding organizational amendments SMEs need to undergo for a sustainable implementation of advanced and innovative solutions.

Inspired by some of the technical and organizational challenges of manufacturing SMEs, we utilize the Design Science Research Methodology (DSRM) to create, illustrate, and assess several AI and innovative techniques that increase manufacturing SMEs' performance. The AI concepts and advanced technological methods we design in this study improve several existing solutions developed by our predecessors in the research field.

Our major research contributions are the creation of a practical guide for the application of various AI techniques and innovative solutions adapted and optimized for manufacturing SMEs such as intelligent predictive maintenance (PM) method, automatic parameter configuration for Supervisory Control And Data Acquisition (SCADA) system, product customization framework, robust communication network prototype, and enhanced safety response mechanism. We implement various advanced technologies like ML, Time-Sensitive

Networking (TSN), speech recognition, and more to empower our innovative solutions. Our study is valuable to the SME research field (in academia). It provides new literature that combines the design of advanced and optimized technological trends with organization structural (business model) changes to promote innovation within SMEs, prompting more research in that regard. Since we are motivated by some of the manufacturing SMEs' challenges, our research is also useful to the industry as it provides practical solutions that manufacturing SMEs can adapt and implement to boost their production processes performances.

Keywords: advanced manufacturing solutions; artificial Intelligence (AI); industry 4.0 (I40); innovative business model; machine learning (ML) algorithms; predictive maintenance (PM); product customization; safety response mechanism; small and medium-scaled enterprises (SMEs); robust network topology.

ACKNOWLEDGMENT

Conducting doctoral research is a demanding and tremendous journey requiring high discipline, hard work, abnegation, and support. It is easy to abandon this journey, lose hope of completion, and even experience depression because of all the activities one must put aside to maintain a productive study rhythm (family, friends, and outings). I am incredibly blessed to have so many people who supported me toward completing my doctoral degree. I will probably not be able to list them all, but I would like to express my gratitude toward them all from the bottom of my heart.

- To my God, Lord and Saviour Jesus-Christ for granting me by His Grace all the necessary resources to complete my research, for giving me the grace of life, good health, and for uplifting me during my worst times. I am incredibly grateful.

- To my wonderful husband, Minerve Mampaka, for being the initiator of this doctoral degree, for your patience, and for your continuous encouragement through my ups and downs. You probably deserve this degree more than I do. I am thankful for you.

- To my supervisor, Professor Zenghui Wang, for the significant inputs, support, and guidance from the very beginning until the completion of this research. Your promptness and willingness to always assist have been a great relief. Thank you very much.

- To my parents, Emmanuel and Alphonsine Kiangala, for instilling in me the desire to pursue more important goals and believing in my ability to accomplish them. I am grateful to you.

- To my siblings, Clovis, Elvis, Jules, Maxime, Sheila, and Quentin Kiangala, for giving me reasons to continue the fight to, one day, by God's grace, improve our conditions.

- To my amazing sisters Gloria Mampaka and Sheila Kiangala for being my housemates, cheerleaders, cook, and companion during this journey. You always did your best to make sure I did not miss anything at home. I am grateful for you girls.

- To my incredible support system, the Kimbodja family, Papa Christian, Maman Juliette, Solange and Patrick Mutombo, Karl Kabwe, Christelle Kala-Konga, Kevin Bibombe, Niclette Milambu, Sheila Kiangala, Gloria Mampaka, Ariel Tshimbalanga and, Nathan Kabwe. Thanks for being my family and strong support in times of joy and pain. Thanks for cheering me up, thanks for your prayers, and for understanding my long self-isolation during this time.

- To my spiritual fathers Simon Mbala and Jules Kiangala, s.j, I am grateful for your continuous prayers, support, and faith in my work. You have always been prompt to check up on me and encourage me through this study. Thank you very much.

- To my friend and colleague, Daniel Nyange, thank you for being a great work and learning partner. Indeed, we made a good team, and this allowed me, in so many ways, to manage a full-time job and to carry out my doctoral research. I am genuinely thankful.

To all of you listed above and to those I might have omitted, May God, our Almighty Lord Jesus-Christ, bless you beyond expectations, and may He grant the deep desires of your hearts for the glory of His Mighty Name.

DEDICATION

I dedicate this work to my parents, Emmanuel and Alphonsine Kiangala. Thank you for all your efforts and sacrifices.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

4IR	Fourth Industrial Revolution
AI	Artificial Intelligence
AMR	Autonomous Moving Robot
ANN	Artificial Neural Network
API	Application Programming Interface
AR	Augmented-Reality
CBM	Condition-Based Monitoring
CNC	Computer Numerical Control
CNN	Convolutional Neural Network
CPS	Cyber-Physical Systems
CPU	Central Processing Unit
DAN H	Double Attached Node Supporting HSR
DAN P	Double Attached Node Supporting PRP
DMLC	Distributed Machine Learning Community
DSRM	Design Science Research Methodology
DT	Decision Tree
ESTOP	Emergency STOP
GAF	Gramian Angular Field
GADF	Gramian angular differential field
GASF	Gramian angular summation field
GBDT	Gradient Boosting Decision Tree
GDP	Gross Domestic Product

GPS	Global Positioning System
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HMI	Human Machine Interface
HMM	Hidden Markov Model
HR	Human Resources
HSRP	High-availability Seamless Redundancy Protocol
I40	Industry 4.0
ICT	Information and Communications Technology
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IIoT	Industrial Internet Of Things
IM	Intelligent Manufacturing
IoT	Internet Of Things
IP	Internet Protocol
IT	Information Technology
KNN	K-Nearest Neighbour
KPI	Key Performance Indicator
LAN	Local Area Network
LR	Linear Regression
MAC	Media Access Control
MC	Mass Customization
MCC	Motor Control Centre
ML	Machine Learning

MLR	Multiple Linear Regression
MRP	Media Redundancy Protocol
MSE	Mean Square Error
MTD	Machine-Type Device
MTS	Multivariate Time Series
NHMI	Natural Human Machine Interface
NIC	Network Interface Card
NMS	Network Management Software
NN	Neural Network
OEE	Overall Equipment Effectiveness
OSI	Open Systems Interconnection
OT	Operational Technology
PC	Personal Computer
PCA	Principal Component Analysis
PCMS	Personalized Customization Manufacturing Systems
PLC	Programmable Logic Controller
PM	Predictive Maintenance
PReLU	Parameterized Rectifier Linear Unit
PRP	Parallel Redundancy Protocol
QL	Q-learning
R&D	Research and Development
RCM	Remote Condition Monitoring
RedBox	Redundancy Box
ReLU	Rectifier Linear Unit

RF	Random Forest
RFID	Radio Frequency Identification
RL	Reinforcement Learning
RLS	Regularized Least Squared
ROI	Return On Investment
RSS	Residual Sum of Square
RSTP	Rapid Spanning Tree Protocol
SCADA	Supervisory Control And Data Acquisition
SMEs	Small and Medium-scaled Enterprises
SR	Symbolic Regression
STP	Spanning Tree Protocol
SVM	Support Vector Machine
TSN	Time-Sensitive Networking
UDP	User Datagram Protocol
UTS	Univariate Time Series
VSD	Variable Speed Drive
XGBOOST	Extreme Gradient Boosting

LIST OF PUBLICATIONS FROM THE THESIS

1. Kiangala, K.S. & Wang, Z. (2020 a) An Effective Predictive Maintenance Framework for Conveyor Motors Using Dual Time-Series Imaging and Convolutional Neural Network in an Industry 4.0 Environment. *IEEE Access*, 8, pp. 121033-121049. https://doi.org/10.1109/ACCESS.2020.3006788 **(This work is part of Chapters 4, 5, and 6). ISI master indexed journal**

2. Kiangala, K.S. & Wang, Z. (2020 b) An adaptive framework for configuration of parameters in an Industry 4.0 manufacturing SCADA system by merging machine learning techniques. *Int. Conf. Artif. Intel., Big Data, Comput. Data Comm. Syst. (icABCD).* Publisher: IEEE, https://doi.org/10.1109/icABCD49160.2020.9183818 **(This work is part of Chapters 4, 5, and 6).**

3. Kiangala, K.S. & Wang, Z. (2021 a) An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment. *Machine Learning with Applications*. Publisher: Science Direct, 4. https://doi.org/10.1016/j.mlwa.2021.100024 **(This work is part of Chapters 4, 5, and 6).**

4. Kiangala, K.S. & Wang, Z. (2021 b) An Effective Communication Prototype for Time-Critical IIoT Manufacturing Factories Using Zero-Loss Redundancy Protocols, Time-Sensitive Networking, and Edge-Computing in an Industry 4.0 Environment. *Processes*. Publisher: MDPI, *9*, 2084. https://doi.org/10.3390/pr9112084 (**This work is part of Chapters 4, 5, and 6). ISI master indexed journal**

5. Kiangala, K.S. & Wang, Z. (2021 c) A Safety Response Mechanism for an Autonomous Moving Robot in a Small Manufacturing Environment using Q-learning Algorithm and Speech Recognition. *Sensors* Journal. Publisher: MDPI. **Accepted – Acceptance certificate attached (This work is part of Chapters 4, 5, and 6).**

Chapter 1 : INTRODUCTION

1.1. Thesis research background

Industry 4.0 (I40), popularly known as smart manufacturing, designates the current revolution in the industrial manufacturing sector, also referred to as the operational technology (OT) world, which aims to convert traditional manufacturing production processes into more intelligent, autonomous, and advanced systems. When correctly implemented, advanced production processes in I40 environments should considerably improve organizations' performance and relevance in a growing competitive market. Hence, I40 constitutes a promising concept to exploit with huge potential to revolutionize the whole OT sector positively (Kiangala & Wang 2020 b).

The application of I40 within manufacturing factories promotes the intensive utilization of several innovative technologies such as big-data (Chen *et al.* 2014), cyber-physical systems (CPS) (Rauch *et al*. 2016), cloud computing (Chen *et al.* 2015), embedded technology (Wan *et al.* 2010), augmented reality (AR) and simulation (Romeo *et al.* 2018), mobile computing (Zhang, Y. 2017 and Zhang, D. 2012), industrial wireless networks (Li *et al.,* 2015), Internet of things (IoT) (Falkenreck & Wagner, 2017), and artificial intelligence (AI) (Marki, B. *et al.* 2016). Data or information collection is the first enabler and the fuel required to produce effective solutions for most of these innovative techniques. Gokalp *et al.* (2016) states that businesses can optimize their production processes and improve their decision-making by significantly analyzing production data information. Therefore, data analytics, manipulation, and collection is a crucial asset in I40 production systems.

In the information technology (IT) environment, researchers have favorably achieved data collection and manipulation for several centuries. The advent of the AI notion, which gives machines the ability to learn, think, and make decisions like human beings, has multiplied the creation of powerful IT data-based applications that presently impact our daily lives. Few of these applications are prediction functionalities in search engines (product suggestion, product recommendation based on previous searches, Google web pages ranking, and much more), email automatic spam filtering functions, speech and image processing tasks in software applications (Facebook pictures facial recognition, smartphones locking and unlocking capabilities, and much more) (Das *et al.,* 2015) (Nayak & Dutta 2017), drones, and autonomous

cars. However, the OT environment faced several challenges regarding effective data collection, manipulation, and analysis due to the data and protocols uniformity coming from different vendors with incompatible formats (proprietary solutions) and possible system failures caused by data manipulation. Unlike in the IT environment where system security (data confidentiality) is crucial, system availability is a non-negotiable component in the OT world. Most system breakdowns in industrial production processes are unacceptable as they can result in extreme losses for organizations.

The integration of AI techniques in manufacturing production processes led to a concept known as intelligent manufacturing (IM) (Yao *et al.* 2017). In 1998, Wright & Bourne (1988) presented the first IM approach. Since then, several studies have designed and developed advanced IM in system diagnosis, system design, control, and inspection (Teti & Kumara 1997). A more recent IM trend, boomed by I40 promises, lies in applying machine learning (ML), a popular AI branch, and other advanced I40 technologies to develop efficacious tools and systems for manufacturing systems. The predictive maintenance (PM) method is one of these tools that allow factories to detect and predict system threats before they occur. Various researchers applied ML techniques to create a more powerful PM scheme. Coraddu et al. (2014) utilized some ML algorithms to design an improved PM system for gas turbines. Leahy et al. (2016) implement ML techniques to create wind turbine predictive maintenance and fault detection systems. Susto et al. (2015) developed an ML-PM-based system to support semiconductors manufacturing, and Paolanti et al. (2018) focused on implementing ML for early fault detection (PM) in manufacturing cutting machines. Apart from its utilization in PM, ML has been implemented as a design supporting tool, particularly for accurate parameter selection, in various manufacturing systems. Jirdehi & Rezaei (2016) created a parameter estimation system for induction machinery using ML. Tessarolo , De Martin, Giulivo et al. (2014) and Tessarolo, De Martin, Diffen et al. (2014) designed a parameter prediction system, using ML algorithms, to select the appropriate machines (with accurate ratings) for factories. Haque (2008) produced a supporting tool for factories (built with ML techniques) to choose the most appropriate electrical circuit parameters by inputting their initial requirements. In advanced manufacturing sectors, the concept of product and mass customization (MC) has also been improved by incorporating various I40 technologies such as cloud computing, IoT, and industrial wireless networks, which produce more flexible customization platforms for manufacturing systems.

Jovanovski et al. (2019) and Kiangala & Wang (2018), reported that although manufacturing small and medium-sized enterprises (SMEs) are active contributors to several countries' economic growth worldwide, especially with regards to jobs creations, they are the least appealing in implementing advanced innovative techniques such as I40 and AI techniques within their production processes. It is mainly due to the required financial, material, and human resources, especially in research and development (R&D), for the adoption of new technologies that SMEs cannot usually afford. However, by ignoring the vast opportunities, new technological trends offer, SMEs set themselves up for limited growth, a high risk of failure, and future irrelevance in a competitive market. Instead, they should invest in finding different ways and solutions to consider innovative technological trends (Matt *et al.*, 2018). As per Ganzarain & Errasti (2016), various manufacturing SMEs interviewed about their lethargy in adopting I40 within their organizations suggested creating more practical solutions adapted to their environment to guide them through the implementation journey effectively.

Various existing studies conducted for the embracement of I40 and innovative techniques within manufacturing SMEs such as Truve et al. 2019; Jovanovski et al. 2019; Dossou 2019, Ganta 2020, Rauch et al. 2020, Safar et al. 2020 and Prause 2019, are either surveys, some do not provide clear, practical guidance for the implementation of innovative techniques. They suggest high-level frameworks and approaches within limited SMEs or others offer practical, innovative solutions and ignore guidance on organization modification needed in factories to ensure a sustainable application of these innovative techniques. In order to secure a global economic advancement, further practical researches and developments in this regard are imperative to support the backbone of our countries' economies, manufacturing SMEs, in growing their organizations and production processes through the embracement of new technological trends.

In the upcoming chapter sections, we explain the importance of designing more practical SME-customized AI, I40, and innovative solutions while guiding an adapted organizational structure (business model change) that promotes sustainable innovation.

1.2. Research problem statement

Most manufacturing SMEs face tremendous challenges with regards to the adoption of innovative technological trends such as AI and I40 methods, principally due to the lack of practical solutions adapted to their environment and their limited capital to invest in new technologies as opposed to large manufacturing organizations. Nevertheless, new and innovative technologies can increase organization performances, and SMEs who constitute the pillar of the global economy should consider embracing innovative technological methods for their production processes to remain apt in an increasingly competitive market.

Several existing studies conducted on implementing AI, I40, and other innovative technologies in manufacturing SMEs are mainly high-level frameworks, advice for readiness assessment, design strategies, and surveys. The few existing ones that offer practical methods for these innovations with explicit application guidance do not tackle the remodeling aspect of SMEs' organizations (business model amendment) which is crucial to create an innovative culture and guarantee the sustainable implementation of innovative techniques within businesses.

Our research addresses the above problems by developing several practical AI and advanced technological methods to guide manufacturing SMEs in I40 environments to adopt innovative techniques for the organizations. We design optimized solutions such as predictive maintenance, automatic parameter configuration system, robust communication network prototype, product customization framework, and safety response mechanism, improved from existing techniques, built based on some challenges manufacturing SMEs encounter in their environment, and mapped in a conceptual, innovative business model. The proposed conceptual business model also displays the organizational changes required to promote an innovative culture within businesses and gives indications on several other innovation areas that could further improve manufacturing SMEs' performances and operations.

1.3. Thesis research questions

In light of problems manufacturing SMEs are facing for the embracement of innovative methods and with regards to solutions we intend to bring through this research, we formulate the following research questions:

- How can manufacturing SMEs develop an innovative culture in their organizations to ease adopting advanced and innovative technological concepts?

- What are some of the current promising innovative methods and tools manufacturing factories could consider implementing to increase their performance in an I40 environment?

- How to optimize and customize some of the existing innovative manufacturing tools or processes using AI to produce better results in an SME environment context?

1.4. Thesis research aim

Our research aims to develop and implement practical, innovative tools and processes that improve the overall performance of SMEs' manufacturing production processes while suggesting a small conceptual business model that supports the sustainable application of these tools and processes in an I40 environment. We integrate AI methods and advanced technological concepts to optimize innovative tools and processes and achieve outstanding outcomes.

1.5. Thesis research objectives

Based on our research guiding questions raised in the previous section, we can summarize our research objectives as follows:

- **To build a simple conceptual business model that encourages innovation within manufacturing SMEs.**

  We design a simple conceptual business model that promotes the embracement of innovative techniques within manufacturing SMEs in an I40 scenario. The conceptual business model provides guiding actions to create an innovative culture and displays major innovation stakeholders within the SME organization.

- **To select manufacturing tools and processes to customize for utilization in SMEs I40 environment.**

  We research some of the existing manufacturing processes and tools intended to increase production processes performances under I40. These tools and processes are data collection and analysis, AR, PM, the use of intelligent robots as additional production workforce, robust network communication infrastructure, decentralized control and remote monitoring, digitalization of human resources (HR) and employee management system, digitalization of business finances linked to production processes, automation of robotic and repetitive operators tasks, product customization, and enhanced safety mechanisms. We select some to cover in-depth through this research and review them: data collection and analysis, PM, robust network communication infrastructure, automation of robotic and repetitive operators' tasks, product customization, and enhanced safety response mechanisms.

- **To improve and optimize selected manufacturing tools and processes using AI techniques and advanced technological methods.**

  We implement AI techniques and advanced technological methods to improve and optimize SME environments' selected manufacturing processes and tools.

  - **Predictive maintenance**: we create an efficient PM framework by using convolutional neural network (CNN), an ML algorithm (ML is an AI branch), and time-series imaging.

  - **Robust network communication infrastructure:** we design a network communication prototype by implementing some advanced networking technological concepts: time-sensitive networking (TSN), edge computing, and network redundancy.

  - **Automation of robotic and repetitive operators' tasks:** we transform a traditional SCADA system into an intelligent automatic parameter configuration system by integrating ML algorithms.

- **Product customization**: we create a product customization framework for customers to personalize their product requirements before manufacturing starts. We incorporated some ML algorithms to design the production customization backend.

- **Enhanced safety response mechanisms**: we develop an enhanced safety response mechanism for an autonomous moving robot (AMR) and human operators in a factory by utilizing an ML algorithm and speech recognition.

1.6. Thesis research originality / contribution

The sustainable implementation of optimized or advanced technological methods such as I40 concepts or AI solutions in companies depends highly on their organizational structure that should encourage innovation (Jovanovski *et al.* 2019). The rareness of customized technology solutions for SMEs discourages them from considering advanced technological trends (Kiangala & Wang 2018), and applying innovative technological concepts in a traditional organizational structure will most likely fail. Our research originality lies in a combination of organizational and technical guidance for manufacturing SMEs. Most researches either focus on only covering organizational changes to facilitate the adoption of innovative concepts or solely on developing advanced technical solutions for companies. Our research merges improvement on these two aspects by suggesting a conceptual business model that promotes innovation. We develop optimized technological solutions for manufacturing SMEs and map them in the proposed business model. Our conceptual business model also lists several other innovative areas (not covered in this research) that could be considered in future research to improve manufacturing SMEs' operations. Therefore, our study offers a more holistic solution (technical and organizational) and better guidelines to boost the embracement of advanced technological methods within SMEs.

1.7. Significance of the research

The significance of research refers to its impact in addressing or solving more significant problems, improving methods, and contributing to the enrichment of the research field (Kothari 2004 and Creswell 2013). Our research finds its relevance or significance in the three following sectors:

- **In society**: SMEs are the backbone of several countries' economies, contributing to increasing their gross domestic product (GDP) and creating jobs for their citizens **(**The World Bank 2021 and Matt & Rauch 2020). Providing them with various possible means to increase their performances is an excellent contribution to the whole society. Our study proposes several solutions to improve manufacturing SMEs' operations. Moreover, some of our proposed innovative solutions originate from current SMEs' challenges; therefore, we bring relief to existing issues.

- **In research**: As advised by few researchers (Rauch et al. 2020 and Kiangala & Wang 2018), the lack of practical, innovative solutions adapted to SMEs' environment considerably slows the adoption of new and advanced technologies within SMEs. There is a crucial need for more researches, studies, and guidance in this regard. Our thesis is a significant addition to the few existing studies that promote the embracement of advanced technological concepts within SMEs.

- **In theory:** Our study offers a clear understanding of the application of some theoretical AI and ML algorithms by bringing forth details on their utilization to create practical applications and solving problems. Hence, our research links the gap between theoretical knowledge and practical implementation.

1.8. Thesis research delimitations

Our research only covers five major domains improved to increase the performance of manufacturing SMEs:

- early detection of conveyor system threats (predictive maintenance)

- an automation of repetitive tasks (automatic parameter configuration for a SCADA system)

- creation of a robust communication network prototype

- customization of product manufacturing based on personalized clients' requirements (product customization)

- a safety response mechanism

We apply and optimize AI techniques in four of the above segments: predictive maintenance, automatic parameter configuration for a SCADA system, product customization, and enhanced safety response mechanism. The AI branch we exploit the most throughout this study is ML, with the implementation of several of its algorithms to achieve outstanding results: multiple linear regression (MLR), decision tree (DT), random forest (RF), extreme gradient boosting (XGBoost), Q-learning, and CNN.

We evaluate the reliability of our results essentially via software simulation. The only hardware testing utilized is in the enhanced safety mechanism segment for the speech recognition part using a personal computer (PC) microphone for emergency commands and a Siemens S7-1200 programmable logic controller (PLC). Apart from that, we do not build any additional hardware equipment.

1.9. Research limitations

Since our study focuses on SMEs operations, we apply a relatively small amount of data to train our ML models. We choose simple ML algorithms for the small size of data. The accuracy of our results is influenced by the type and quantity of data loaded. Companies with larger datasets could require different ML algorithms, different kinds of settings, and experience non-identical results. SMEs often have limited financial resources for investing in new technologies; we utilized mainly open source and free software, which are not necessarily the most effective in terms of functionalities. We develop several programming steps to achieve an action that could be completed in a single step using paid software. Depending on the

processor's capacity and the program's size, several programming steps could result in a delayed response for the developed method.

Because we test our solutions via simulations, we cannot guarantee a hundred percent results replication on real hardware devices. Several adjustments would be needed when applying our proposed methods to actual manufacturing equipment.

1.10. Research methodology

A research methodology aims to provide the appropriate steps, tools, and activities for undertaking research. It is often presented in an architectural structure and also gives necessary means to evaluate the research goals (Wahyuni 2012).

In this study, we utilize the Design Science Research Methodology (DSRM) to create, demonstrate, and assess optimized AI and innovative methods for I40 manufacturing SMEs. The DSRM gives researchers the ability to identify problems in their research field, set objectives that will help resolve identified problems, innovative design artefacts, and evaluate them (Peffers *et al.* 2007; Vaishnavi *et al.* 2017 and Hevner *et al.* 2004). DSRM is intensively applied for research in academia and the industry with projects related to software applications, software design, methods evaluations, and solution development.

Some studies suggested few guidelines for researching with the DSRM. Hevner et al. (2004) proposed a DSRM that emphasizes the choice of a specific audience for which researchers develop the innovative artefact, on the impact of the innovative artefact to solve an identified problem, on a rigorous design and evaluation of the innovative artefact, on the improvement of current field research knowledge information to develop new knowledge, and on proposing the developed knowledge and innovative artefact to the targeted audience. Peffer et al. (2007) recommended a systematic DSRM with detailed steps to design and evaluate the proposed innovative artefact. They also suggested guidelines for research reviewers to judge the outcome of DSRM activities. The proposed DSRM steps adapted to our thesis can be summarized as follows:

- **Problem identification**: The problem identification is the first step of the research methodology. It can be done via several literature reviews, careful observations of facts

and actions, or exchanges and discussions with other people. This step determines whether the research is worth being conducted or not.

For our research, the author's background in the industry providing technical and system integration solutions to various manufacturing SMEs triggered significant interest in this domain. The author witnessed manufacturing SMEs' challenges in adopting advanced technological trends in the working environment and confirmed this struggle through additional literature reviews. The literature review also supported the need for more practical studies regarding the embracement of new and advanced technological solutions for manufacturing. The significant role of SMEs in growing the global economy justifies the interest in such a topic.

- **Solution objectives definition:** In this second step, the author defined research objectives based on the problem(s) identified during the first step. Reaching the set solutions objectives should solve the research problems. The same solutions objectives are utilized to assess the quality of the innovative artefact designed at the end of the research.
  We present detailed aim and solution objectives for our research in sections 1.4 and 1.5. In short, our research intends to provide practical guidance to promote the use of advanced technological trends within manufacturing SMEs and to contribute to the SMEs' research academic field with additional supporting documents and literature resources.

- **Proposed artefact design and development:** After setting up the research objectives, the design and development step can start where all helpful knowledge, tasks, rules, and activities to create the artefact are detailed. This stage is a bit more theoretical than practical and varies based on the kind of artefact proposed. The creation of the actual innovative artefact happens at this step. Each chapter (from 5 to 9) describing our proposed optimized solutions has this section.

- **Designed artefact utility and practicality demonstration:** The designed artefact practicality demonstration stage proves that the proposed solution is appropriate to solve the identified problems.

In this research, we get inspired by some manufacturing SMEs' challenges to develop our innovative artefact to respond to them. Each chapter (from 5 to 9) describing our proposed optimized solutions has this section.

- **Evaluation:** The evaluation stage is about measuring and discussing the benefits or relevance of the designed artefact regarding results obtained in the previous practicality demonstration step. This stage also determines whether the outcome should be further improved before actual use or not. Each chapter (from 5 to 9) describing our proposed optimized solutions has this section.

- **Sharing the DSRM outcome:** The DSRM results sharing stage is the last step of the research methodology which consists of communicating the research outcome and the whole design through the suitable publication channels to reach the desired audience. The publication channels are books, academic and professional conferences, journals, magazines, and webinars.

  Our research proposed designs and solutions are shared with several journals/conferences as displayed in the publications section.

We illustrate in Figure 1-1 a graphical representation of our DSRM process flow based on the above descriptions. We have adapted our DSRM process flow from Peffers et al. (2007).

1.11. Ethical considerations

Our study on the application and optimization of AI techniques to improve manufacturing SMEs' performance was assessed and classified as research with negligible risk by the UNISA SOE review committee. In general, our research does not require immediate interaction with human beings or actions that could directly compromise humans. We mainly deal with machine simulations, software, and platform that are harmless to the environment. We evaluate our results and the solutions' deployment through software with extremely low risks of danger to individuals. We present the official ethical clearance approval by the UNISA SOE review committee in Appendices 2.

1.12. Chapter summary and research outline

In this chapter (introduction), we highlighted the background of this research. We described the problems we intend to resolve through this study. We stated the research questions and the associated objectives to answer these questions. We presented the overall research aim stating the ultimate goal to achieve in this study. We demonstrated the originality and the contribution of the research. We highlighted the significance of the research in society, in the research field, and theory. We presented delimitations and limitations of the study, and the methodology followed to achieve our research objectives. In this last section, we bring forth the overall thesis structure with small highlights of each chapter.

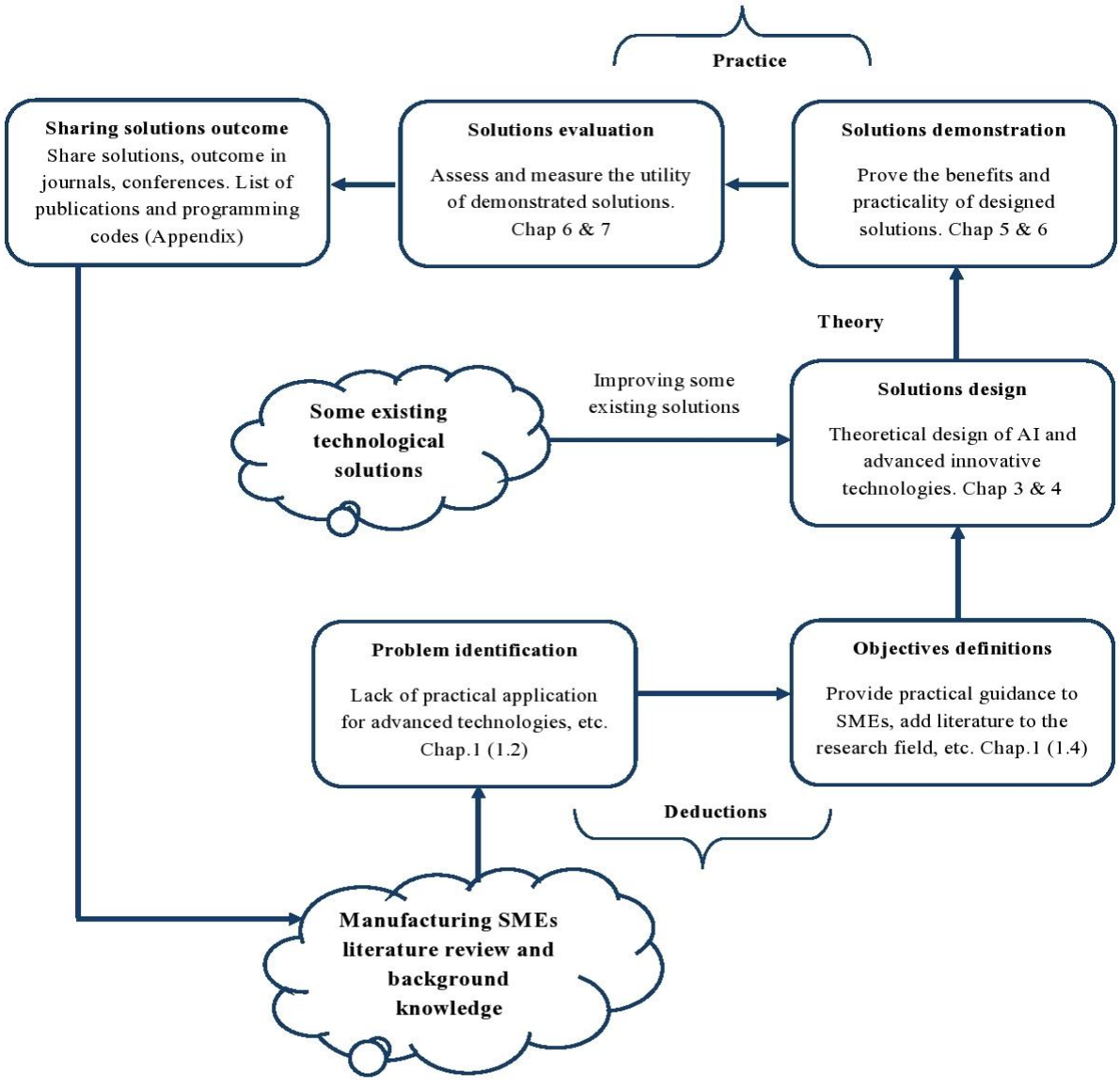

Figure 1-1: DSRM process flow adapted from Peffers et al. (2007)

Chapter 2: **Background and Literature review**

In this chapter, we conduct an explicit background and literature review on essential concepts utilized and resourced to reach our research goals: I40, SMEs, I40 challenges and opportunities for SMEs, AI, ML, IoT, predictive maintenance, product and mass customization (MC), Ethernet and network switches, and business model for innovation.

Chapter 3: **A conceptual business model guide for innovative SMEs**

This chapter aims to design a simple conceptual business model that can guide manufacturing SMEs planning to adopt innovative and new technological solutions in their organization. The conceptual business model is the foundation for developing our optimized AI methods in the remaining thesis chapters.

Chapter 4: **Developed AI techniques and innovative methods overview**

This chapter presents the innovative areas we focus on improving in this research using developed AI techniques and ingenious methods from the proposed conceptual business model: an intelligent PM framework for conveyor motors using time-series encoding and CNN, a robust network communication prototype with advanced communication technologies and network redundancy, and automatic parameter configuration method to transform a traditional SCADA system into an intelligent predictive device using MLR and DT ML algorithms, an adaptive customization framework for clients developed with XGBoost and RF ML algorithms and, and enhanced safety mechanism for AMR and human operators using Q-learning algorithm and speech recognition processes. We also highlight data collection and analysis information as the foundation for designing improved AI techniques.

Chapter 5: **An intelligent predictive maintenance (PM) framework for conveyor motors using CNN**

This chapter presents a detailed theoretical overview of concepts and technologies required to design the intelligent PM framework for conveyor motors using CNN: CNN algorithms, Principal Component Analysis (PCA), and time-series data imaging. In this chapter, we also design, implement and evaluate the impact of our proposed innovative solution results.

Chapter 6: **Improved network infrastructure with advanced communication techniques and redundancy protocols**

This chapter presents a detailed theoretical overview of advanced communication techniques and redundancy protocols implemented to design our improved network infrastructure: TSN, edge computing, and redundancy protocols (RSTP, MRP, PRP, and HSR). We also present the design, the implementation, and the evaluation of our proposed solution results.

Chapter 7: **Implementation of automatic parameter configuration for a SCADA system using ML techniques**

This chapter presents a detailed theoretical overview of ML techniques and algorithms implemented to develop our automatic parameter configuration scheme: Multiple Linear Regression (MLR) and Decision Tree (DT). We also present our proposed innovative solution results' detailed design, implementation, and evaluation.

Chapter 8: **An adaptive product customization platform for customers interactions with production system**

This chapter presents a detailed theoretical overview of ML techniques and algorithms implemented to develop the product customization platform: XGBoost ML regression and Random Forest (RF) classification. We also present our proposed adaptive product customization components design, implementation, and evaluation.

Chapter 9: **Creating an enhanced safety mechanism for operators and Autonomous Moving Robots (AMR) using Q-learning algorithm and speech recognition**

This chapter presents a detailed theoretical overview of concepts and technologies implemented to design our proposed enhanced safety response mechanism: Reinforcement learning (RL), Q-learning algorithm, and speech recognition. We also present the detailed design of our safety response mechanism, its implementation, and the evaluation of our results.

Chapter 10: **Conclusion and future recommendations**

This chapter provides a compact summary and detailed summary of the work done throughout this research. We illustrate how our results meet the objectives set at the beginning of the research. We suggest several recommendations on the way future studies could advance this research and improve our current results.

Chapter 2 : KEY CONCEPTS AND LITERATURE REVIEW

2.1 The impact of Industry 4.0 in SMEs

2.1.1 The Industry 4.0 (I40)

The I40 is the designation of the current industrial transformation characterized by the integration of information and communications technologies (ICT) and the internet into traditional industrial production processes to improve production systems with more intelligent strategies (Matt & Rauch 2020). The appellation I40 originates from the German Hannover Fair in 2013, where an academy of scientists Acatech referred to the I40 as the genesis of the fourth industrial revolution (4IR). Acatech introduced instructions and guidance for the successful implementation of I40 in the German government and industry (Kagermann *et al.* 2013). Since then, the concept of I40 has been expanded and adapted to different parts of the world. Few examples are "the Impresa 4.0", which is an adaptation of I40 for the Italian industry, "the Platform Industrie 4.0", widely utilized in Austria, and Thailand 4.0, used in Thailand (Orzes *et al.* 2020). It is worth mentioning that while the appellation I40 is widely employed in European countries, American territories utilize the term "Smart Manufacturing" to mention the fourth industrial revolution, and Asian countries speak about "Smart factory" (Thoben et al. 2017 and Mittal et al. 2018).

Some of the essential attributes of I40 can be summarized as follows (Davies 2015):

- **ICT's implementation:** The integration of ICT in the production system has the advantage of digitizing information from all the production stakeholders: customers, suppliers, logistics, and the factory bringing more flexibility and transparency in terms of production quantification, accountability, and system monitoring at every step of the production/supply chain (Spath *et al*. 2013).

- **The change from centralized to decentralized systems:** As opposed to traditional production systems, I40 components are more independent and self-configurable. They do not necessarily rely on a centralized master to operate but can make some decisions independently.

- **The introduction of product personalization and customization:** The I40 encourages customers to be part of the production process by increasing customization schemes from which customers can modify and edit products requirements as per their needs before the production begins. This attribute generates new production strategies such as configure-to-command, make-to-command, and build-to-command).

- **Cyber-Physical Systems (CPS) implementation:** The use of CPS in I40 production systems facilitates the self-monitoring, self-control, and the self-configuration of production components. CPS usually involves utilizing intelligent sensors, computers, actuators, and motors connected to the internet and favouring the combination of the digital (virtual) and the physical world (Rauch *et al.*, 2016).

- **Digital assistance implementation for production processes:** The implementation of I40 increases the use of the digital assistant of human workers such as AMR, intelligent machines, AR, and much more.

- **The increase of data analytics and big data technologies:** With the advent of I4.0, there is a significant increase of data in factories due to the interconnection between production components, internet connection, and intelligence added to system operations. There is an imperative need for appropriate platforms like big data to using the data to analyze and process them efficiently.

- **Network communication systems improvements:** The industrial network communication infrastructure of I40 systems has many internet and wireless technologies to achieve a better interconnection of all production stakeholders.

- **The increase use of simulation:** I40 systems stimulate methods such as product modeling and virtualization for manufacturing processes and product design before their implementation. The simulation of these processes is advantageous to make modifications before the production and save on the design.

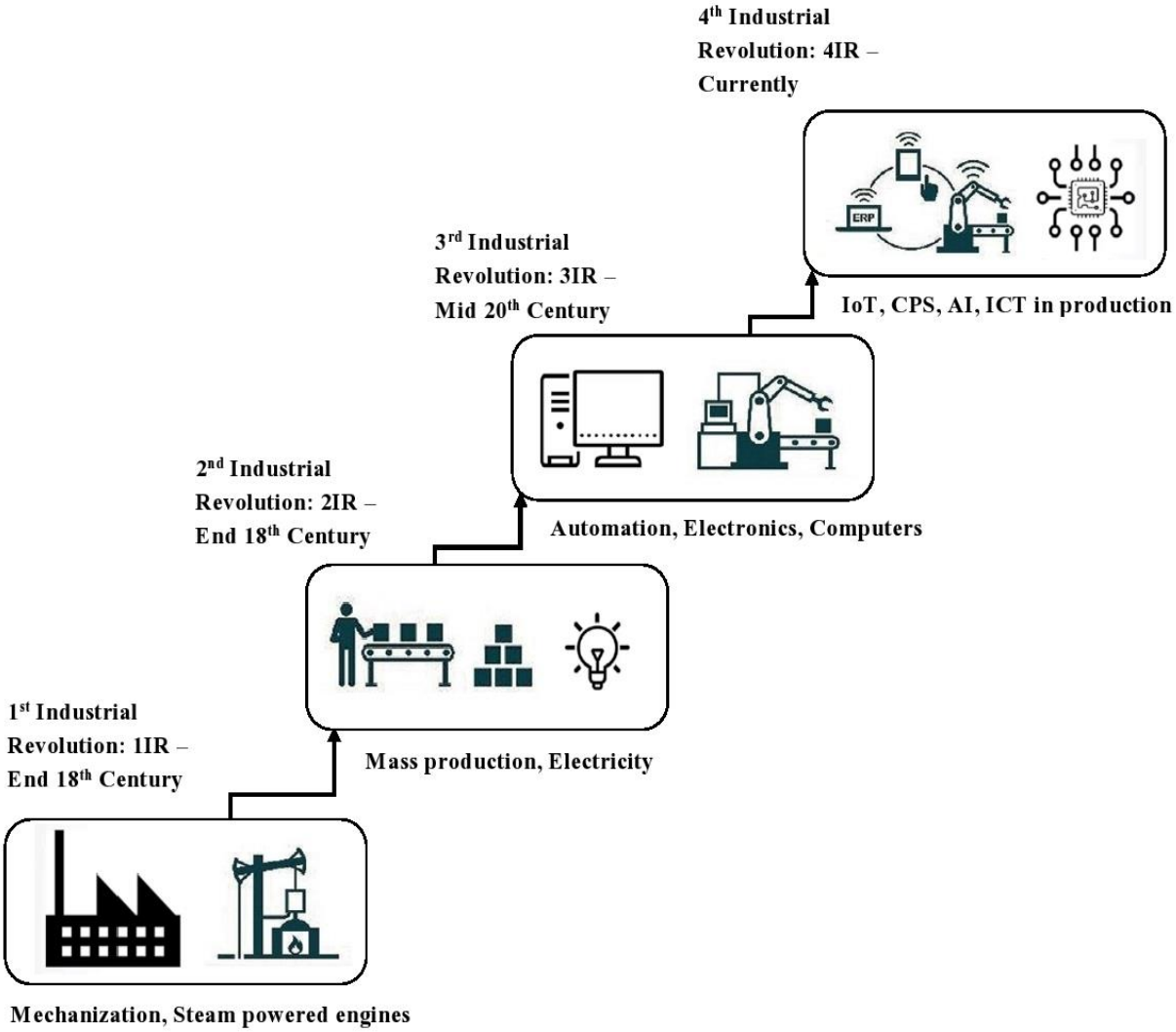

Figure 2-1: The four industrial revolutions schematic adjusted from Williams (2019) and Velliyur (2019).

The fourth industrial revolution comes after three major ones. The first industrial revolution began at the end of the eighteenth century (around 1765), distinguished by mechanization, machines (engines) powered by steam, and replacing the human-being muscles. The second industrial revolution came to light at the end of the nineteenth century (around 1870) with the invention of a new energy source such as oil, gas, and electricity (Kagermann *et al.*, 2013). During this industrial revolution the mass production and the assembly lines concepts started in factories. The 4IR comes after the third one that was initiated after the middle of the twentieth century (around 1969). In the third industrial revolution, there was an increase of human brain attributes in production processes with the integration of electronic components, computers, and telecommunication. The goal of the integrated components was to create the concept of

“automation” for production processes and bring intelligence to factories. PLCs and robots are two major components introduced to the industrial production processes during the third industrial revolution. Figure 2-1 is a graphical summary of the four industrial revolutions with a few of their significant features.The concept of I40 relies on several external technologies that play the role of enablers for a successful implementation of I40 technologies. Two of these technologies are “the AI” and “the IoT” (Hansen & Bøgh 2020).

2.1.2 Small to medium-sized enterprises (SMEs)

A relatively standard and straightforward definition of a small to medium-sized enterprise (SME), as per Liberto (2020), is an organization that conserves its number of employees, revenues, or assets below an established threshold. The definition of an SME threshold varies from one country to another or from one economic model to another. It is not easy to have a single explicit definition of an SME worldwide. We provide below some of the SMEs definition attributes worldwide (Muriithi 2017). The European Commission concluded in 2003 that, in the European context, an SME is a business whose annual turnover does not exceed 43 million euros and does not have more than 250 employees (Szedlak *et al.* 2020). In North America, when considering the United States (US) and Canada, an SME is an organization with not more than 499 employees**.** However, they consider a “small enterprise” a business with less than 100 employees within the SME category. In most developing countries, a business with more than 50 employees is considered a medium-sized enterprise, with fewer small enterprises. At some point, businesses with more than 100 employees were falling into large companies, but this threshold remains very debatable. We can notice that most SME definitions are quantitative with regards to employee numbers and assets. Ussif, & Salifu (2020) proposes a summary of SME attributes for developing countries, especially in Africa. We present the attributes summary in Table 2-1.

Table 2-1: Different SMEs attributes in an African context (Ussif & Salifu 2020)

Type of SME firms	**Number of employees**
Medium firms	50 to 250
Small firms	10 to 50
Micro firms	1 to 9

Although SMEs can seem negligible because of their sizes, they are the pillar of most economies worldwide, especially in developing countries (The World Bank 2021). The World Bank confirms that SMEs are major contributors to the worldwide GDP. It estimates SMEs' contribution to about half of the global GDP. The reason is that SME businesses constitute around 90% of worldwide businesses (Kamunge *et al.* 2014), which is indisputably the majority. They are also rated to employ over 50% of the population around the globe. In emerging countries, formal SMEs provide around 40% of the GDP. It was also estimated that seven out of ten jobs in developing countries are created by SMEs. When considering informal SMEs, which are quite considerable in developing countries, the contributing percentage of SMEs to the GDP increases even more. The worldwide projection for job creation by 2030 is currently at about 600 million. Therefore, the creation and improvement of SMEs become a priority for most economies.

A survey conducted by Ussif & Salifu (2020) for the contribution of SMEs in the sub-Saharan African economy provided the following results: formal Ghanaian SMEs contributed to 70% of the country GDP and 49% of the employment rate (Abor & Quartery 2010). In Kenya, SMEs were the source of 50% of the country's GDP and contributed to 80% of the overall country employment rate (Mwarari & Ngugi 2013). In South Africa, SMEs provided about 60% of the country's GDP and a similar rate in the employment sector (Willemse 2010). The same survey reported that formal SMEs contributed to 3.4% of the country's economy and 90% of the employment rate in Ethiopia. In Nigeria, SMEs contributed up to 50% of the country's GDP and 70% of the country's employment rate, as reported by Kolasiński (2012). The Rwandese SMEs contribute to 20.5% of their GDP and 60% of the employment rate of their country. When considering about 25,000 SMEs in emerging countries in the manufacturing sector, their contribution to the total sales is about 7.6% versus 14.1% contributed by larger companies (World Trade Report, WTO 2016).

When comparing SMEs to larger companies, we can pinpoint a few of the blatant differences at various levels. On one side, SMEs' internal organizations are not as structured as in larger companies. SMEs do not often invest in R&D activities to improve their processes, while most large companies do. On the other side, SMEs are usually readily willing to take more considerable business risks than larger companies. SMEs are not very bureaucratic hence business risks decisions can be taken faster. SMEs are prone to smoothly adapt to market adjustments instead of larger organizations (Truve *et al.*, 2019). Parida et al. (2012)

demonstrate that SMEs are more favored to gain in new technological advancement and innovative ventures because of the structural flexibility.

2.1.3 Opportunities and challenges of the I40 adoption by SMEs

The worldwide economic impact of SMEs arouses the need to provide them with tools, guidance, and measures that assist them to grow and prevail in a highly competitive market. Advanced technological trends and innovative techniques in the likes of I40 are some of the measures expected to improve SMEs' operations remarkably. Various sources (Yang & Gu 2021; Matt & Rauch 2020; Petrillo *et al.* 2018, and ATCC Finance 2015) indicates that large companies are actively integrating I40 practices and technologies in their organization while most SMEs are still reluctant and struggling to come on board. Matt et al. (2018) reported that manufacturing SMEs are already aware of the new industrial revolution but face challenges in adopting advanced practices to improve their production processes. Several interviewed SMEs stated that they needed guidance to initiate I40 practices (Ganzarain & Errasti 2016). A general term of realizing and adapting I40 within SMEs is "SME 4.0". The appellation refers to a group of adjusted I40 methods regarding SMEs' environment (Matt & Rauch 2020).

2.1.3.1 Opportunities of I40 in SMEs

A study by ATCC Finance (2015) demonstrated that SMEs could highly benefit from the innovative advantages of the current industrial revolution. It is mainly due to their flexible structures allowing easy implementation of new technologies. Matt (2007) and Matt et al. (2016) declared that based on the outcome of the previous global economic and financial crisis, SMEs had shown the ability to be more resistant to economic challenges than multi-national and large organizations.

The I40 ensures to offer several advantages to any manufacturing business in general and to manufacturing SMEs in particular. A few of these opportunities are:

- **Increase their competitiveness:** The global manufacturing market witnesses the birth of new businesses at an exponential rate. The use of the latest technological practices in production processes allow businesses to remain competitive. Using

I40 methods in their factories, manufacturing SMEs increase their competitiveness in the global market (ATCC Finance 2015).

- **Grow their business venture:** The use of I40 concepts enables manufacturing SMEs to expand their current business activities in line with new technologies..

- **Grow their customers' database**: By using I40 concepts such as product customization and personalization, manufacturing SMEs allow customers to be part of the production processing. It increases customer satisfaction, their value and provides an additional means to attract more clients.

- **Increase manufacturing productivity**: Proper implementation of I40 techniques in production processes increases the system productivity using existing methods, for example, predictive maintenance empowered by intelligent techniques such as AI to achieve better results (Kagermann *et al.* 2013 and Wenking *et al.* 2016).

As per Rauch et al. (2020), SMEs should not be dubious about adopting I40 methods in their production processes; they should instead focus on the best and fastest possible ways to enable the implementation of I40 in order to remain relevant facing a competitive market.

Mittal et al. (2018) reports that the successful application of I40 techniques in manufacturing SMEs depends highly on designing practical SMEs-customized methods and strategies more adapted to an SME environment. There is a crucial need for best practices, practical guides, and methods to assist SMEs practitioners in implementing I40 methods. They also provided few requirements for the adoption of I40 by manufacturing SMEs summarized as follows:

- An integration of the I40 in SMEs businesses culture
- A cooperation between SMEs and some specialized research institutions and universities
- Sufficient financial resources to implement I40 technologies
- An implementation of advanced manufacturing methods in production processes
- An integration of all production stakeholders to the I40 culture: customers, employees, and suppliers.

Our research includes some of these requirements to suggest a practical conceptual business model that upholds the adoption of I40 by manufacturing SMEs.

2.1.3.2 SMEs challenges in adopting I40

Although manufacturing SMEs constitute the backbone of several countries' economies, they face numerous challenges when it comes to adopting I40 technologies. Some of these challenges are due to missing human and financial resources to assess the I40 requirements and introduce factories to the current trend of automation (Matt *et al.* 2020). Jovanovski et al. (2019) states that several SMEs still implement a high level of manual operations and activities in their factory shop floor compared to large corporations making it harder to adopt fully automated processes. In order to guarantee their future existence in the competitive manufacturing market, SMEs should not ignore the adoption of I40 methods but think of possible ways to slowly but surely implement these techniques (Mittal *et al.* 2018). The adoption of I40 in manufacturing SMEs will more likely be a progressive activity rather than a drastic change.

We present a series of common challenges and obstacles slowing the implementation of I40 in manufacturing SMEs adapted from Jovanovski et al. (2019) and Matt et al. (2020) findings:

- Insufficient financial resources to invest in new technologies and on R&D is crucial to study the concrete feasibility of implementing I40 techniques in the production process. R&D should also be able to determine the economic benefit of applying I40 methods.

- Insufficient skilled personnel that understands advanced technologies and high-tech solutions.

- The absence of an innovative culture that first integrates employees' experience to improve the production process, then other production stakeholders such as suppliers and customers.

- The need to create new business models that favour the implementation of innovative techniques.

- The lack of customized-SMEs practical techniques to support them in initiating I40 methods.

- The fear of exposing manufacturing data through the internet and to cyber attackers.

Although some researchers (Truve et al. 2019; Jovanovski et al. 2019; Dossou 2019 and Ganta 2020) aspire to guide SMEs through the adoption of I40 methods, they are mainly frameworks or high-level approaches lacking a touch of practicality to ease the understanding and the implementation of I40 in manufacturing SMEs. Several other studies in this area are surveys assessing the current state of I40 application within SMEs (Rauch et al. 2020; Safar et al. 2020; Ingaldi & Ulewicz 2019, and Prause 2019). Our research intends to bring a more practical aspect to proper innovative techniques enabling a starting point for applying I40 methods in manufacturing SMEs. An extraordinary study was conducted by Matt et al. (2020), combining several surveys and advanced practical I40 methods for European SMEs.

2.2 Artificial Intelligence (AI) & Internet of things (IoT) enablers of the I40

As previously mentioned in the I40 study segment, various technologies enforce I40 methods (Deloitte Insights 2020). This section reviews Artificial Intelligence (AI) and the Internet of Things (IoT) as two enablers of I40.

2.2.1 Artificial intelligence (AI)

AI is a hot topic in several life domains because of its already existing innovative solutions and its numerous guarantees to offer more applications aiming to ease our daily lives. Several definitions exist to explain the reality behind this term. For example, Haenlein & Kaplan (2019) describes AI as an intelligent structure capable of interpreting data meanings, learning crucial and intrinsic information from this data, and utilizing this information to provide several

solutions, to build applications, or to reach any desired goal after various modifications. Marki et al. (2016) summarizes AI as an analytical activity created by computer-based programs to design and produce intelligent solutions. It is worth raising that the human brain inspires the creation of AI. It aims to give computers and machines the capability of making decisions, learning, and analyzing different problems, especially those requiring complex approaches (Najafabadi et al. 2016) like the human brain does. AI systems can then be applied to offer solutions to various life areas. Malica (2012) sums up some of the benefits of AI in reducing human efforts for various brain demanding and robotic tasks, easing complex problem solving, and reducing the labor cost of some repetitive activities.

AI offers several opportunities for the successful implementation of new technological trends such as I40 or smart manufacturing. It increases and improves the creation of supporting manufacturing tools and measures in predictive and preventive maintenance, faults and error detection systems (Rauch *et al.* 2019), intelligent inventory system control, planning, and forecasting (Min 2010). Applying these AI-improved manufacturing process supporting measures increases the efficiency of businesses that take the risk to implement them (Rönnberg & Areback 2020). AI also proposes innovative ways to make humans and machines work together to their full potential (Wangler & Botthof, 2019). In an I40 scenario, the combination of the physical (IT-based systems) and the virtual (CPS) world and the proper evaluation of data streams coming from it are possible through the utilization of AI technologies. The growth and the flourishing application of I40 rely on the excellent exploitation of AI technologies (Veit *et al.* 2017; Duan *et al.* 2019 and Szedlak *et al.* 2020).

By analyzing the growth of the present technological trend, we could conclude that it would be almost impossible to stay away from the use of AI, particularly in the manufacturing section embracing the 4IR. The enormous amount of data currently available, especially in large manufacturing companies, and those intended to be created by increased manufacturing tools and components require specialized technology to handle them. Furthermore, when associating the data exchange between all manufacturing factories stakeholders through a medium like the internet implementing AI techniques to handle them becomes almost compulsory (Stalidis *et al.* 2015). I40 favors intelligent manufacturing concepts such as mass customization (Complexity Management in Mass Customization SMEs) over traditional mass production that generates more data through a product manufacturing life-cycle. The Big Data concept relates to the large amount of information generated by intelligent devices through exchange and

interconnection. The same reality applies to an I40 context (Qi & Tao 2018). AI comprises several branches for application in different domains. One popular AI segment that we cover in this research is machine learning (ML). ML has its various powerful techniques and algorithms applied to create intelligent models. Figure 2-2 is a graphical representation of a few AI segments.

The implementation of AI is gaining broadness in the IT environment, and some popular software companies like Facebook and Google dealing with a massive amount of information and data daily (Abadi *et al.* 2019 and Paszke *et al.* 2017). An AI learning technique called Deep learning is currently used as a robust base algorithm to process, compute, and analyze large datasets effectively. Deep learning is recently making use of techniques such as image classification from collected extensive data and has proven to achieve better results than previous traditional ML data algorithms (Schmidhuber 2015; Redmon & Farhadi 2018; Mnih et al. 2015 and Hessel *et al.* 2018). In the industrial manufacturing sector, although there is presently fewer impact of AI applications than in the IT environment, several studies and research are conducted to gain from the numerous advantages of AI and improve industrial production systems. Researchers have adapted AI techniques to achieve outstanding results in domains like maintenance, predictive analysis, and fault diagnosis (Wang et al. 2018). Cloud computing platforms such as Microsoft Azure providing AI analytics and big data computing are made available to large manufacturing companies on their quest to adopt I40 methods (Barga et al. 2015).

2.2.1.1 Machine Learning (ML)

ML is a well-known AI segment created to train computers and machines to manipulate data (Dey 2016) beneficially. Its goal is to extract valuable knowledge or pattern from data and build intelligent models to estimate or predict different information. ML comprises various algorithms ranging from simple lower-hand, applicable to relatively uncomplicated tasks, to robust algorithms appropriate for larger datasets and complex systems. ML algorithms have the incredible ability to go beyond known pattern recognition and recognize features on a new dataset they were not previously exposed to (Mullainathan & Spiess 2017).

We classify ML algorithms in six major classes representing different types of learning systems for computer structures (Truve et al. 2019 and Dey 2016). Figure 2-3 is an overview of the six ML categories with a few of their most common algorithms.

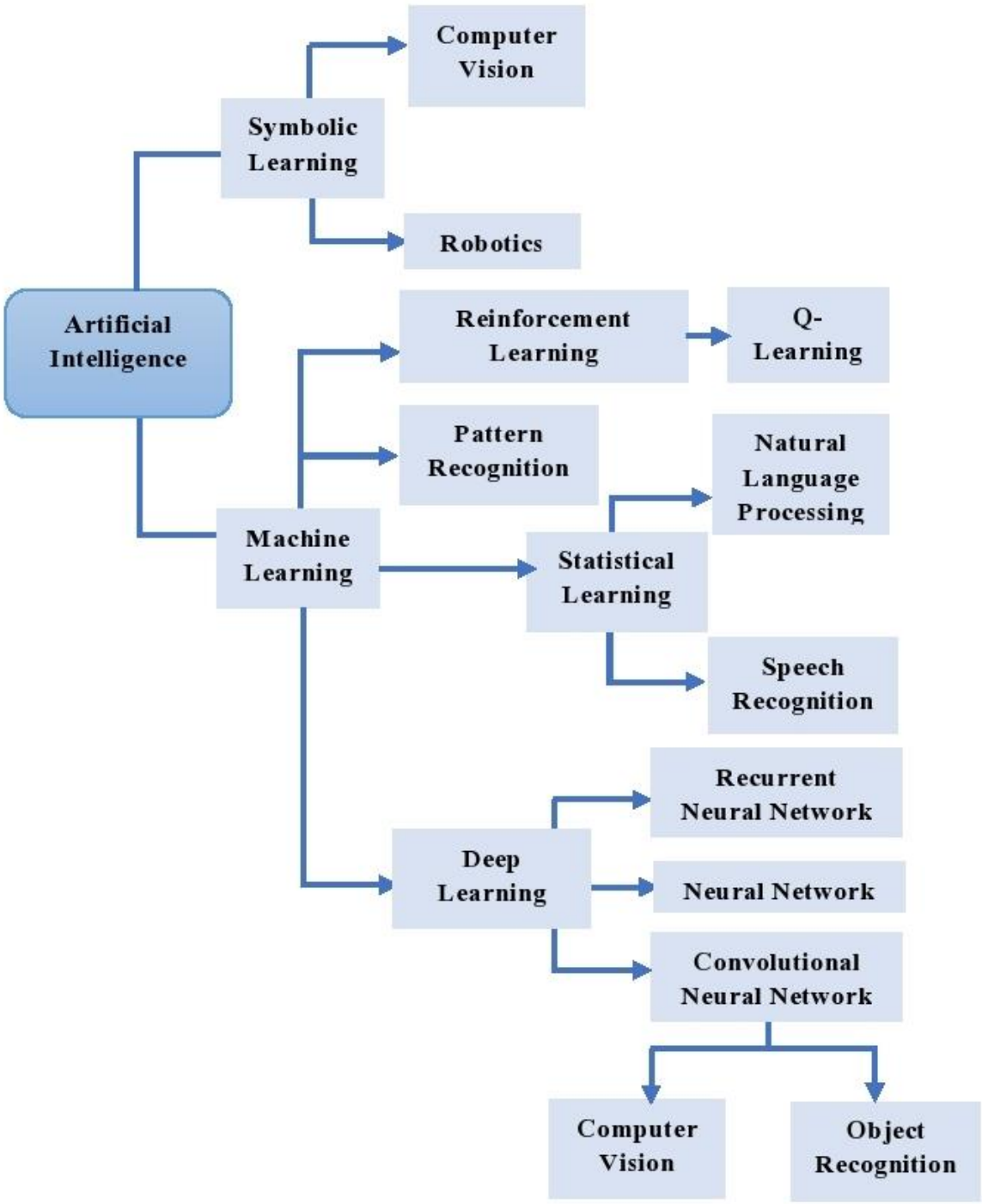

Figure 2-2: An overview summary on few AI branches

- **The supervised learning category:** In the supervised learning category, ML algorithms require supervisors' assistance to start the learning process on the data. The dataset contains labels that clarify the data composition to differentiate dataset classes from one another. As part of the learning process, the algorithm divides the dataset into two subsets: a training set and a test set. The training set has a higher proportion than the test set and contains the desired output variables for the classification or the prediction tasks (Kotsiantis 2007). The supervised learning algorithms utilize the test set to evaluate the reliability of the ML model built.

- **The unsupervised learning category:** The unsupervised learning category is quite the opposite of the supervised learning one. ML algorithms deal with unlabeled data, a dataset with no explicit knowledge of data information patterns. The unsupervised learning ML algorithms can extract, on their own without supervisors' assistance, some features from the dataset they are exposed to and later use those features to perform prediction or classification activities.

- **The reinforcement learning (RL) category:** The RL category is a learning system inspired by the psychological model of recompensing positive actions and punishing negative ones. RL algorithms learn independently, with no external assistance, using a trial and error method (Sutton 1992) to discover the best possible decision based on positive rewards. The RL category is suitable for problems where there are no explicit datasets available or distinct dataset categories.

- **The ensemble learning category:** The ensemble learning algorithm consists of combining several individual learner algorithms to obtain a more powerful learner. Some studies prove that ensemble learning algorithms are more likely to produce better, more accurate, and stable models than individual ML algorithms (Opitz & Maclin 1999). The ensemble learning algorithm achieves outstanding outcomes by using two major processes. The first process exposes its base individual learning algorithms, also known as the based predictors, to several datasets perspective from which they gain a better understanding of existing features. This process is known as heterogeneous learning. In the second process, the ensemble learning algorithm learns information on

a single algorithm retrieving information from several random sub-datasets of the initial training set, called homogeneous learning (Pharm *et al.*, 2021).

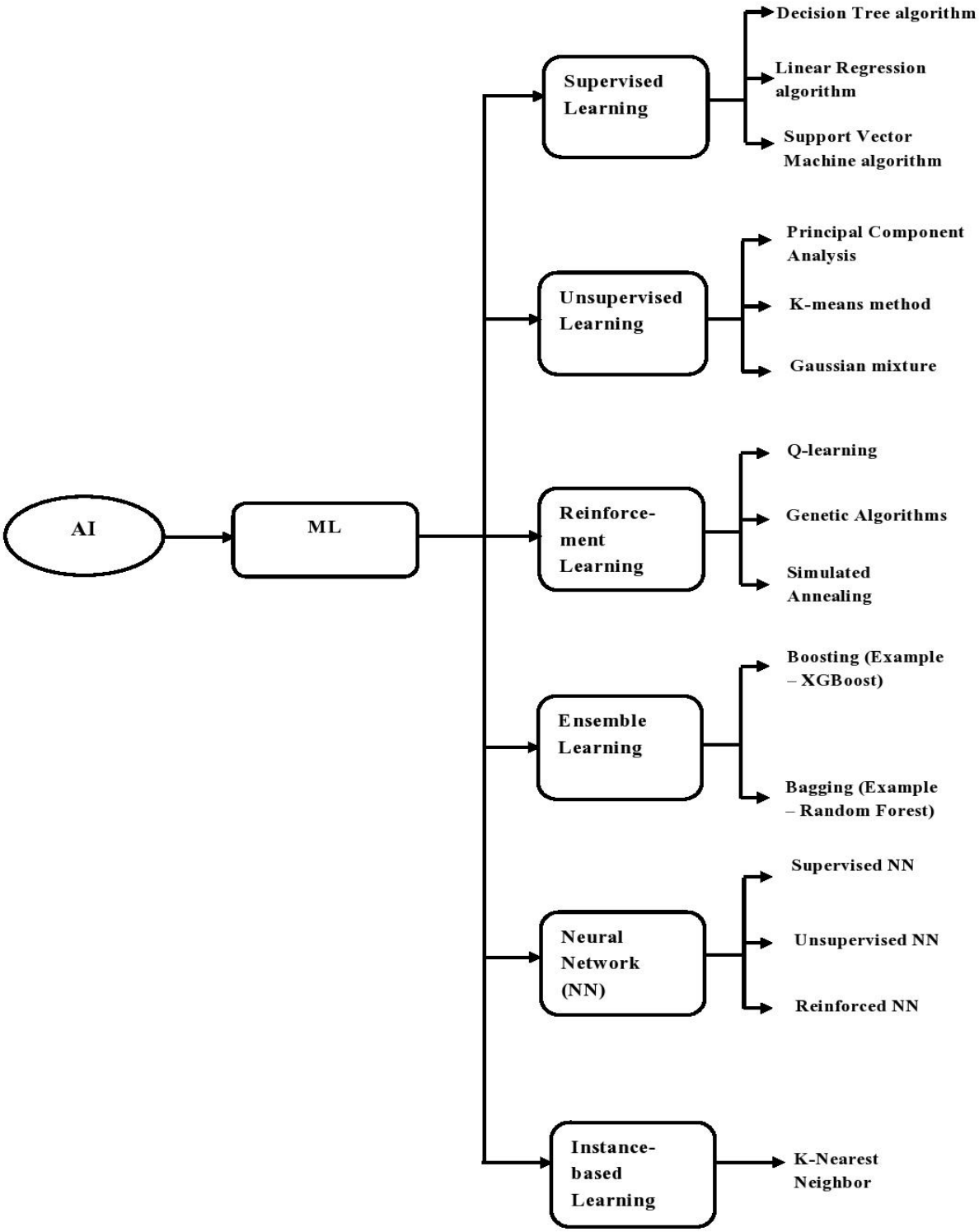

Figure 2-3: ML categories overview

The ensemble learning algorithm category contains two popular learning techniques: the boosting and the bagging technique. The boosting learning technique groups several individual-based predictors' output weak results to reduce their variance and bias to create a more robust predictor. We display in Figure 2-4 a graphical representing of the boosting ensemble learning technique. The bagging, standing for bootstrap aggregating, learning technique trains several independent ML classifiers generated from training on random sub-training sets of the original training set and combines their classification results to improve the final classification outcome. Figure 2-5 is a representation of the bagging learning technique operation. The bagging learning algorithm can reduce the risk of overfitting in the ML model building process (Dey 2016).

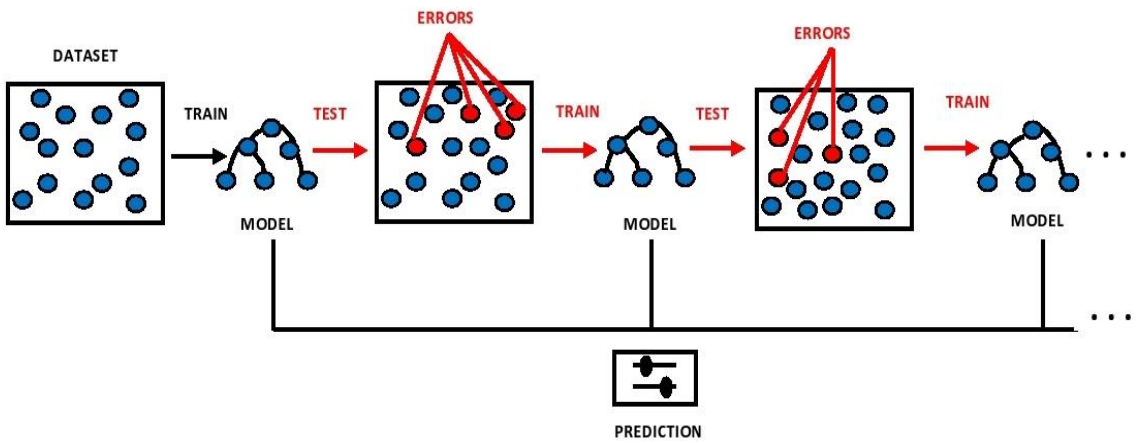

Figure 2-4: Boosting learning operation principle (Kiangala & Wang 2021*a* and Ramzai 2019)

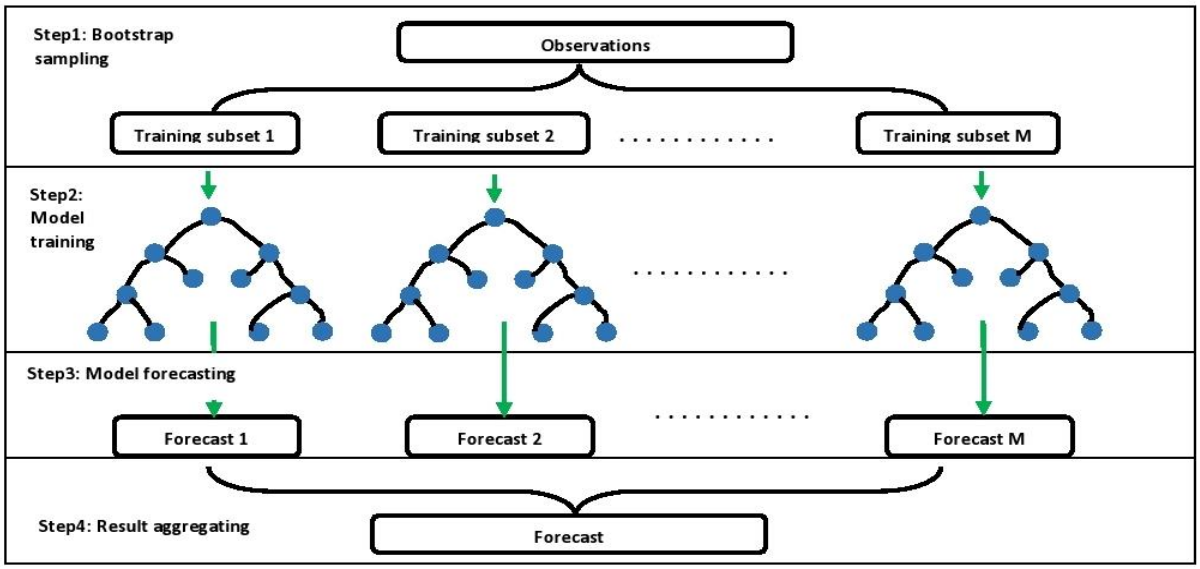

Figure 2-5: Bagging learning operation principle (Kiangala & Wang 2021*a* and Ramzai 2019)

- **The neural network category**: The neural network (NN) category is one of the most advanced ML algorithm classes. It contains various appropriate algorithms for relatively complex tasks. This category is often referred to as artificial neural networks (ANN). The NN ML algorithms design comes from the working principle of a neuron, a crucial human brain cell for decision making and learning abilities.

- **The instance-based learning category**: The instance-based learning category is a lazy ML algorithm method because it always relies on previously learned patterns to apply to the new fed dataset without including them in the training process to gain additional knowledge from them. It assumes that the new fed dataset has features already incorporated in its previously generated model. This method is appropriate for neighboring data or data of the same cluster with similar attributes.

2.2.2 Internet of Things (IoT)

Another interesting topic like AI is the IoT. It refers to the interconnection of several entities and devices such as computers, sensors, machinery, processes, and other electronic components in a local network or through the internet. The IoT devices can exchange information and be identified in their networks, where they perform several tasks individually without human assistance. The concept of IoT facilitates data collection from smart devices and real-time monitoring to analyze and build intelligent systems (Falkenreck & Wagner 2017)**.** Nowadays, the utilization of IoT devices is more and more common in various areas. For example, in the health sector, IoT devices provide various reporting for first health care services (Pasluosta et al. 2015) and condition monitoring of patients (Yang et al. 2014). The area of smart homes also intensively utilizes IoT technology to monitor and control home functions such as air conditioning adjustments, electricity control, motion sensing, alarm activation, and several more activities (Lin et al. 2017). IoT devices often have the following attributes in common: embedded intelligent computing capability to perform various functions on their own, advanced sensing materials to collect valuable information, and analytical functions for the collected data. In the industrial section, an adapted version of The IoT with similar functions and advantages for the industrial environment is called the Industrial Internet of Things (IIoT) (Schriegel et al. 2018 and Kharb & Singhrova 2019). The IIoT utilizes several advanced methods like cloud computing, enhanced automation functions, energy management, and much more (Perera et al. 2014) to improve various existing traditional manufacturing processes. The

IIoT enables improved data analysis and computation of information exchanged by IIoT devices in local or remote cloud servers having more excellent computational capabilities than individual IIoT devices, especially for a large amount of data (Rezaeibagha et al. 2019). Because of the requirement to continuously transmit information between IIoT devices, IIoT communication networks need to be robust, have a high network availability, have a low communication latency, and allow direct communication between cloud servers and IIoT sensors (Kobzan et al. 2018). Therefore, the implementation of appropriate network infrastructure is crucial for the successful communication of IIoT components.

Some communication standards will empower even more the potentialities of IoT systems: the IEEE 802.11ax communication standard, commonly known as Wi-Fi 6, is intended to improve the utilization of spectral resources, to enlarge the communication throughput, and to better the power efficiency (Afaqui et al. 2017). The fifth generation of the telecommunication standard (5G) can enlarge IoT features regarding latency reduction and the communication speed increase (Li, S. et al. 2018). The application of the reserved internet protocol IPv6 IP addresses ranges to cater for the future growing number of IoT devices (Madakam et al. 2015). Few common IoT communication standards for consumers are Lora, Sigfox, and NB-IoT (Hansen & Bøgh 2020). Hansen & Bøgh (2020) develop an exciting diagram, presented in Figure 2-6**,** demonstrating the possible outcomes of implementing AI and IoT methods within manufacturing SMEs.

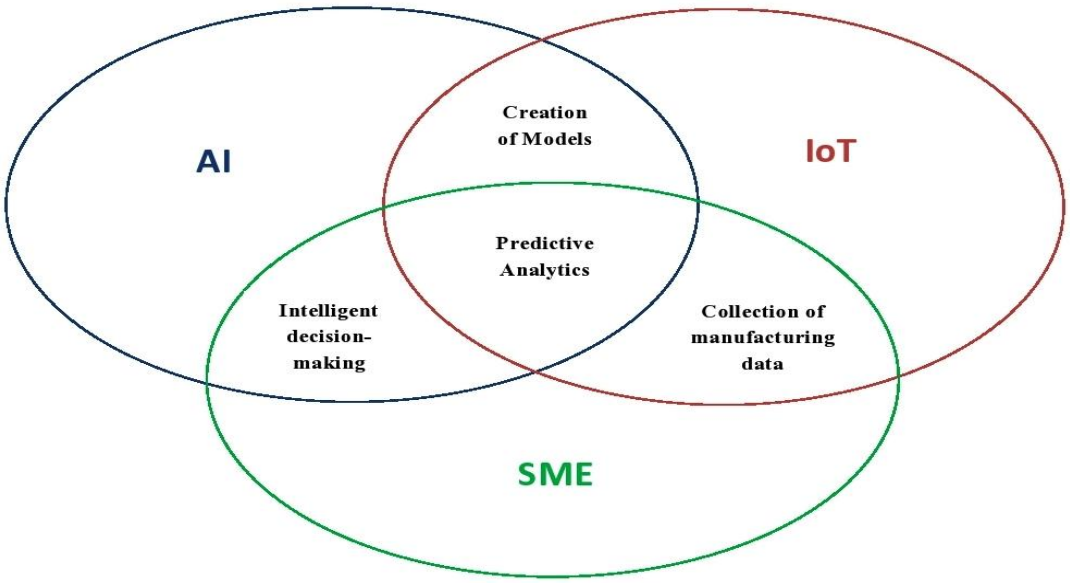

Figure 2-6: Interconnection of AI, IoT and SMEs (Hansen & Bøgh 2020)

2.3 Advanced predictive maintenance techniques for industrial manufacturing

2.3.1 Predictive maintenance

Asset degradation causing system unavailability or shutdown is one of the essential motivations behind the creation of PM (Diamond & Marfatia 2013). PM is an insightful procedure that detects any possible assets threats or upcoming failures before they change into critical conditions and anticipating their maintenance activities accordingly. The PM concept contributes to maintenance schedule optimization for various assets by determining machinery's most suitable maintenance timing. PM often uses data-powered methods to achieve early fault detection (Meyer Zu Wickern 2019). In the era of I40, IoT devices empower the implementation of PM techniques in several sectors (Spiegel *et al.* 2018).

A PM process can establish an asset's estimated remaining life-cycle through specific data collection and device monitoring. Therefore, the PM process is a valuable tool that can be incorporated in production processes to improve maintenance and manufacturing systems and the overall production value (Grall et al., 2002). Unlike previous maintenance policies, the PM process does not depend on a regular periodic interval to initiate repairs and services. It reads real-time information from the monitored asset and acts accordingly. This procedure lessens the risks of unnecessary maintenance tasks resulting in system downtime and loss of resources. When implemented efficiently, a PM process has the high potential of improving an overall system performance (Nguyen *et al.* 2015 and Yam *et al.* 2001).

A part of the PM process we just described, we can explore two other maintenance policies, known as conventional maintenance techniques: correctional maintenance (reactive or run-to-failure maintenance) and preventive maintenance (scheduled maintenance).

- **Corrective maintenance**: A corrective maintenance policy, also known as reactive maintenance or run-to-failure maintenance, is a procedure that occurs only after a system breakdown or when a device fails. This maintenance procedure is the most simple to implement as it does not require any additional device, program, or software integration in a running system, but it is, at the same time, perilous for companies since it produces extended system downtime while fixing the faulty equipment (Krupitzer *et al.* 2020).

- **Preventive maintenance**: A preventive maintenance policy, also known as scheduled maintenance, is a procedure that initiates maintenance activities based on scheduled periodic intervals or based on process changes. Although the preventive maintenance policy is quite effective in detecting early threats, it is not very cost-effective because of the use of resources on repetitive maintenance tasks that are usually unnecessary for the system.

We present in Figure 2-7 a schematic diagram of few maintenance policies as per Krupitzer et al. (2020).

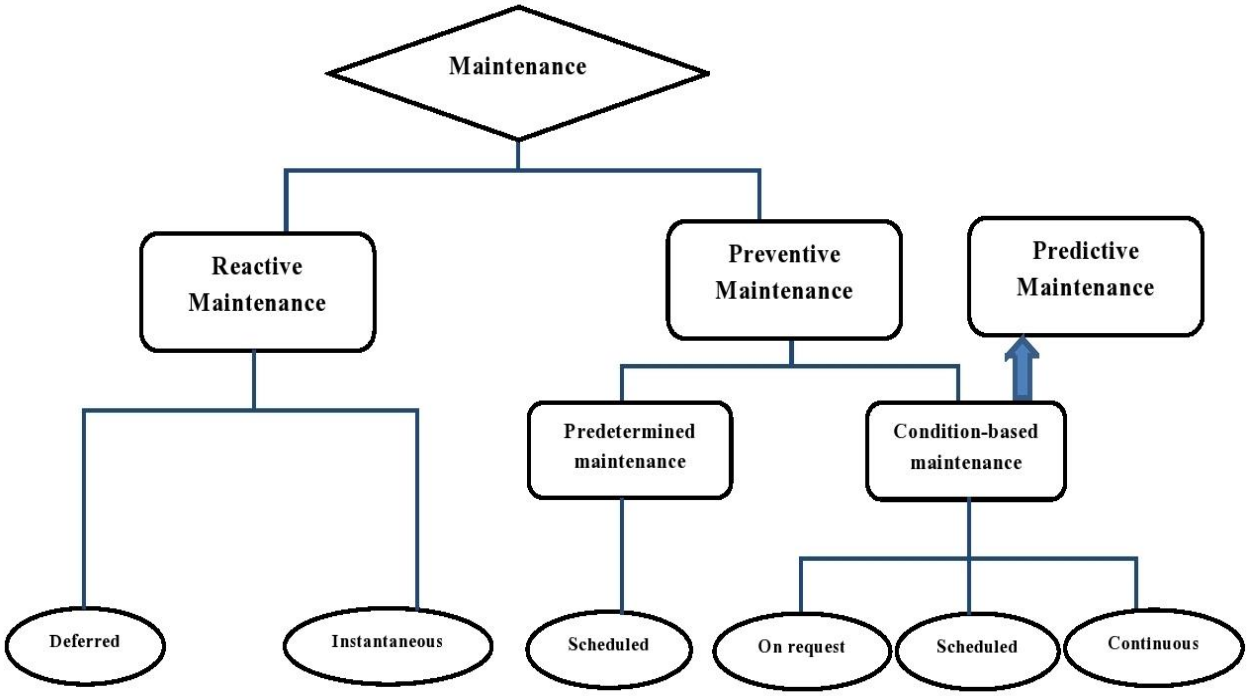

Figure 2-7: Maintenance policies schematic diagram (Krupitzer et al. 2020).

2.3.1.1 Some additional advantages of predictive maintenance

- **Improving the Overall Equipment Effectiveness (OEE):** By considerably reducing the risks of unplanned system failure, PM improves systems OEE, a key performance

indicator (KPI) measured by machine performance, output quality, and availability (Venkatesh 2007).

- **Improving product output quality:** As per Jin et al. (2016)**,** a product's output quality highly relies on the excellent health status of its production machinery. Defected parts on a system harm the final product outcome (Iravani & Duenyas 2002 and Soltani 2018). By detecting early machinery faults, the PM enhances a machine heath status and improves product quality.

- **Improving stock management:** Because of its ability to detect early devices at risk of failures, PM can be included in stock management to anticipate the purchase or the reservation of repair parts and keep them in stock until the maintenance activity starts.

- **Enhancing Remote Condition Monitoring (RCM)**: The PM concept actively uses condition-based monitoring (CBM) to withdraw information about the health status of machinery. When applied to remote sensing technologies, a PM procedure becomes a practical means of continuous insightful feedback on machinery located in remote and unsafe areas for human beings (Golightly *et al.* 2017).

2.3.1.2 Some challenges of Predictive Maintenance

- **Lack of sufficient financial means:** Implementing a PM procedure requires various additional resources: hardware, software, and human resources to extract information, install sensors, create and improve PM models. Not all companies (SMEs in particular) are always willing to invest in those additions, especially without a proven return on investment (ROI).

- **Limited source of data:** As previously described, PM is data-driven technology. The successful application and outcome of a PM process depend thoroughly on the viability of data monitored. A reliable source of data is crucial to design a PM model. Diamond & Marfatia (2013) reported that data collection represents, currently, a challenge in

several companies because of the lack of intelligent sensors or appropriate systems to extract useful information.

- **Lack of visibility on actual machine maintenance activities**: Although the PM process performs a remarkable job in detecting early faults and abnormalities, it is not responsible for conducting the actual maintenance task, which depends on the human operators' skills to produce expected results (Lee *et al.* 2014).

- **Lack of innovation culture:** The PM process is a relatively new concept that falls within innovative technologies for organizations. Unfortunately, several companies, especially SMEs, do not have the appropriate business models that encourage the implementation of innovative techniques such as PM.

2.3.1.3 Predictive Maintenance techniques using ML methods

The promising benefits of PM to enhance the overall reliability of production processes have encouraged several researchers to investigate various intelligent techniques to improve the effectiveness of the PM concept. For the past years, the research trend has focused more on incorporating ML techniques and algorithms to better PM procedures (Kiangala & Wang 2020 a).

Coraddu et al. (2014) proposes a PM structure built on Support Vector Machine (SVM) and Regularized Least Square (RLS) ML methods to predict the best maintenance time of gas turbines depending on few collected data such as the gas turbines decay, its speed, the compressor decay. Based on the results accuracy obtained, they concluded that the SVM algorithm was more suitable than RLS on their specific gas turbine PM model. Leahy et al. (2016) conducted a similar study that implemented the SVM algorithm to design a PM framework for wind turbines. They developed an SVM classification model capable of predicting up to six wind turbines' faulty conditions referred to as feeding faults, fault/no-fault, excitation faults, air cooling faults, mains failure faults, and excitation faults. Susto et al. (2015) directed their studies in the semiconductors manufacturing industry by designing a PM scheme for the early detection of faulty tungsten filaments while in the ion implantation process. They

utilized two ML algorithms: SVM and K-Nearest Neighbor (KNN), to compare the accuracies of their PM models and found that SVM slightly outperformed KNN. Paolanti et al. (2018) built an RF classification ML model in a PM system to predict a cutting machine's possible threats using its rotor statuses as input of the ML classification model. Kulkarni et al. (2018) designed an RF ML classifier to integrate into a PM procedure to predict early components outages in a refrigerator system. They implemented a feature extraction method in the early stages of their ML modeling process to learn various dataset patterns and seasonality decomposition through clustering and dynamic time wrapping. Their PM classifier was able to determine whether the dataset pattern was portraying an abnormality or not.

Zenisek et al. (2019) implemented Symbolic Regression (SR), RF, and Linear Regression (LR) to design a PM framework that monitors industrial machines' health conditions. Using continuous data streams, they developed a PM system that predicts and detects industrial machine drifting behavior. They tested their PM methodology accuracy on some real industrial fans and achieved successful results for detecting concepts drifts. Nevertheless, they encounter few challenges for predicting concept drifts because of the absence of reliable deterioration data. Another interesting PM framework with ML techniques came from Hesser & Markert (2019). They created an ANN classification model to determine the health statuses of milling machinery in a Computer Numerical Control (CNC) system. They monitored the acceleration data of milling machines for the PM system. They installed sensors in a programmable sample platform to collect relevant data from the tool wear and monitor it. The authors built two other classification models with the same data using SVM and KNN to choose the best-performing system. They found out that the initial ANN classification model outperformed the two other models. Falamarzi et al. (2019) created a PM system using SVM and ANN ML techniques to detect impairments and degradations in the gauge structures. They implemented their PM system for two rail tracks categories that integrated curved and straight segments. The performance evaluation of the ML models was done using a coefficient of determination and mean squared error. The two ML models (SVM and ANN) produced accurate results, but the SVM model outperformed the ANN model to predict gauge deviation in curve parts of rail tracks. For the forecasting of printing machines failures, Binding et al. (2019) constructed a PM framework utilizing historical printing machine unstructured data to build ML classification models with Extreme Gradient Boosting (XGBoost), LR, and RF algorithms. From the historical unstructured printing machine data, the researchers chose the monitoring of information such as the segment or the area under the receiver characteristic curve (AUC),

empirical cross-entropy, receiver operating characteristic curve (ROC), number of true negatives (TN), and false positives (FP) located at several calibration curves and threshold of the estimated probabilities, and the precision-recall curve (PRC). In assessing the results, they established that XGBoost and RF models were the best models for decision thresholds.

All these studies prove that there is great potential in integrating ML techniques and algorithms to create more intelligent PM frameworks. ML methods are flexible and can be adapted to resolve different problem scenarios. Researchers should exploit this attribute for PM systems in SMEs that need customized solutions applicable to their environment.

2.4 Product and mass customization in an Industry 4.0 environment

2.4.1 Product and mass customization

One of the current industrial revolution goals is to significantly increase customers' satisfaction by implementing advanced manufacturing and business concepts. Product customization is an advanced manufacturing method that continuously includes customers' personalized requirements in designing and manufacturing their goods before starting the production process (Wan et al. 2016). Unlike traditional manufacturing processes, where customers have limited choices because of pre-manufactured products without their contribution, modern manufacturing processes involve clients' customized preferences through the production process to end up with products that better meet their customers' needs.

Inspired by the operational principle of product customization, Davis (1989) introduced a now popular manufacturing concept known as mass customization (MC). MC is an advanced business approach that aims to manufacture and sell products personalized to meet individual customers' requirements while maintaining the low-cost advantage of mass-produced goods (Brunoe & Nielsen 2016). In short, MC merges the benefits of a low-cost mass production process and the flexibility of product customization to increase customers' satisfaction (Ismail et al. 2007). In the early 1990s, Pine (1993) played a massive role in popularizing and operationalizing the concept of MC after its introduction by Davis (1989). Presently, the

research interest in MC applications is considerably increasing, and the concept is gaining ground in several industries.

Brunoe & Nielsen (2016) reported that businesses that decide to adopt MC in their production processes should focus on developing the following competencies:

- A thorough identification of their various product features based on diverges customers' requirements. This process is known as "Solution space development."

- A capability to convert and improve their existing production structures and organization resources into "Robust process operation."

- A user-friendly or self-explanatory system that teaches customers how to derive personalized products and solutions that effectively respond to their needs.

Pine (1993) defines four significant customization process categories that permit clients to personalize product requirements before the actual production starts:

- **Adaptive customization process**: An adaptive customization process consists of personalizing a standard manufactured product within guided limits. This customization process is suitable for small companies and those with no experience in customization processes (Safar *et al.* 2018).

- **Collaborative customization process**: When using a collaborative customization process, the factory cooperates closely with clients to receive as many requirements as possible to manufacture their goods. The customer should provide details on various requirements such as functionality, color, and size for the manufacturer to understand their final product better. This customization category is helpful for customers who do not have a good idea of the ending product (Tieng *et al.* 2017).

- **Transparent customization process**: A transparent customization process is about pre-manufacturing individualized products for clients without knowing that the final products have been customized (Safar *et al.* 2018).

- **Cosmetic customization process**: In a cosmetic customization process, a manufacturing company produces several standard goods that they can modify to present in various formats or forms depending on the customer (Safar *et al.* 2018).

Tieng et al. (2017) advise that the utilization and the integration of the appropriate IT system in factories will ensure the successful application of customization processes within factories. The information system needs to be robust, responsive, and stable to handle, in real-time, the considerable amount of data transferred from customers to the factory and vice versa. It implies that the IT systems run on an effective network communication system with high availability and low communication latency.

Some of product and MC benefits can be summarized as follows (Safar *et al.* 2018):

- An increase in customers' satisfaction and customers' value.

- An increase in the production system's flexibility by allowing the production of various products formats

- A better positioning of the organization in the competitive market with more customer recommendations and references.

- A limitation of material waste since the company produces based on customers' requirements and is more likely to sell the manufactured product than to stock them for a long time.

- An increase in a company product range offering with lower production cost. It becomes easier for the factory to derive several products from the initial ones.

- An improvement in companies' cash flow system because of an easier production selling procedure.

2.4.2 Advanced customization approaches for Industry 4.0 systems in manufacturing

The increasing competition in the manufacturing industry calls for more flexible and intelligent manufacturing approaches (Wan *et al.* 2016). Product and MC are customer-based approaches intended to increase the manufacturing system's flexibility and customers' value. The technological advances of the current trend of automation, I40, has enabled the development of several personalized customization manufacturing systems (PCMS) with the opportunity of having more intelligence (Wang, Wan, Zhang *et al.* 2016) and more advancement in production processes. The integration of concepts such as CPS (Zhang *et al.* 2015 and Wan *et al*. 2013) suggest several benefits for PCMS. (Wan *et al.* 2016) incorporated mobile services and cloud computing technologies to create a PCMS platform structure for industrial manufacturing systems. They contributed to building an effective customization system with improved interaction between suppliers, manufacturers, and customers.

Similarly, Wang, Wan, Li et al. (2016) combined industrial wireless networks technologies and cloud platforms in physical components like conveyor belts in production lines to design an intelligent manufacturing production system. Regarding product and MC, the intelligent manufacturing production system offered more visibility of the product manufacturing through its journey on production lines accessible to all production stakeholders. Leong & Koshijima (2018) integrated the IoT concept in MC processes to bring advantages such as Internet connection through the production process. The concept of IoT coupled with cloud computing allowed them to realize real-time monitoring of manufactured goods with another benefit of accomplishing advanced analytics on IoT devices. For Wan et al. (2016), integrating cloud computing in an organization's product and MC processes facilitates effective information exchange between production, sales, and management. It also lessens risks of communication latencies and data losses between production, sales, and organization management. Modrak et al. (2015) designed a methodical customization framework to create as many product variants as possible before the manufacturing process commences. This feature reduces work and pressure when the factory receives actual customers' personalized requirements because the previous variants will cater to at least some of their requirements. To optimize existing production lines, Wang, Wan, Imran et al. (2016) proposed the creation of a multi-parts processing line that is flexible enough to produce and process different product variants on the same production line. Their system receives various commands based on the product variant to manufacture and requires a calibration accordingly. Zhao et al. (2019) suggest introducing

more AI techniques such as ML in improving manufacturing processes like product and MC. AI would include its benefits of processing a massive amount of data, creating intelligent management services, and maintaining computing assets to build more powerful PCMS.

2.4.3 Product and mass customization for SMEs

As per Kiangala & Wang (2018), SMEs in the manufacturing industry face a slow adoption of advanced manufacturing processes like product and MC because of the absence of practical and affordable advanced customized applications that meet their environment's needs. Embracing new technological methods always implies investing in additional R & R&D services to decide on the appropriate solution and clearly understand the implications of new technologies in an organization's structure and production process. SMEs often have limited R&D resources or cannot afford any.

Several studies were conducted to guide SMEs on the embracement of product and MC in their production processes. Ismail et al. (2007) proposed a measuring tool for SMEs to establish the most appropriate customization possibilities their system could handle based on their product knowledge. Their tool is handy during the product design phase. Stojanova et al. (2012) designed another tool close to Ismail et al. (2007), represented as a graphical theoretical model for manufacturing SMEs to generate various product variants before the actual production begins. This tool is a design aid. Antonelli et al. (2007) suggests two significant steps for adding new customizable products in manufacturing SMEs' production lines. During the first step, the factory should highlight the additions and benefits of the new products when comparing them to the existing products offered by the factory. During the second step, the manufacturing SMEs will plan to integrate new product patterns and amendments discovered during the first step in the actual production processes. Another study done by Boer et al. (2018) reported various strategies for SMEs' successful implementation of MC. A few of these strategies are active communication and cooperation between suppliers and manufacturing plants, integrating the product and MC principle in the business culture, and introducing new flexible production processes. Brunoe & Nielsen (2016) focused on discovering the negative consequences of product and MC adoption by SMEs. They assessed how having multiple customizable product variants could affect manufacturing SMEs and whether MC would be profitable.

From the above researches, we can conclude the need to produce more practical solutions and MC techniques for manufacturing SMEs. Not only adoption guidance but also practical technical procedures in implementing the customization process within factories.

2.5 Ethernet and network switches

Implementing I40 approaches in factories promotes the utilization of robust and effective communication networks for a flawless transmission of data between various factory entities. I40 communication networks are moving away from Fieldbus communication protocols to Ethernet communication protocols that are more advantageous in transmission speed, ease of interconnection and integration, and standardization.

Ethernet is a robust networking communication protocol located at the data link layer of the Open Systems Interconnection (OSI) communication model to interconnect two or more endpoints (Huynh *et al.* 2011). The Institute of Electrical and Electronics Engineers (IEEE) developed the Ethernet protocol under the code IEEE802.3. Ethernet has been quite successful for communications between local area networks (LANs) devices in the office environment since the 1970s. From the year 2000, various automation companies created an optimized version of the Ethernet protocol to meet the demanding requirements of the industrial automation section, such as real-time communication responses and more determinism (Prinz *et al.* 2018). Most of the modified Ethernet optimized protocols were not standard. They are proprietary (vendor-dependent) and do not favor interoperability between them. Although designed from the same base protocol, Ethernet, these protocols were optimized with divergent features (Bruckner *et al.*, 2019).

Devices in an Ethernet network communicate via the exchange of "Ethernet frames" containing the information. We present in Figures 2-8 and 2-9 graphical schematics of an Ethernet version 2(v2) and an Ethernet IEEE 802.3. These are two Ethernet formats designed by the IEEE and that have evolved with time. Each frame representation contains various segments responsible for accomplishing different tasks to ensure good communication from the source to the destination. The major segments of an Ethernet frame are:

- **The preamble**: The preamble is the first segment of the Ethernet frame. The primary preamble function is to synchronize the Ethernet frame receiver. It is often presented as a sequence of logical ‘0s’ and ‘1s’ in 7 bytes.

- **The Start of Frame Delimiter (SFD)**: The SFD is the second frame segment after the preamble. The SFD denotes the commencement of the actual Ethernet frame. It comprises a series of logical‘1s’ and ‘0s’ forming 1 byte.

- **The communicating devices source and the destination addresses:** The source and destination address segment contain the media access control (MAC) address of the two communication endpoints, also known as the physical network interface card (NIC) address**.**

- **The type:** The type segment saves information about the transport protocol used to transmit frames: IP or User Datagram Protocol (UDP). It is worth mentioning that the type segment is only seen in Ethernet V2.0 frames (not in IEEE 802.3 frames types).

- **The length**: The length segment is another optional segment only present in IEEE 802.3 frames (not in Ethernet v2.0 frames). The length segment contains the transmitted frame data field size.

- **The protocol data unit (PDU)**: The PDU segments carry the actual data (information) exchanged between the two endpoints communicating.

- **The Frame Checking Sequence (FCS):** The FCS segment contains a checksum to assess frames carrying errors during the transmission.

- **The Inter Frame Gap (IFG)**: The IFG is a segment 12 bytes long, determining the minimum space required between the transmissions of two consecutive frames.

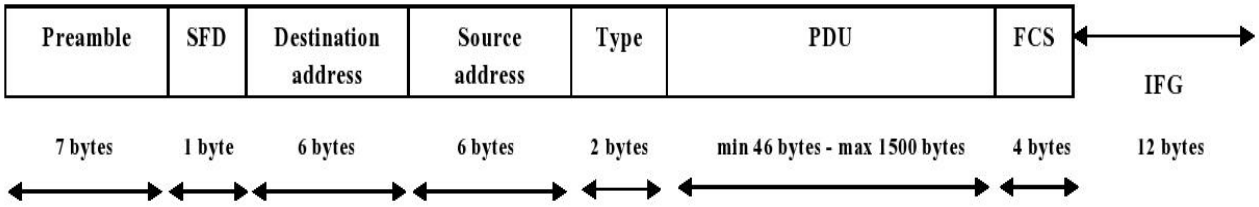

Figure 2-8: Ethernet v2.0 frame segments

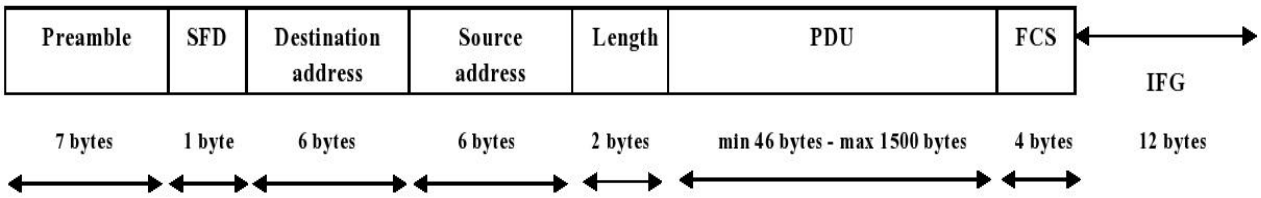

Figure 2-9: Ethernet IEEE 802.3 frame segments

The Ethernet protocol utilizes layer 2 (from the OSI model) devices called network switches to forward frames from one end-device to another. A switch has several functions: forwarding Ethernet frames, analyzing them, learning source and destination MAC addresses, and detecting possible faulty frames through the FCS segment. The switch saves the learned MAC addresses of devices connected to each port in an address or forwarding table. Once the switch saves a device MAC address, it does not need to forward frames to all its ports (avoiding reducing the switch bandwidth) anymore but only to the desired port matching the saved MAC address. It is the main difference between a layer 2 switch and an entry-level hub device. The process of sending frames to all ports is called flooding and only happens when a switch has not yet learned a node MAC address (for example, for a new switch connected to a port). We display in Figures 2-10 and 2-11 frame transmission illustrations. Switches transmit frames sequentially, from ingress or input ports to egress or output ports (when transmitting to the same output ports). For different output ports, switches can transmit frames in parallel. Switches utilize internal memories to store exceeding frames while a port is busy during a transmission (Lo Bello & Steiner 2019).

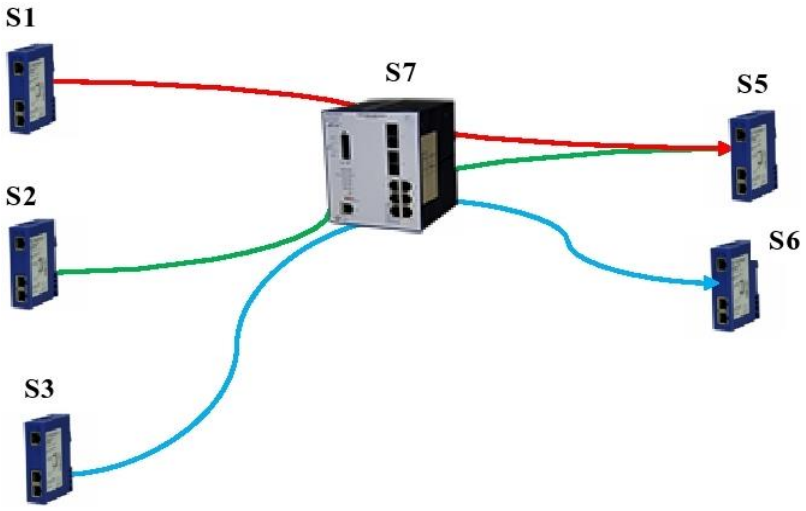

Figure 2-10: Switch frames transmissions (1)

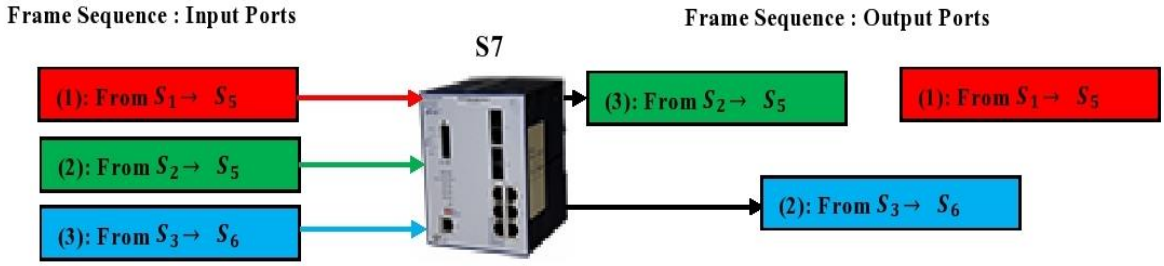

Figure 2-11: Switch frames transmissions (2)

From Figure 2-10, we consider three frames transmission paths ($\boldsymbol{p_1}$, $\boldsymbol{p_2}$, and $\boldsymbol{p_3}$) from the three devices $\boldsymbol{S_1}$, $\boldsymbol{S_2}$, and $\boldsymbol{S_3}$ (source devices) to the two devices $\boldsymbol{S_5}$ and $\boldsymbol{S_6}$ (destination devices) via the middle switch $\boldsymbol{S_7}$. We can define the data paths transmission for frames from source to destination with expressions in (2.1), (2.2), and (2.3).

$$p_1 = S_1 \rightarrow S_7, S_7 \rightarrow S_5 \tag{2.1}$$

$$p_2 = S_2 \rightarrow S_7, S_7 \rightarrow S_5 \tag{2.2}$$

$$p_3 = S_3 \rightarrow S_7, S_7 \rightarrow S_6 \tag{2.3}$$

Frames transmission and processing from one switch to another does happen instantaneously. Switches experience several types of delays when forwarding Ethernet frames. All these delays contribute to slowing down communication responses between network devices. Some of the well-known Ethernet network delays have been described by Lin et al. (2019) and summarized as follows:

- **The minimum transmission frame delay**: The minimum transmission frame delay is the slightest delay a frame goes through in a network when travelling from one switch to another without being stored in a switch memory. (2.4) is a simple mathematical expression of the minimum frame transmission delay.

$$\delta_{cmin} = \delta_{src} + \delta_{dst} + 2(\delta_{frm} + \delta_{cbl}) \tag{2.4}$$

where $\boldsymbol{\delta_{src}}$ represents the delay a frame experiences at the source switch due to processing tasks, $\boldsymbol{\delta_{dst}}$ is the delay a frame experiences at the destination switch due to processing tasks; $\boldsymbol{\delta_{frm}}$ represents the delay caused by the frame forwarding itself. A mathematical definition of $\boldsymbol{\delta_{frm}}$ is displayed in (2.5). $\boldsymbol{\delta_{cbl}}$ is the delay caused by the frame physical medium through which the frame travels. The frame physical medium could be a copper or a fiber cable depending on the network installation.

Lin et al. (2019) state that the worst frame transmission delay in a cable is assumed to be at more or less $\frac{2}{3}$ the speed of light. The approximated speed of light is 3,000,000,000 m/s. When assuming that the cable sizes connecting the source (transmission) node to the destination (recipient) node is equal, we can calculate the worst frame cable delay in a distance of 20 meters as:

$$\delta_{cbl} = \left(\frac{2}{3} \times 3{,}000{,}000{,}000\right) \div 20 = 0.1\mu s$$

$$\delta_{frm} = \frac{\eta}{x} \tag{2.5}$$

where $\boldsymbol{\eta}$ represents the transmitted frame dimension expressed in bits and $\boldsymbol{x}$ represents the transmission rate in bits per seconds.

- **The frame communication delay when waiting in a switch memory:** In most busy networks, ports receive several frames at the same time for transmission. In this case, frames forwarding is not executed immediately from the source to the destination but stored in the source memory until the transmission port is free. Frames are stored in queues based on the priority level. (2.6) is the mathematical expression for the frame communication delay stored in switch memory before transmission.

$$\delta_c = \delta_{cmin} + \delta_{mry} \tag{2.6}$$

where $\boldsymbol{\delta_{mry}}$ represents the queuing delay or the delay frames wait in a switch memory before transmission. We display in (2.7) a mathematical expression for the queuing delay.

$$\delta_{mry} = \sum_{n=1}^{F_m} \mathrm{IFG} + \max\,(\mathrm{S_n} + \mathrm{S_{hd}})\frac{1}{x} \tag{2.7}$$

where $\boldsymbol{F_m}$ represents the total count of queuing frames in the switch memory. The IFG represents the size the inter frame gap (12 bytes), $\mathbf{S_n}$ represents the dimension of the nth data frame in the switch memory queue, $\mathbf{S_{hd}}$ is the size of frame overhead, and $\boldsymbol{x}$ is the frame forwarding rate in bits per seconds.

The frame delays that we described earlier, and their calculation only applies to switches applying the store and forward switching method when processing and transmitting frames. This method consists of analyzing the entire frame content before the transmission. Few other switching methods such as cut-through forwards frame without examining and analyzing its whole content.

Lin et al. (2017) defined a summarized version of the forwarding delay from a source node point Y to a destination node point Z connected in a Ethernet communication network having a total number of switches *k*. Its mathematical expression is presented in (2.8).

$$\delta_{YZ} = \delta_{t1} + \sum_{p=1}^{k} (\delta_{\mathrm{lnk_p}}) + \sum_{p=1}^{k-1} (\delta_{\mathrm{swt_p}}) \tag{2.8}$$

where $\boldsymbol{\delta_{t1}}$ represents the transmission delay to forward all frames from a switch into a link, $\boldsymbol{\delta_{lnk}}$ represents the frame transmission delay with regards to its data rate, and $\boldsymbol{\delta_{swt}}$ is the frame processing delay from a source switch input port to its output port.

2.6 Business models for innovative organizations

2.6.1 Business model

As reported by Kaplan & Winby (2007), most businesses intending to venture into delivering sustainable, innovative products and solutions need to consider redesigning their business model to one that promotes an innovative culture and business expansion. The paradigm of I40 offers several innovative technologies for businesses to grow their flexibility and production. It implies that the successful adoption of I40 depends not only on technological upgrades for the production systems but also on appropriate business models for innovative techniques (Jovanovski *et al.*, 2019).

A model refers to a simplified representation of a complex system or structure (Stähler 2002). When reflecting on a business model, researchers suggested various definitions to understand the concept better. Osterwalder & Pigneur (2010) define a business model as an instrument through which entrepreneurs and business owners can define, capture, and visualize their fundamental business values regarding services and the production of goods they supply to their customers. For Teece (2010), a business model is an architecture or a design that reflects an organization's way of creating value, capturing production (service delivery) mechanisms, and delivering products or services to clients. Osterwalder & Pigneur (2010) describe a business model as an illustration of the rational approaches a business utilizes to create, capture and develop value. Although authors present the definitions of a business model in various

formats, there are common keywords that carry the meaning of what a business model is: "the creation of value" and the "capture of value." The creation of value means benefits that businesses add to customers. The capture of value refers to how businesses represent all thriving procedures they use internally to produce value for their clients (Gudiksen et al. 2014). As per Timmers (1998), a prevalent explanation of the business model concept includes all service providers or production stakeholders, their tasks in the production process, source of income, and the value created. Zott & Amit (2010) mention that a business model is a group of several activities (that depends on each other) engaging human beings, physical components, and capital resources intending to reach companies' objectives. Elvesæter et al. (2010) research proposes a more straightforward definition of a business model as a description of valuable processes utilized to make money.

From the above definitions, a business model's purpose is to identify internal processes implemented by companies to offer customer value and capturing these processes for the continuity of their activities (Johnson 2010). Osterwalder & Pigneur (2010) propose few elementary questions companies should consider answering when designing their business models:

- Who are clients or targeted clients are?
- Which services or products are we offering to clients?
- What are the internal processes in place to produce profitable and sustainable value?

Glova et al. (2014) suggest two different business model perspectives: a process business model and a value business model. In a process business model, an organization highlights process architectures helpful to reach desired objectives. In a value business model, the company presents essential details on all value creation stakeholders. The value business model can be effectively utilized as a strategic method for companies to assess their current market positioning and readiness to embark on new market opportunities.

2.6.2 Business model for innovation

Companies in the industrial manufacturing sector are presently facing an increase in competition. They are also witnessing new innovative technologies that they need to consider implementing to remain relevant to market standards and demands. Some of them embrace

globalization where they require more competitive prices, shorter product manufacturing time, and less time-to-market. The demanding requirements of innovative technologies and globalization arouse the need to design new business models that promote innovation and improvement, which current traditional business models fail to achieve (Hecklau et al. 2016). Traditional business models are primarily reflectors of business production processes or simply of how organizations make money. They do not often tackle several aspects such as creating a collaborative environment to favor innovation, R&D practices for new technologies, improvement of current product quality and processes, reducing production costs, and creating significant and sustainable value for customers. These practices could help companies adopt innovative technologies and become more competitive in their market sectors (Gupta & Sharma 2003).

Referring to the booming experiences of few companies such as Airbnb and Uber, Weking et al. (2018) reported that business models could be excellent drivers of innovation and new technologies within companies when effectively exploited. According to various literature, Weking et al. 2018; Foss & Saebi 2017 and Zott et al. 2011, integrating innovation in companies' business models constitute an excellent foundation of revolution in businesses, mainly when associated with a process, product, and service innovation. As per Baden-Fuller & Haefliger (2013), even though the business model concept has nothing to do with technological approaches, it represents a strong connection between the application of technology in businesses and their performance. When adopting new technology trends such as I40, business models should become more innovative by actively incorporating technology in different business levels. The concept of I40 highlights the implementation of methods like cloud computing, CPS, AI, and IoT devices through which companies can acquire more intelligence by getting more insights from components and analysing them. Innovative business models for I40 technologies include designing efficient algorithms and solutions that bring more value to manufacturing factories.

In summary, we can state that integrating innovation into business models is about making relevant changes to significant components of an organization's traditional business model and modifying its architecture to match new sustainable requirements (Foss & Saebi 2017).

2.6.3 Some reference business models

Several organisations and research design or upgrade their business model ideas based on some popular reference models or guides that they adapt to their realities (Osterwalder & Pigneur 2010 and Safar *et al.* 2018). Some reference business models are:

- The traditional business model: The traditional business model is a model focusing on the effectiveness of executive power to drive the whole organisation. In this type of model, the success or failure of the company depends highly on the manager or founder's efficacity in the business. This model is also often referred to as a hierarchical business model (Bass 2021). It displays the organisation stakeholders and employees in terms of power, starting on top of the pyramid (hierarchy) with the founder, followed by experienced managers, and ending with employees having straightforward tasks with less impact on business operations. This model is simple to adopt for small businesses and start-ups because it allows the founder(s), at the early stage of the business, to be in complete control of their vision and performance without relying too much on employees. In traditional business models, employees feel less involved in their organisation advancement or growth as they solely rely on their managers' decisions.

- The business model canvas: The business model canvas is a more innovative model designed to encourage the embracing of new technological concepts within companies such as Industry 4.0 (Safar *et al.* 2018). Its structure provides four fundamental elements from which organisations can focus to shift from a traditional business model to a more innovative one. From its essential elements, the business model canvas can be a valuable template for companies to understand their business goal by asking themselves fundamental questions. These four key elements are:

    - The value and/or product proposition: This section allows the organisation to look closely into their market positioning and competitivity.
    - The customer relationships: This segment involves distribution channels and other important tools to grow customers relationships.
    - The key partners: This section highlights key activities and company resources.

- The Finance: This segment focuses on the company revenue stream as well as the cost structure.

The business model canvas does not give practical recommendations in adopting innovative technologies but presents several fields worth considering when adopting new technological trends.

- The STOF (Service – Technology – Organization - Finance) business model: The STOF business model is another innovative model for organizations. Like the business model canvas, it focuses on four main elements to embark on new technological trends. However, researchers consider the STOF business model simpler in structure and easier to embrace by small companies (Safar *et al.* 2018). These elements are:

  - The service: this segment provides different ways and approaches to create customer value.
  - The technology: the technology segment focuses on new technological components and infrastructure needed for innovation.
  - The organization: this section focuses on all external and internal organizational activities that produce value.
  - The Finance: This segment focuses on the company revenue stream as well as the cost structure.

2.7 Chapter Summary

In this chapter, we reviewed various literature and background on essential concepts and technologies covered in this research. The background and literature review enable us to gain sufficient knowledge to reach our research objectives. The concepts covered are I40, SMEs, I40 challenges and opportunities for SMEs, AI, ML, IoT, predictive maintenance (PM), product and mass customization (MC), Ethernet and network switches, and business models that promote innovation.

Chapter 3 : A CONCEPTUAL BUSINESS MODEL GUIDE FOR INNOVATIVE SMES

As described in the previous background and literature review chapter, the sustainable implementation of innovative technological trends such as AI methods within organizations depends on the business model in place. This chapter is about setting up a conceptual business model guide that can be adapted and utilized as a foundation for SMEs intending to embrace innovative techniques within their companies.

3.1 A conceptual business model for innovative solutions under I40

The successful implementation of advanced and intelligent techniques or tools to achieve an innovative production system, as suggested by the I40 paradigm, depends on an organization's business model. A company's business model under the I40 should promote an innovative environment where all production stakeholders collaborate to reach the intended goal, a product, or a service (Kaplan & Winby 2007).

In the previous chapter, under subsections 2.6.3, we went through an overview of three existing reference business models and their fundamental elements: the traditional business model, the canvas, and the STOF business model, for innovative technologies. Our proposed innovative business model moves away from the traditional business models (Gupta & Sharma 2003) structure that is not very collaborative because of its pyramidal (hierarchical) structure. The top management gives instructions to employees and collects results at the end of the process. Establishing innovative solutions in such working models is quite tricky to perpetuate. Learning from the canvas and STOF innovative business model layout, our model goes a step further by providing a more detailed structure of the interaction between all organization stakeholders, practical guidance, and recommendations regarding several types of innovative technologies and valuable tools to consider within the organization. Compared to canvas and STOF, our proposed business model offers a more thorough and practical guide for SMEs to adopt innovative technological trends.

We propose a conceptual business model for SMEs under the I40, in Figure 3-1, as a foundation for implementing AI techniques that improve production processes.

Our proposed business model lies on the following aspect:

- The organization stakeholders
- The innovation areas
- The training
- The operations
- The customers feedback

3.1 The organization stakeholders

A business model that encourages innovation considers the participation of every production's stakeholder: **management, employees**, and **customers**. From our conceptual business model in Figure 3-1, we group these three stakeholders in the same category, visible by the same group colour (blue), and we give them equal responsibilities in successfully running a business. In an I40 environment, **customers** have a more crucial role as they are included in the operations via techniques such as products or mass customization. It becomes compulsory for an organization to develop a strong relationship with their customers to know them better, propose improved solutions, understand their needs, and find solutions to their existing problems. Providing such services is a good step for a company to ensure its competitive positioning in the market.

In our proposed business model, **management** and **employees** frequently collaborate to understand their current operations and structure better. Each party brings its expertise and present knowledge of the system to assess, amend or improve the overall operations. Unlike in pyramidal business models, **employees** are as important as management or customers. They are the pillar of the operations as they work on the system daily, and it is easier for them to pinpoint its feats and flaws. **Management** has the mission to create a climate that encourages sharing and collaborative sessions; they should also apply mechanisms that extract strategic information from employees' and customers' inputs.

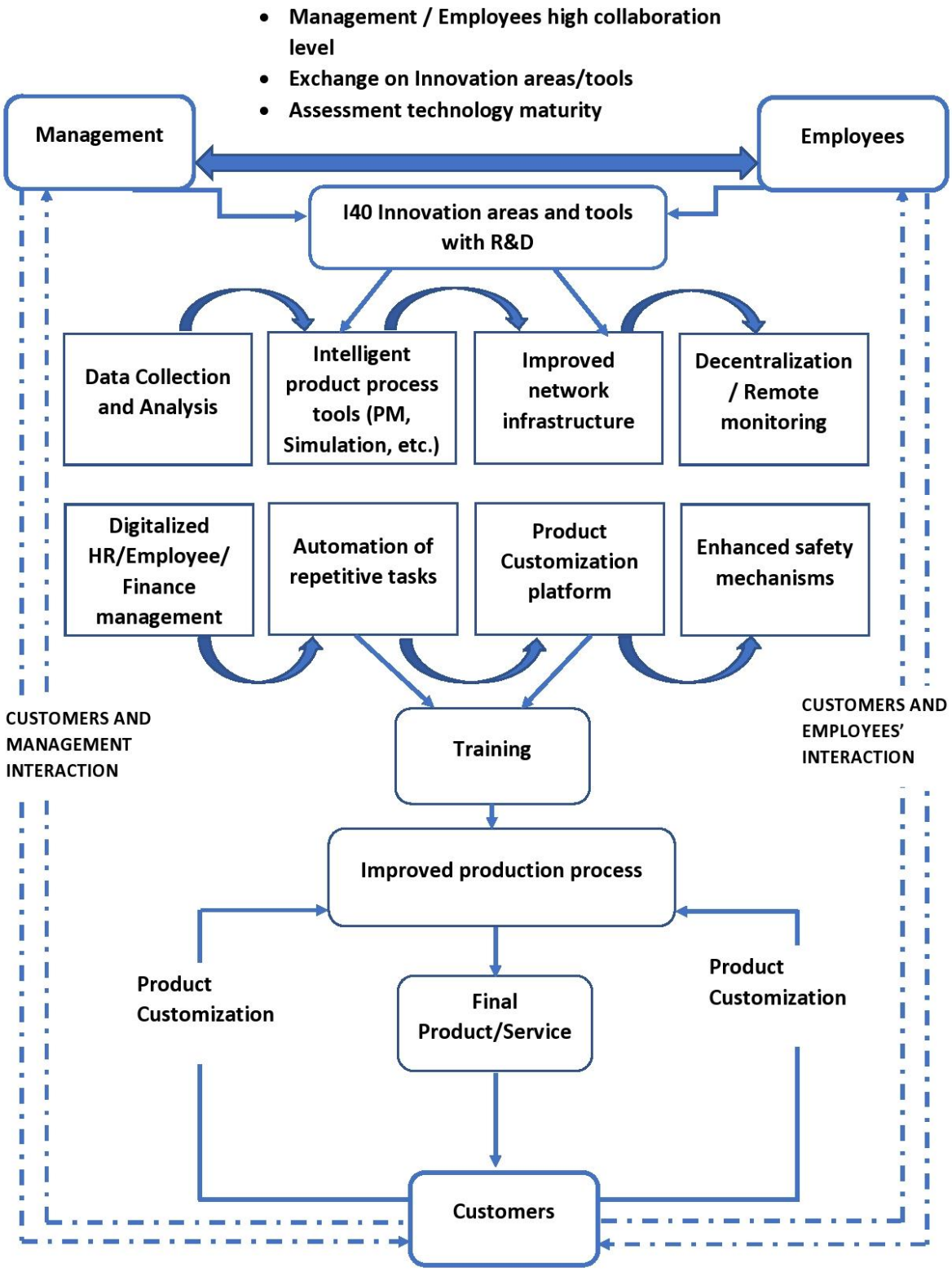

Figure 3-1: Conceptual business model for innovative SMEs

3.2 The innovation areas

There are several innovation areas for an I40 environment. Most of them originate from new and advanced technologies. Some of these new technologies are AI, ML, AR, Big data, Edge computing, Digital twin, and much more. In our business model, we choose eight innovation areas SMEs could consider in an I40 environment.

3.2.1 Data collection and analysis

Data is the fuel for the implementation of new technologies. It helps gather important information about the system, and it allows the creation of models that better represent the system. Collecting and analyzing data at strategic areas of operations (product composition, customer preferences, machine parameters, and system configuration) is crucial in accomplishing improvement.

3.2.2 Using intelligent tools to improve production processes

There are various intelligent tools and procedures that can be added to the production process to create more value. Some of them are:

- **The predictive maintenance:** that consists of detecting system impairments before they occur or before they convert into fatalities.
- **The RT monitoring:** that reports the system operations as they appear in real-time. It is useful to generate alarms in case of faults, to stop the system in case of danger and to view the system responses based on different conditions.
- **The Simulation**: that allows the testing of a system with factors and elements close to reality prior its actual physical implementation. Simulation reduces the risks and costs of wasting physical resources especially when they involve several changes and amendments.

3.2.3 Improved network infrastructure

A network infrastructure facilitates information sharing and communication between different sections, machines, and production stakeholders in a manufacturing environment or a plant. Depending on the type of physical media installed (fibre, copper, or wireless), machines located farther apart or remotely can still exchange data and operate. A communication network comprises several devices (network switches) connected in a specific topology that carry information from source to destination. An unstable network jeopardizes the whole operation of the plant and affects its productivity. In an I40 environment, it becomes imperative to implement a solid and stable network that ensures deterministic communication (with minor delays possible in delivering information) and caters for unforeseen failures. Some of the suggested ways to improve a network infrastructure are:

- Choosing an efficient network topology that allows smooth communication with few delays.

- Implementing redundancy protocols to have backup routes in case of failures.
- Installing network management software (NMS) to have visibility of all network equipment and instantaneously report errors.

3.2.4 Decentralized and remote monitoring of production processes

Two of the most popular characteristics of an I40 environment is realizing a decentralized and remote monitoring system. Unlike in a traditional manufacturing environment where all production equipment depends on a sole controller to function, an I40 environment encourages a decentralized system in which production machines are given more intelligence to make decisions on their own without relying on the central controller. This feature reduces waiting time in the decision-making and is very handy in case of breakdown at the central controller's side. In other words, a decentralized system offers redundancy to the production brain structure.

The remote monitoring task adds flexibility to the system allowing operations to be accessible from different areas without a physical connection to production machines. Remote monitoring also includes error reporting. It becomes easier for supervisors and management to remain updated about operations and productions in real-time without depending on operators' reports.

3.2.5 Digitalization of HR, employee management and linking finances to actual production

Technologies such as Radio Frequency Identification (RFIDs) and biometrics are utilized in I40 to track and trace production levels, products, and employees' actions. The manufactured product can be remotely tracked at any stage of the production and its manufacturing time determined (the relationship to the final product or service delivered to clients), and customers updated. HR management can generate statistical reports linking employees' tasks to production stages and overall cost by these intelligent tracking methods.

3.2.6 Automation of repetitive tasks to redirect production workforce

In an I40 environment, employees should move away from robotic and repetitive tasks requiring little intelligence to more strategic duties to improve current operations continuously. With intelligent computer software applications and algorithms in AI and ML, certain production chores like parameter configuration, the decision on an overtime production run, production reporting, and much more can be taken care of by the system itself. Therefore, freeing operators to be trained on high-level production duties like data analysis and production planning.

3.2.7 Develop a customer-oriented product system through product customization

Customers are the backbone of any business. They are why an organization exists, and the organization always must ensure customers' satisfaction. Standard customer service tips such as delivering products on time, updating customers on production stages or any delays, providing after-sales support, and offering cost-effective solutions, are initial steps in keeping pleased customers and perhaps attracting new ones. In addition to the above mention tips, innovative production systems consider customers' preferences. Innovative manufacturing companies no longer produce standard products but allow clients to amend their composition as per their needs before manufacturing them. The operation is called product customization.

3.2.8 Enhanced safety procedures for operators and intelligent machines (robots)

A safe working environment is a good foundation for a productive organization. The advent of I40 promises an increase in intelligent machines such as robots to the production workforce and for more interaction between human operators and production robots. A room for enhanced safety procedures through advanced technologies is crucial to avoid frequent incidents at the workplace. Robots, like humans, should undergo proper safety induction to learn appropriate responses to carry out in emergencies and reduce the risks of fatalities.

3.3 Training

It is compulsory for management and employees to regularly undergo proper training regarding the innovative areas they would like to implement. It will ensure a better understanding of both parties and a fast implementation of these innovation areas for the organization.

3.4 The operations

The manufacturing operations in an I40 environment are closely tracked, regularly quantified, regularly assessed, remotely accessible, and influenced by customers' preferences through product or mass customization. Usually, a customer portal with a user interface linked to the production interface enables customers to send their preferences to the factory. An organization should take all these factors into account when designing its production structures and operations.

3.5 The customers feedback

Adopting innovation strategies to improve an organization's production is not a once-off activity to achieve the desired goal but a continuous task to improve a current system. It should ultimately become the organization's culture to promote innovative activities as suggested by our conceptual business model. One of the best ways to improve a system's operation is to recurrently receive customer feedback through interactions with management and employees and brainstorm on how to gain the most from those feedback during management and employee sharing sessions.

From our business model, we encourage management and employees to get acquainted with innovation areas that would benefit their organization. Having a good understanding of their operations through frequent interactions, they can agree on which innovative areas would be advantageous for their business. However, the apprehension of advanced techniques requires specialized know-how that most SMEs do not have. Hence, we suggest another collaboration between SMEs (management and employees) and academia or R&D projects focusing on I40 innovation techniques to guide them in selecting the best innovative areas for their operations. Academic works and researches, such as this thesis, conducted to stimulate the adoption of advanced and intelligent techniques for SMEs are excellent pilots to instruct management and employees.

3.6 Chapter Summary

In this chapter, we laid down the organizational foundation to guide SMEs in considering adopting advanced technological trends (for example, adopting I40, the current trend of automation. Embracing innovative techniques depends highly on the type of business model in place for an organization. Our conceptual business model outlines different business segments and areas that SMEs should consider for a successful move into innovative technologies. We highlighted the main differences between our proposed innovative business model and the reference business models discussed in chapter 2. We developed the conceptual, innovative business model as the foundation for designing and implementing practical, innovative techniques.

Chapter 4 : DEVELOPED AI TECHNIQUES AND INNOVATIVE METHODS OVERVIEW

This chapter presents an overview of the developed AI techniques and innovation areas covered in this research. These areas allow manufacturing SMEs to improve their production processes in an I40 environment. We highlight these areas from the conceptual business model produced in Chapter 3. We also provide details on "data collection and analysis" that is the foundation to modelling most of our AI techniques.

4.1 Innovation areas covered

Through this research, we aimed to develop and optimize AI techniques to improve the operations of manufacturing SMEs. These techniques cover some of the innovation areas SMEs can adopt in an I40 environment to optimize their productivity. Figure 4-1 presents, in green coloured cells, the innovation areas and tools our study addresses from the conceptual business model in Figure 3-1.

We cover the following innovation segments:

- **Data collection and analysis**: Under this section, we explored knowledge-based data utilized in the design of our AI techniques (time-series data).

- **An intelligent PM framework using CNN algorithm**: In chapter 5, we describe the applicable theories behind the CNN algorithm, PReLU activation function, PCA, and time-series imaging exploited to design the intelligent PM framework. We design the system and evaluate results.

- **An improved network communication infrastructure**: In chapter 6, we discuss the theory behind some network communication theories such as TSN, edge computing, and network communication theologies utilized to develop our improved communication network prototypes. We design the system and evaluate results.

- **An automatic parameter configuration using ML techniques**: In chapter 7, we explore the theories behind DT and MLR ML algorithms to create regression models

implemented in the design of the automatic parameter configuration method for a SCADA system. We design the system and evaluate results.

- **An adaptive customization platform for clients' interaction with the production system**: In chapter 8, we analyse the theoretical principle of the XGBoost algorithm for the development of regression models and the RF algorithm for classification models. These two ML models are part of the core design of the adaptive customization platform. We design the system and evaluate results.

- **An enhanced safety response mechanism for AMR and operators using Q-learning and speech recognition:** In chapter 9, we go through theories on the RL algorithm, the Q-learning algorithm, and the speech recognition system that we utilize to produce our enhanced safety response mechanism. We design the system and evaluate results.

4.1.1 Data collection and analysis

An appropriate collection and analysis of system data are essential to build and design the desired AI techniques and advanced tools for manufacturing SMEs. We record and examine our data based on a knowledge-based system (Avram 2004). A knowledge-based system group makes available and organizes several factual information about a structure, a method, or a system that it implements as supporting materials to bring more intelligence to the system. It is a well-known technique to construct sophisticated AI models and intelligent algorithms that better express the meaning of a system. Figure 4-2 is a graphical representation of a simple knowledge-based system (Hopgood 2005). The system has three principal components: an evidence-based component, a deduction component, and an external communication interface. The evidence-based component records the previous essential data knowledge on the system that serves as the base to forming intelligent models. The deduction component is the brain of the system. It contains the control and the intelligence to build desired models. The interface component is the communication bridge between external users (to load knowledge-based data or to inquire about the intelligent constructed model) and the knowledge-based system. We present another more detailed knowledge-based system representation in Figure 4-3 for fault diagnosis as developed by Gao et al. (2015).

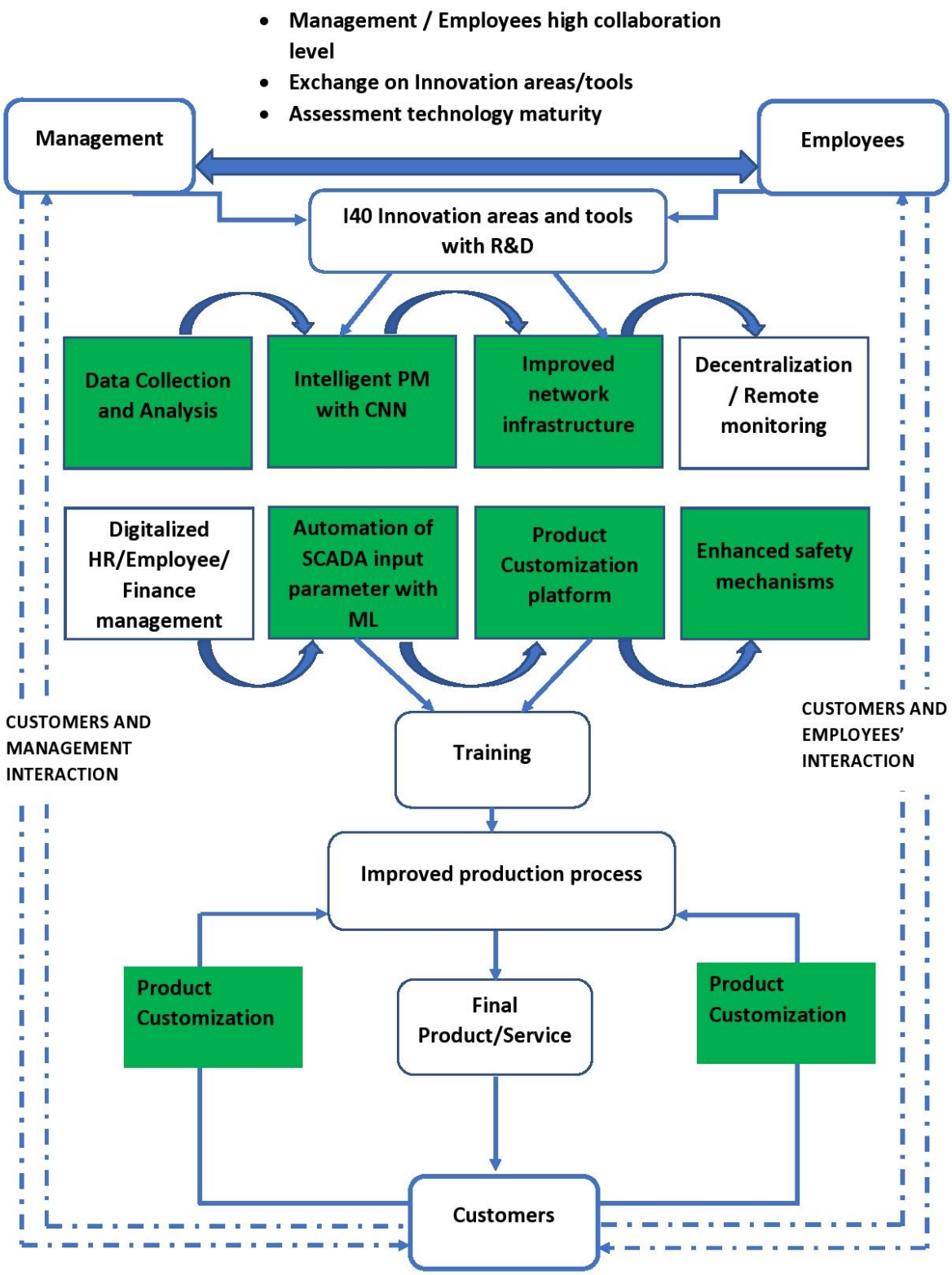

Figure 4-1: Conceptual innovative business model with innovation areas covered

This system can be a guide for fault diagnosis models built on knowledge-based data. In this research, we use time-series data as a knowledge-based data source for designing innovative

techniques for manufacturing SMEs. Through data analytics, a system gathers and assesses information from several sources. The collected data should be presented clearly to support business decision-making and organization operational activities (Lichtblau et al., 2015).

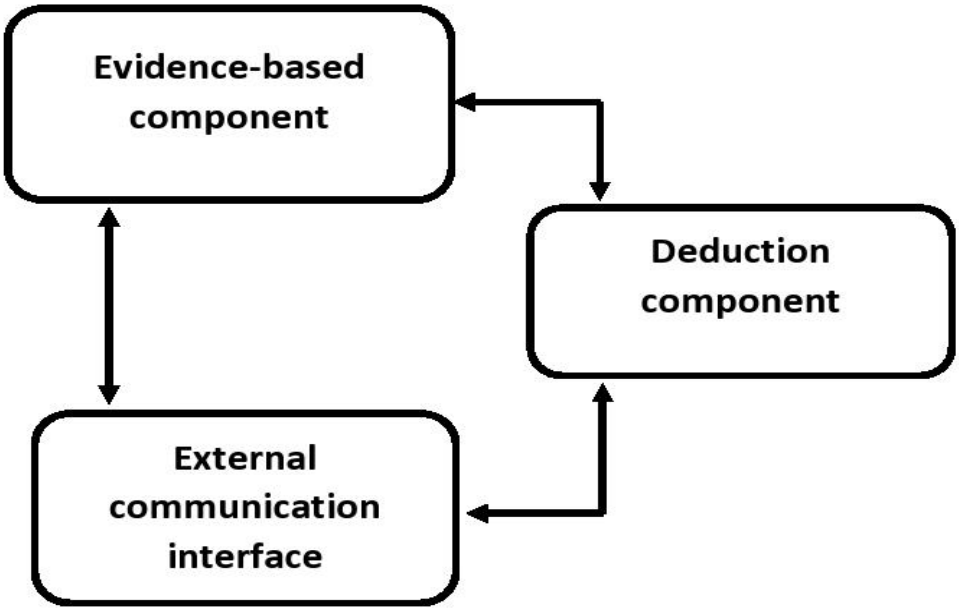

Figure 4-2: Basic knowledge-based system module (Hopgood 2005)

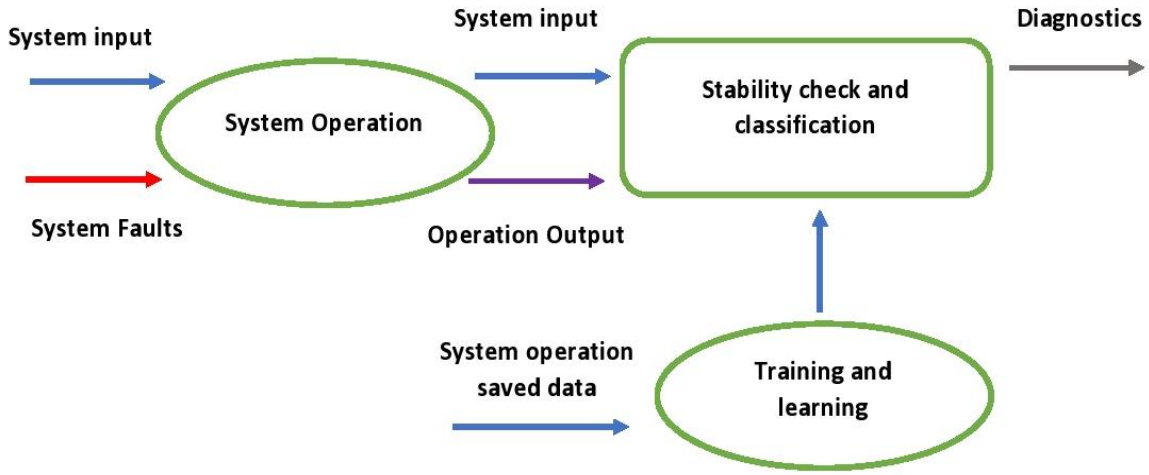

Figure 4-3: Fault-diagnosis knowledge-based system schematic (Gao et al. 2015).

4.1.1.1 Time-series data

Time-series data is a sequence of findings monitored and saved over successive intervals (Adhikari & Agrawal 2013). There are two elementary classes of time-series data: Multivariate Time Series (MTS) and Univariate Time Series (UTS). UTS contains a single observation (or a single variable) monitored over a specific period. Unlike UTS, MTS has more than one variable recorded over time (Sadouk 2018).

We present in (4.1) a mathematical expression of UTS data and in (4.2) the one for MTS.

$$B = [b_1, b_2, b_3, \cdots, b_n, \cdots b_t] \tag{4.1}$$

where $\mathrm{b_n} \in \mathbb{R}$, $\mathrm{t} \in \mathbb{N}$ and represents the dimension of the time series data.

$$D = [B_1, B_2, B_3, \cdots, B_i, \cdots B_m] \tag{4.2}$$

where $m \in \mathbb{N}$ and means the dimension of the MTS data. *m* also refers to the amount of UTS data in D. *i* is the position index for every UTS in the MTS, D. The UTS values in D are equal to the mathematical expression in (4.1). The expression of one UTS, B, a MTS, D, with several UTS can be presented in (4.3) as:

$$B_i = \left[b_{i(1)}, b_{i(2)}, b_{i(3)}, \cdots, b_{i(n)}, \cdots b_{i(t)}\right] \tag{4.3}$$

where $\mathrm{t} \in \mathbb{N}$ and represents the dimension of the UTS in the MTS (Box, G.E.P., et al. 2015 and Górecki & Łuczak 2015). *i* is the unique index position for every UTS, B in the MTS.

The expression of UTS, in (25), represents a mathematical vector and we can express the MTS one in (4.2) as a matrix since it is composed of several vectors. From (4.2), we assume the following parameters: an MTS, D with t = 4 and m = 4. We can express (4.2) in a matrix format in (4.4) as follows:

$$D = \begin{matrix} b_{11} & b_{21} & b_{31} & b_{41} \\ b_{12} & b_{22} & b_{32} & b_{42} \\ b_{13} & b_{23} & b_{33} & b_{43} \\ b_{14} & b_{24} & b_{34} & b_{44} \end{matrix} \tag{4.4}$$

We display in Figures 4-4 and 4-5 the graphs for UTS and MTS data.

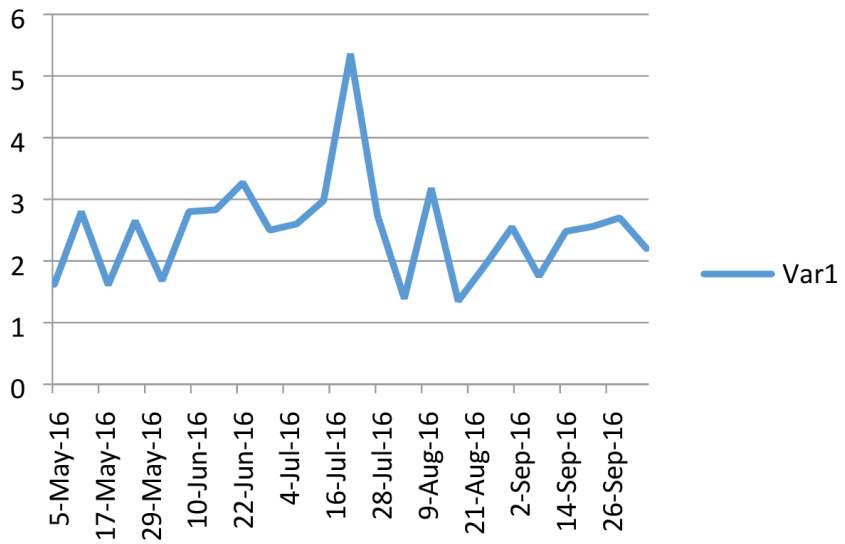

Figure 4-4: UTS graphical representation (Kiangala & Wang 2020 a)

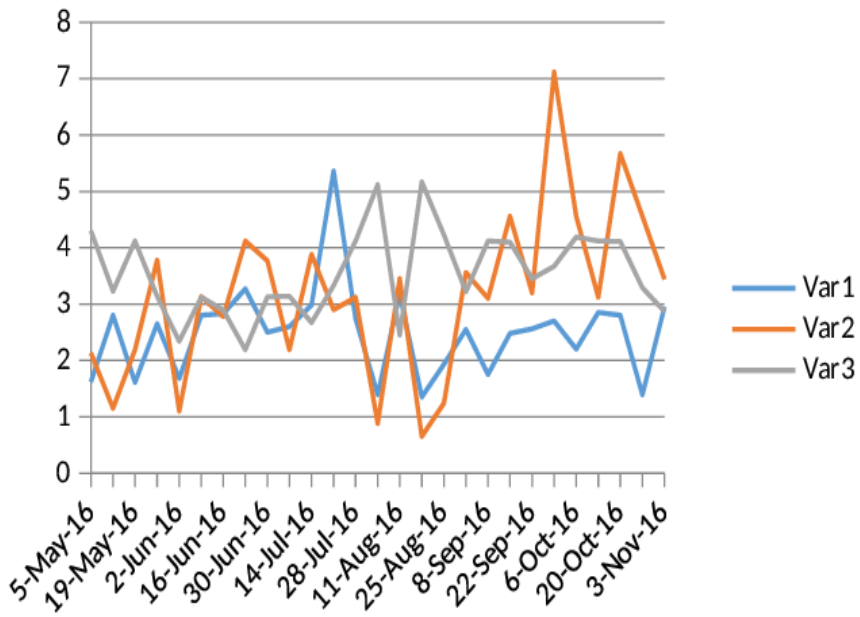

Figure 4-5: MTS graphical representation (Kiangala & Wang 2020 a)

From Figure 4-4, we can observe a UTS variable 'Var 1' recorded between 05 May 2016 and 26 September 2016. The values monitored in this variable are ranging from 0 to 6. From Figure 4-5, we monitor three variables: Var1, Var2, and Var3 that are part of an MTS dataset.

4.2 Chapter Summary

In this chapter, we introduced the innovation areas covered by our research from the conceptual business model in Figure 3-1. We presented the AI techniques and innovative methods designed and optimized for manufacturing SMEs in Figure 4-1 as follows: data collection and analysis, an intelligent PM framework using CNN algorithm, an improved network communication infrastructure, an automatic parameter configuration using ML techniques, an adaptive customization platform for clients' interaction with the production system, and an enhanced safety response mechanism for AMR and operators using Q-learning and speech recognition. We focused on describing the data collection and analysis area as the base to design most of our AI innovative techniques. We present and develop the remaining innovation areas in the following chapters.

Chapter 5 : AN INTELLIGENT PREDICTIVE MAINTENANCE (PM) FRAMEWORK FOR CONVEYOR MOTORS USING CNN

Equipment downtime and failures can cause significant losses for a production line. Detecting threats on a machine before they turn into breakdown reduces the risks of disasters and allows anticipating a maintenance process to fix issues. We propose an experimental PM structure design that aims to detect a conveyor motor functioning's impairment for early intervention on system's menaces and to decrease the rate of incorrect faults diagnosis. We carry out this task by implementing a CNN ML model that determines whether the abnormalities detected are harmful to the operation or not. We create an ML classification model by learning information from time-series data imaging (conveyor motors parameters observations) and by exploiting the advantages of CNN (Kiangala & Wang 2020 a).

5.1 CNN Theoretical overview

5.1.1 Convolutional Neural Network (CNN) and its limitations

CNN is a deep learning (a machine learning branch illustrated in Figures 2-2 and 2-3**)** algorithm intensively implemented in image classification scenarios. Its algorithm extracts significant spatial correlation of input data (images) and generates essential information attributes for pattern detection. Hubel and Wiesel developed the CNN approach based on animals' visual cortex concept (Patterson & Gibson 2017). Their initial research focused on the visual cortical neurons of cats and monkeys. LeCun et al. (1990) was the precursor of the initial CNN structure called LeNet. From the first CNN model, several researchers designed other CNN frameworks like AlexNet (Krizhevsky et al. 2012), ResNet (He et al. 2015a), Inspection v3, and VGGNet (Simonyan & Zisserman 2015).

A general CNN framework has the following layers:

- **A convolutional layer:** This layer is responsible for the input image features extraction using feature detectors or filters. The layer also generates new images called feature maps of smaller dimensions than the input ones, containing the extracted features of the initial input images. Feature maps usually go through an activation function that enlarges the non-linearity of the generated images. Increasing the non-linearity of the

image produces a more realistic format since most images are predominantly non-linear. It is the last step before entering the next CNN layer. A well-known activation function for CNN is the Rectifier Linear Unit (ReLU) (Glorot *et al.*, 2011). This research integrates an additional activation function called Parameterized Rectifier Linear Unit (PReLU) as a future-proofed option when training larger datasets. PReLU is suitable for a larger dataset than the standard ReLU.

- **A pooling layer:** One of the essential aims of the pooling layer is to create a spatial invariant property for the feature maps. The invariant property enables the CNN model to recognize the same image regardless of its position. The pooling layer also reduces the dimension of the feature maps that become smaller at the output than in the input. The "Max Pooling" (Zhou & Chellappa 1988) is a popular pooling layer method for CNN networks. A CNN model can have several pooling layers to better its accuracy.

- **A flattening layer**: The layer performs the format conversion of the feature map image from the pooling layer from 2-Dimension (2D) to a 1-Dimension (1D) vector. The generated 1D vector becomes the input of the next layer (Yang *et al*. 2019).

- **A full connection layer:** This layer is the brain of the CNN framework. It contains several neurons layers that give intelligence to the whole CNN structure. The neurons are interconnected via synopsis to produce the outcome. The first layer of neurons connects directly to the flattening layer from which it receives the 1D vector as input. Usually, several hidden connection layers exist between the first layer and the output layer. When training a CNN model, the weight of its inputs can be adjusted a couple of times to improve the final accuracy. The count of inputs iterations to improve the CNN model's accuracy is known as Epochs (Devarakonda *et al.* 2017).

Some limitations of CNN algorithm: The CNN algorithm requires a large amount of training dataset to build a reliable model. The training phase of the CNN is time-consuming. It requires a powerful central processing unit (CPU) for faster processing time. Usually, graphics processing units (GPUs) are preferred when training CNN models.

We display in Figure 5-1 a CNN framework basic structure.

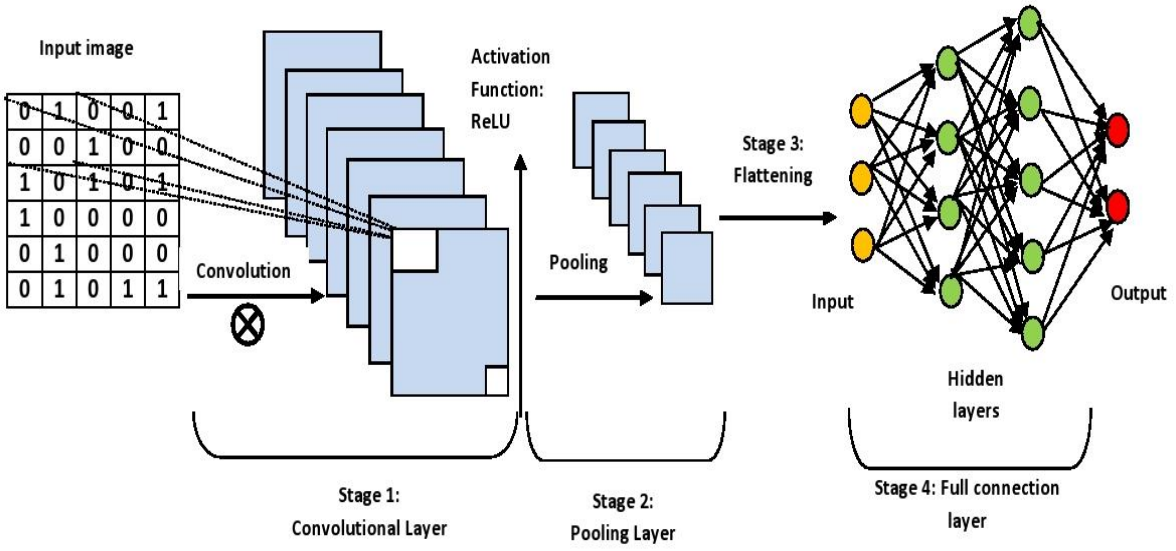

Figure 5-1: CNN framework basic structure (Kiangala & Wang 2020 a)

5.1.2 Parameterized Rectifier Linear Unit (PRELU) activation function option for better results in larger datasets

Various strategies and techniques exist to improve ML models' performances and prevent errors such as overfitting. In the area of deep learning, some of these strategies are: aggressive data augmentation (Krizhevsky *et al.* 2012; Howard 2013; Simonyan & Zisserman 2015 and Szegedy *et al.* 2014), implementation of smaller strides (Zeiler & Fergus 2014; Sermanet et al. 2014; Chatfield et al. 2014 and Simonyan & Zisserman 2015), using large-scale data (Krizhevsky et al. 2012 and Simonyan & Zisserman 2015), implementingmore enormousr depth (Simonyan & Zisserman, A. 2015 and Szegedy *et al.* 2014), new nonlinear activations (Nair & Hinton 2010; Maas *et al.* 2013; Zeiler *et al.* 2013; Lin *et al.* 2013; Srivastava *et al.* 2013 and Goodfellow *et al.* 2013), enlarged width (Zeiler *et al.* 2013 and Sermanet *et al.* 2014), sophisticated layer designs (Szegedy et al. 2014 and He et al. 2015b), and effective regularization techniques (Hinton *et al.* 2012; Srivastava *et al.* 2014; Goodfellow *et al.* 2013 and Wan *et al.* 2013) that enhances generalization. Incorporating activation functions like the ReLU is one method that improves deep networks models (Krizhevsky *et al.*, 2012). In order to improve the design of our CNN classification model, we implement an activation function called PreLU that essentially deals with large-sized networks. We add this technique as an

optional feature for larger datasets. The PReLU activation function has the merits of achieving greater accuracies than traditional ReLU on large networks, and it reduces the risks of overfitting errors when training the models. The PReLU activation function is an improved version of ReLU achieved through several additional parameters boosting the models' operations for deep networks (He *et al.* 2015c).

We present in (5.1), a mathematical expression of the activation function:

$$f(Z_i) = \begin{cases} Z_i & \text{if } Z_i > 0 \\ c_i Z_i & \text{if } Z_i \leq 0 \end{cases} \tag{5.1}$$

where Z_i represents the activation function f input of the *ith* channel and c_i is the slope coefficient.

It is worth mentioning that from (5.1), when c_i value is equal to zero, (5.1) represents the mathematical expression of a traditional ReLU and when its value is equal to a value different than zero it becomes a PReLU activation function equation. We display in Figure 5-2 and 5-3, the graphical representations of the ReLU and PReLU activation functions as defined by (5.1).

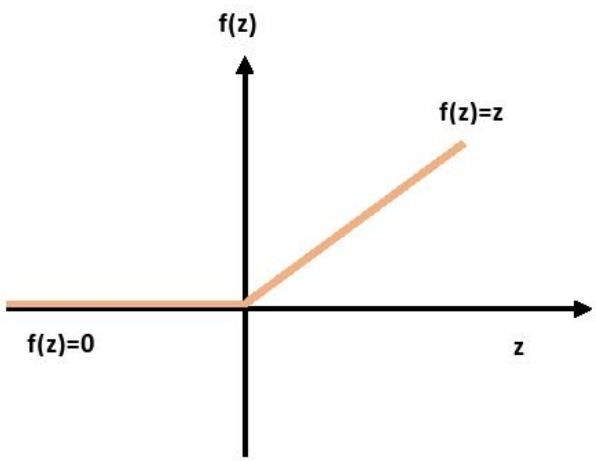

Figure 5-2: ReLU activation function (Kiangala & Wang 2020 a)

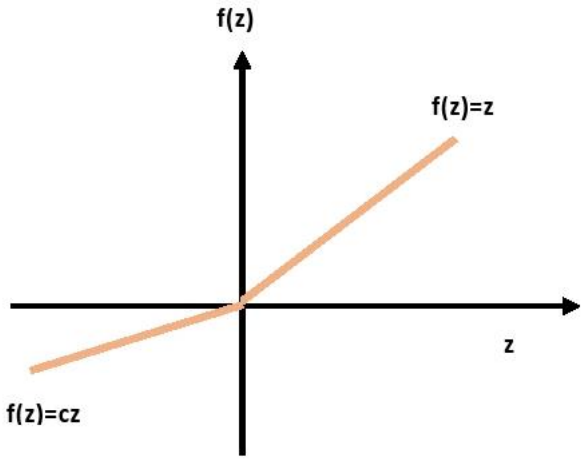

Figure 5-3: PReLU activation function (Kiangala & Wang 2020 a)

We can expand and re-write (5.1) in another well-known format of the activation function (5.2) presented as follows:

$$f(z_i) = \max(0, z_i) + c_i \min(0, z_i) \tag{5.2}$$

A backpropagation method (LeCun *et al.* 1989) is utilized for training deep networks using the PReLU activation function, and its optimization is applied on all layers simultaneously (He *et al.* 2015c).

Another crucial element participating in deep networks models training using PReLU is a gradient. Gradients are optimization algorithms that contributes in obtaining more accurate results. We present in (5.3) the gradient expression of the constant c_i on a single layer:

$$\frac{\partial \varepsilon}{\partial c_i} = \sum_{z_i} \frac{\partial \varepsilon}{\partial f(z_i)} \frac{\partial f(z_i)}{\partial c_i} \tag{5.3}$$

where ε is a symbol for the objection function and $\frac{\partial \varepsilon}{\partial f(z_i)}$ represents the gradient propagation of a neural network deeper layer.

(5.4) is a mathematical expression of the activation function gradient.

$$\frac{\partial f(z_i)}{\partial c_i} = \begin{cases} 0 & if \;\; z_i > 0 \\ z_i & if \; z_i \leq 0 \end{cases} \tag{5.4}$$

5.1.3 Principal components analysis (PCA)

Under the age of I40, there is a tremendous increase in data. This data can originate from production processes and devices monitoring or instruments measurements. However, collecting data without extracting useful information hidden from them could be a waste of resources. The challenge is even higher when dealing with vast amounts of data. We need appropriate methods to lower data dimension, keep the original data features, and ease their interpretability. PCA is one of the oldest techniques developed to achieve this purpose (Giordani & Kiers 2004). PCA's goal is to reduce large datasets' dimensions while minimizing the information loss rate to carry as much as characteristics of the original large datasets to the reduced data version. As part of the dimension reduction process, PCA generates new uncorrelated variables that favorably maximize data variance. The new variables' numerical values are the outcomes of eigenvectors and eigenvalues problems solved through the process. PCA is considered an adaptive approach for data analysis (Jolliffe & Cadima 2016).

Pearson (1901) and Hotelling (1933) are considered the pioneer of the earliest knowledge on the PCA theory. Several developments and advances exist on the concept, especially with computerization that brought forth new types of datasets. To this day, several disciples integrate and exploit PCA's advantages for successful data interpretation and analysis. When reducing data dimension, PCA discloses the correlation between variables of the dataset and performs its operation on the initial data space. Its operation principle also relies on several statistical analyses on the dataset to ensure feature retention of new variables (Zhang et al., 2019).

We consider a dataset matrix, B, composed of *c x r* elements with c representing the dataset variables on the matrix column and r the data samples on the matrix rows. We can express B as a total of the r vectors cross products as presented in (5.5) (Zhang et al. 2019):

$$B = g_1 s_1^G + g_2 s_2^G + \cdots + g_c s_c^G = GS^G \tag{5.5}$$

where g_i represents the score vector of the dataset and s_i is its load vector. Both factors belong to R^n. G represents the load matrix of the dataset. The score vector of the dataset B is also known as the principal component of B.

B can be further transformed in (5.6) by ignoring some negligible factors considered as residual elements and only maintaining the first *n* principal elements.

$$B = \sum_{i=1}^{n} g_i s_i^G + \sum_{i=n+1}^{c} g_i s_i^G = G_n S_n^G + E \tag{5.6}$$

By using some principal components, process data can be reformulated. $\hat{B}$ is the estimated dataset after the reformulation and is presented by (5.7).

$$\hat{B} = \sum_{i=1}^{n} g_i s_i^T = B S_n S_n^G \tag{5.7}$$

where S_n represents a matrix made of the load matrix first vector *n*.

Another useful factor for the generation of PCA is the statistics component G^2 that enables the monitoring of several principal components concurrently. (5.8) is an expression of G^2 for a process variable vector B_i at a time *i*.

$$G^2 = g_i \theta^{-1} g_i^G = B_i S \theta^{-1} S^G B_i^G \tag{5.8}$$

where g_i represents the element of an ith row of a matrix G_n that contains *n* principal component score vectors. θ is a parameter representing the diagonal matrix of the first diagonal *n*.

Figure 5-4 is a summary of steps required to generate PCA variables.

Some limitations of PCA: Although the PCA concept has the advantage of not restricting parameters for its configuration, it has the drawback of limiting the improvement of its results

through the addition of prior knowledge of data parameters since users are not permitted to intervene in the PCA processing part using external methods or parameterization.

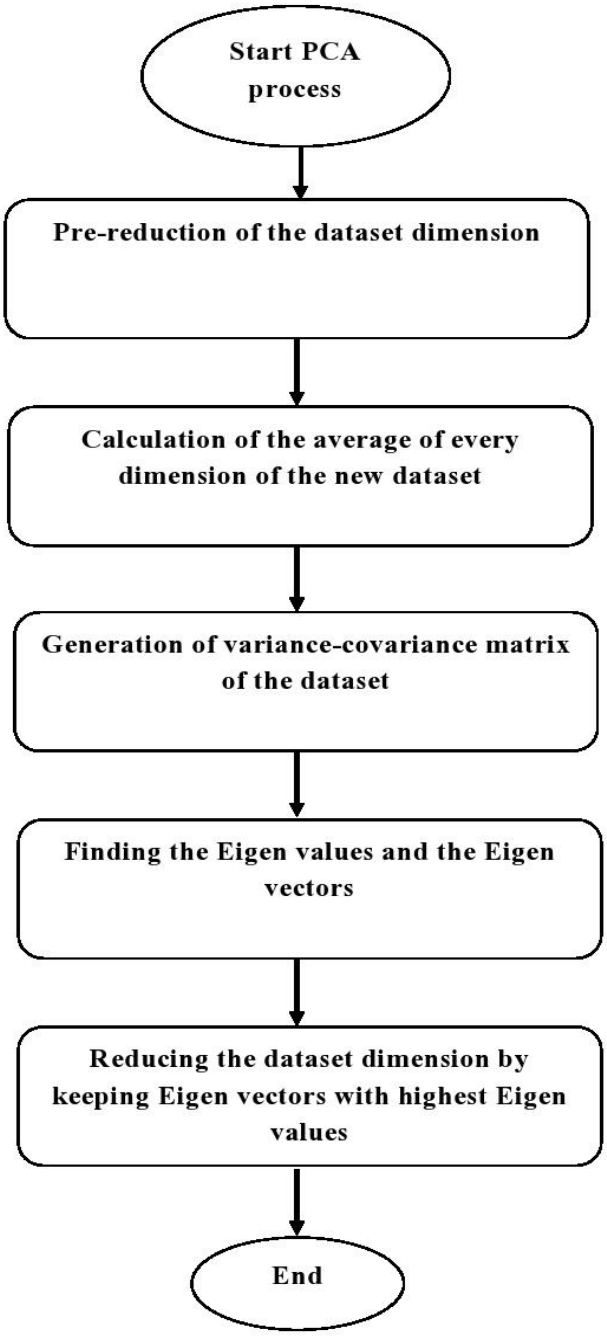

Figure 5-4: PCA design steps

5.1.4 Time-series data imaging

Time-series data imaging is the process of transforming time-series data into images. This process benefits our intelligent PM model since we carry out the classification task in a CNN model. As mentioned in previous sections, the input of a CNN structure is an image. Therefore, we need to implement a data transformation process that converts the time-series data into images. Time-series imaging is part of "data augmentation," in which smaller dataset sizes are changed to higher ones, for example, increasing a 1D dataset to a 2D or 3D one. A method utilized to obtain time-series images by encoding time-series information is the Gramian Angular Field (GAF) introduced by Wang & Oates (2015). In the imaging process, GAF implements a polar coordinates-based matrix to represent the time-series data. As opposed to the popular Cartesian coordinate**,** polar coordinate can maintain temporal data correlation. The GAF process produces two categories of images: Gramian angular summation field (GASF) and Gramian angular differential field (GADF). Image encoding using GAF has the following steps:

- **Time-series data input normalization:** Considering the time-series data presented in (4.1), a normalization method is applied in the intervals of [-1, 1] to produce a normalized or scaled time-series data as displayed in (5.9).

$$\widetilde{b_{-1}}^{\,i} = \frac{(b_i - max(B)) + (b_i - min(B))}{max(B) - min(B)} \tag{5.9}$$

where $\widetilde{b_{-1}}^{\,i}$ represents the normalized value of each initial time-series data b_i.

- **Normalized time-series data conversion to polar coordinates:** In this step, the previous normalized time-series data $\tilde{B}$ with elements $\widetilde{b_{-1}}^{\,i}$ is represented into a polar coordinates. The polar coordinate comprises the angular cosine of every normalized time-series data $\widetilde{b_{-1}}^{\,i}$ and the radius of the time-series data which depends on a time stamp. (5.10) and (5.11) are mathematical expressions of the two polar coordinates.

$$\theta = \arccos\ (\widetilde{\widetilde{b_i}}) \tag{5.10}$$

where $-1 \leq \widetilde{b_i} \leq 1, \quad \widetilde{b_i} \in \tilde{B}$.

θ is the equivalent of a time-series value in the polar coordinates format.

$$r = \frac{t_i}{N}, t_i \in \mathbb{N} \tag{5.11}$$

where t_i is the time stamp of the time-series data .

N is the stabilizing factor (a constant) of the polar coordinate system's space.

- **Generating the GASF and GADF from the polar coordinate**: The GASF and GADF of the time-series data are the trigonometric sum and difference of each polar point computed to obtain the spatial relationship between them. We present in (5.12) and (5.13) the mathematical expressions of GASF and in **(**5.14) and (5.15) those representing GADF.

$$GASF = \left[\cos\left(\theta_i + \theta_j\right)\right] \tag{5.12}$$

$$GASF = \tilde{b}'.\tilde{b} - \sqrt{1 - \tilde{b}^{2\prime}}.\sqrt{1 - \tilde{b}^2} \tag{5.13}$$

$$GADF = \left[\sin\left(\theta_i - \theta_j\right)\right] \tag{5.14}$$

$$GADF = \sqrt{1 - \tilde{b}^{2\prime}}.\tilde{b} - \tilde{b}'\sqrt{1 - \tilde{b}^2} \tag{5.15}$$

- We can also represent the GASF and GADF in a matrix format that is usually easier to understand in the design process. (5.16) and (5.17) are the matrices representations of GASF and GADF.

$$GASF = \begin{pmatrix} \cos(\theta_1 + \theta_1) & \cdots & \cos(\theta_1 + \theta_n) \\ \vdots & \ddots & \vdots \\ \cos(\theta_m + \theta_1) & \cdots & \cos(\theta_m + \theta_n) \end{pmatrix} \tag{5.16}$$

$$GADF = \begin{pmatrix} \sin(\theta_1 - \theta_1) & \cdots & \sin(\theta_1 - \theta_n) \\ \vdots & \ddots & \vdots \\ \sin(\theta_m - \theta_1) & \cdots & \sin(\theta_m - \theta_n) \end{pmatrix} \tag{5.17}$$

We display in Figure 5-5, the required steps for time-series imaging using the GAF process.

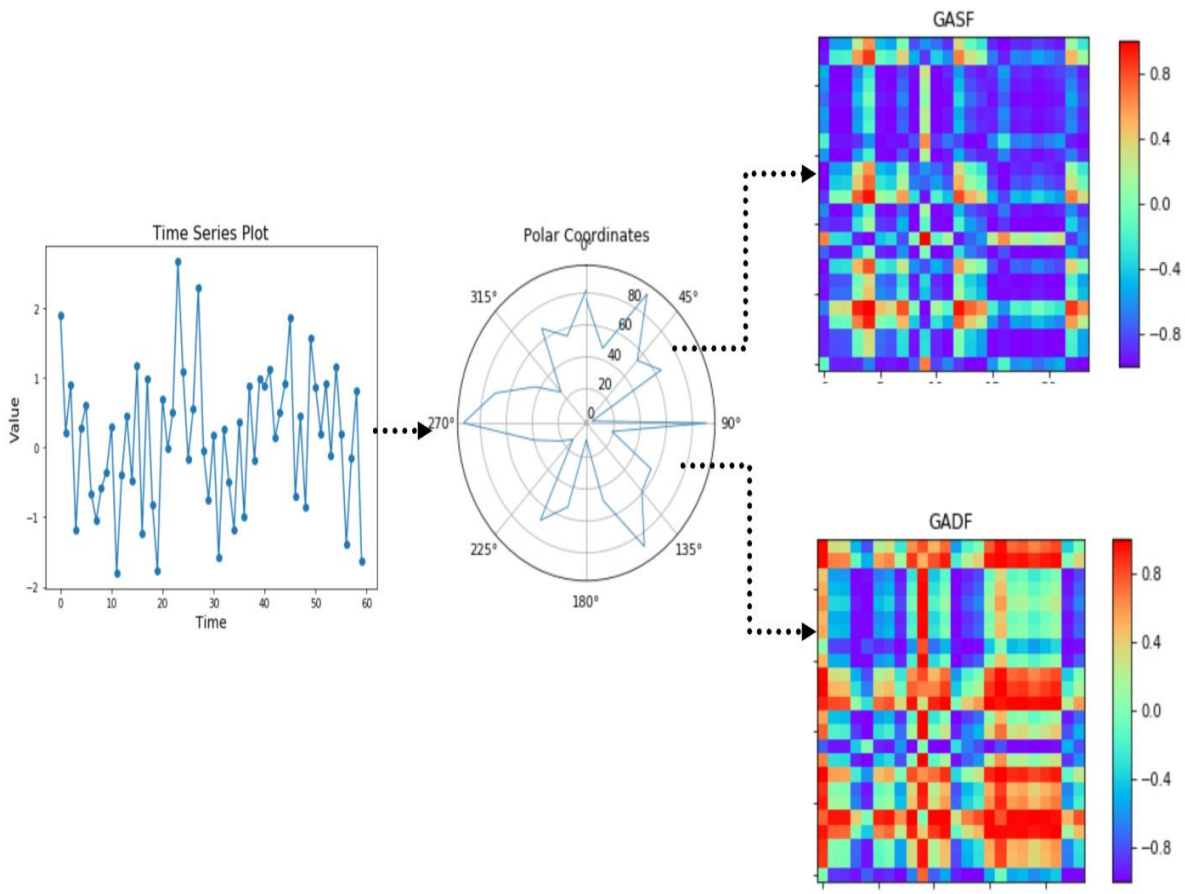

Figure 5-5: Time-series imaging process highlight using GAF (Kiangala & Wang 2020 a)

5.2 Modelling of an intelligent PM framework with developed AI techniques and Innovative methods

Our intelligent PM framework goal is to classify conveyor motor conditions into threats for the overall system or acceptable for operations by transforming motors observations (time-series data) into images loaded into a CNN model to accomplish the classification task. We divide the PM framework into three segments:

1) The feeding segment

In this framework segment, we enable the system to cater to two types of time-series data, two inputs: MTS and UTS. We are dealing with MTS data inputs. We do not feed the input data directly to the image encoding segment but enable a **"Dimensionality Reduction Stage"** sub-segment to decrease the MTS data inputs dimension to two channels by implementing PCA. The MTS dimension reduction improves the system performance and lessens its complexity. Processing data in smaller volumes is advantageous for the system.

2) The imaging segment

In this segment, we convert our UTS input data or the reduced version of the MTS input data into GAF images that will be later used as the CNN classification model input.

3) CNN classification modelling segment

This segment of our PM framework process the GAF images from the imaging segment and classifies them into system threats or normal condition with the CNN ML algorithm. In order to build a future-proof model prepared for more extensive data volumes, we implement a PReLU activation function option in the CNN model. The PReLU activation function also has the merit of bettering the non-linearity attribute of input images. We test our experimental PM framework on small manufacturing industries data and obtain similar performance results for CNN with standard ReLU and PReLU.

Note: Some researchers claim that accuracy improvements for CNN models when implementing PReLU activation function over ReLU are negligibly ranging between 1% to 2% accuracy enhancement and are sometimes not worth the implementation effort. However, a slight accuracy improvement for critical manufacturing activities could make a massive difference in a

manufacturing factory where machines' availability is critical to the overall production. An increase of 1% accuracy could be the information required to avoid chaos in the factory.

Figure 5-6 is a representation of the overall PM framework architecture and Figure 5-7 its work flow diagram.

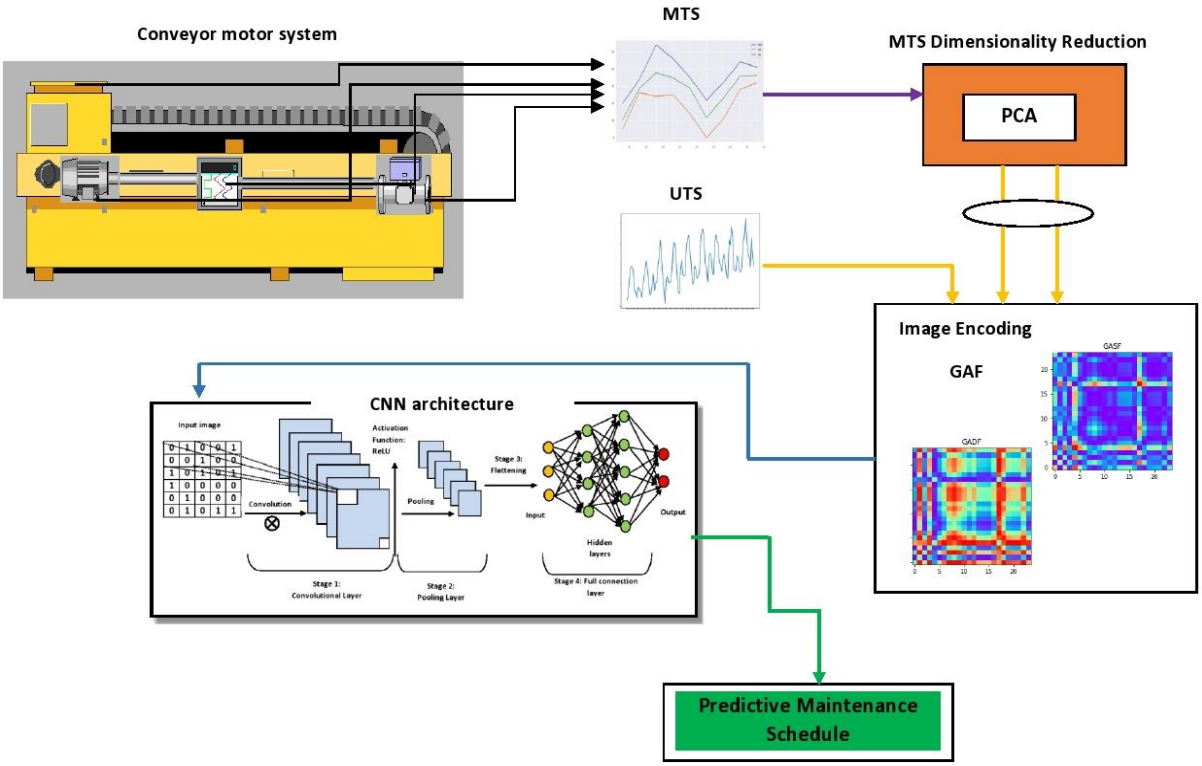

Figure 5-6: Intelligent PM framework overall architecture (Kiangala & Wang 2020 a)

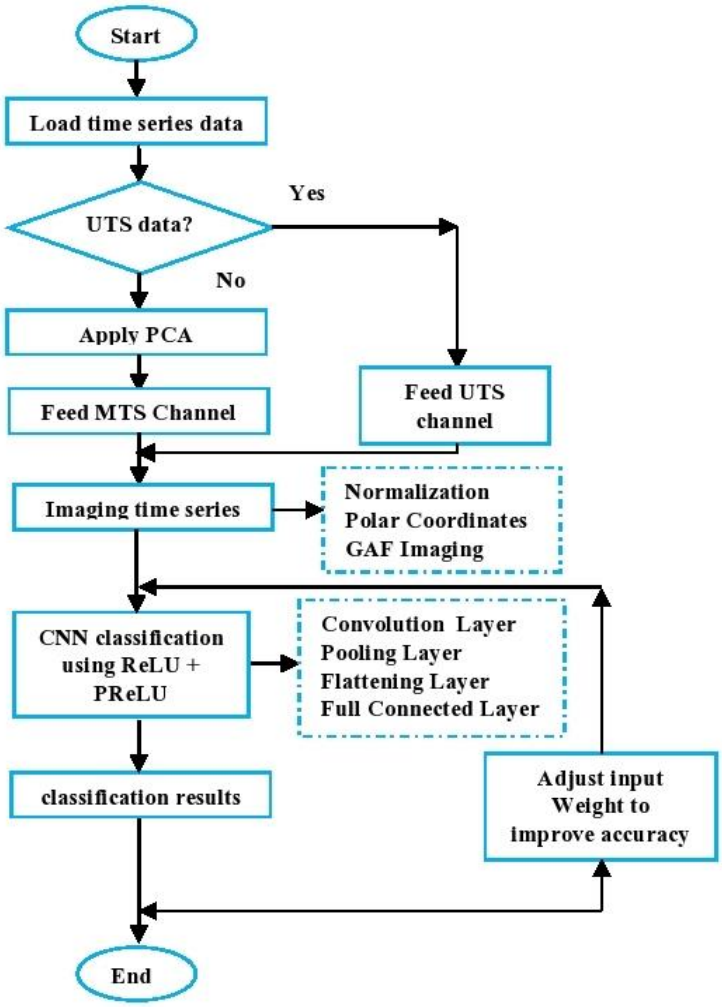

Figure 5-7: PM flow chart diagram (Kiangala & Wang 2020 a)

5.2.1 Applying PCA

As described in Figure 5-4, we apply several steps to our input dataset to achieve PCA's final data dimension reduction. We detail the PCA processing steps of our experimental dataset as follows:

1) Pre-reduction of dataset dimension

Our experimental dataset has twelve parameters. Eleven of these parameters are conveyor motor features such as Vibration speed, Motor torque, Acceleration, Motor Speed, Air pressure, Product Weight, Deceleration, Current, Belt tension, Motor tension, and Temperature. The last parameter is the conveyor motor Fault type detected in the system over time for each parameter

combination. The total number of observations in a specific interval per parameter is 15,000. We can write an expression of our overall experimental dataset based on the motor parameters as p + 1, with p representing the number of independent time-series variables (or the observations) and '1' the number of the dependent variable of the dataset. The dependent variable in our experiment is the motor fault. The pre-reduction of the dataset dimension consists of discarding the dependent variable and keeping the number of independent variables. For our experimental dataset, the new dataset dimension equals the value of p, which is eleven.

2) Calculate the average of every dimension of the new dataset

From the new reduced dataset dimension p = 11, we can extract eleven vectors of observations, each with different values. We present the observation vectors as follows (Kiangala & Wang 2020 a):

$$\mathrm{P1(vibration\ speed)} = [\mathrm{p}_{11}, \mathrm{p}_{12}, \cdots, \mathrm{p}_{1\mathrm{n}}]$$

$$\mathrm{P2(motor\ torque)} = [\mathrm{p}_{21}, \mathrm{p}_{22}, \cdots, \mathrm{p}_{2\mathrm{n}}]$$

$$\mathrm{P3(acceleration)} = [\mathrm{p}_{31}, \mathrm{p}_{32}, \cdots, \mathrm{p}_{3\mathrm{n}}]$$

$$\mathrm{P4(motor\ speed)} = [\mathrm{p}_{41}, \mathrm{p}_{42}, \cdots, \mathrm{p}_{4\mathrm{n}}]$$

$$\mathrm{P5(air\ pressure)} = [\mathrm{p}_{51}, \mathrm{p}_{52}, \cdots, \mathrm{p}_{5\mathrm{n}}]$$

$$\mathrm{P6(product\ weight)} = [\mathrm{p}_{61}, \mathrm{p}_{62}, \cdots, \mathrm{p}_{6\mathrm{n}}]$$

$$\mathrm{P7(deceleration)} = [\mathrm{p}_{71}, \mathrm{p}_{72}, \cdots, \mathrm{p}_{7\mathrm{n}}]$$

$$\mathrm{P8(current)} = [\mathrm{p}_{81}, \mathrm{p}_{82}, \cdots, \mathrm{p}_{8\mathrm{n}}]$$

$$\mathrm{P9(belt\ tension)} = [\mathrm{p}_{91}, \mathrm{p}_{92}, \cdots, \mathrm{p}_{9\mathrm{n}}]$$

$$\mathrm{P10(motor\ tension)} = [\mathrm{p}_{101}, \mathrm{p}_{102}, \cdots, \mathrm{p}_{10\mathrm{n}}]$$

$$\mathrm{P11(temperature)} = [\mathrm{p}_{111}, \mathrm{p}_{112}, \cdots, \mathrm{p}_{11\mathrm{n}}]$$

where *n* represent the number of observations for each parameter.

As mention earlier, in this experiment, we use 15 000 observations of our parameters. In other words, the value of n equals 15 000 (n = 15 000). From the above observation vectors, we can

create a matrix representing our overall dataset. The matrix is of size p x n (11, 15 000) and is presented in (5.18).

$$D = \begin{pmatrix} p_{\{1,1\}} & \cdots & p_{\{6,1\}} & \cdots & p_{\{11,1\}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ p_{\{1,7500\}} & \cdots & p_{\{6,7500\}} & \cdots & p_{\{11,7500\}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ p_{\{1,15000\}} & \cdots & p_{\{6,15000\}} & \cdots & p_{\{11,15000\}} \end{pmatrix} \tag{5.18}$$

(5.19) is a mathematical expression of the average of each dimension of the experimental dataset matrix D.

$$AvgP1 = \frac{\sum_{i=1}^{n}(\mathrm{P_{1n}})}{\mathrm{n}} = \overline{\mathrm{P1}} \tag{5.19}$$

We can also compute an illustration of the average matrix, D, by using the expressions (5.18) and (5.19). The average matrix is presented in (5.20) and summarized in (5.21).

$$\overline{D} = [\frac{\sum_{i=1}^{n}(p_{1n})}{n} \dots \frac{\sum_{i=1}^{n}(p_{6n})}{n} \dots \frac{\sum_{i=1}^{n}(p_{11n})}{n}] \tag{5.20}$$

$$\overline{D} = [\overline{P1} \cdots \overline{P6} \dots \overline{P11}] \tag{5.21}$$

3) Generate the variance-covariance matrix of the dataset D

In order to generate the variance-covariance matrix, also known as the covariance matrix, we apply (5.22) to create a variance relationship for each vector of the dataset D.

$$\mathrm{VC(P1, P2)} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{P_{1n}} - \overline{\mathrm{P1}})\,(\mathrm{P_{2n}} - \overline{\mathrm{P2}}) \tag{5.22}$$

The outcome of the variance-covariance matrix section after applying (5.22) is a square matrix with the size **p x p;** Based on our experimental dataset, the variance-covariance matrix to build the PM framework has the size **11 x 11.** We present in Table 5-1 a representation of the variance-covariance matrix. We simplify the matrix representation due to space limitations.

Table 5-1: Variance-covariance matrix result

	P1	P2	...	P11
P1	VC(P1,P1)	VC(P1,P2)	...	VC(P1,P11)
P2	VC(P2,P1)	VC(P2,P2)	...	VC(P2,P11)
⋮	...	...	...	...
P11	VC(P11,P1)	VC(P11,P2)	...	VC(P11,P11)

It is worth mentioning that in Table 5-1 VC (P1, P2) =VC(P2, P1).

4) Find the eigenvalues and their eigenvectors

Munir, M. (2015) describes an Eigenvector as a vector that does not change directions after going through any linear transformation to it. Assuming (5.23) to be an expression of our square variance-covariance matrix:

$$CVM = \begin{pmatrix} VC(P1,P1) & \cdots & VC(P1,P6) & \cdots & VC(P1,P11) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ VC(P6,P1) & \cdots & VC(P6,P6) & \dots & VC(P6,P11) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ VC(P11,P1) & \dots & VC(P11,P6) & \dots & VC(P11,P11) \end{pmatrix} \tag{5.23}$$

We implement (5.24) to find a mathematical expression of the Eigenvalues for the CVM matrix in (5.23).

$$\det(\mathrm{CVM} - \lambda \mathrm{I}) = 0 \tag{5.24}$$

where λ represents the Eigenalue of the CVM matrix and I is an expression of the identity matrix. We define in (5.25) an expression of the identity matrix associated with the matrix CVM in (5.23).

$$I = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 1 & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 1 \end{pmatrix} \tag{5.25}$$

As per (5.25), we notice that the identity matrix, I, is a square matrix with the size of the CVM matrix (11 x 11). We can generate an equation of the eleventh degree (5.26) by substituting the expressions of (5.23) and (5.25) into (5.24). The variable λ is the unknown.

$$a\lambda^{11} + b\lambda^{10} + h\lambda^{4} + \cdots + i\lambda^{3} + j\lambda^{2} + k\lambda + l = 0 \tag{5.26}$$

From (5.26), we solve the equation for the unknown λ and find eleven values: λ_1, λ_2, λ_3, λ_4, λ_5, λ_6, λ_7, λ_8, λ_9, λ_{10} and λ_{11} that represent the Eigenvalues of the original matrix. Using the Eigen values, we should compute the Eigenvectors corresponding to each Eigenvalues. We consider the following Eigenvalues with their Eigenvectors:

$\lambda_1 \rightarrow \mathrm{E}_1 = [\mathrm{e}_{11}, \mathrm{e}_{12}, \cdots, \mathrm{e}_{111}]$

$\lambda_2 \rightarrow \mathrm{E}_2 = [\mathrm{e}_{21}, \mathrm{e}_{22}, \cdots, \mathrm{e}_{211}]$

$\lambda_3 \rightarrow \mathrm{E}_3 = [\mathrm{e}_{31}, \mathrm{e}_{32}, \cdots, \mathrm{e}_{311}]$

$\lambda_4 \rightarrow \mathrm{E}_4 = [\mathrm{e}_{41}, \mathrm{e}_{42}, \cdots, \mathrm{e}_{411}]$

$\lambda_5 \rightarrow \mathrm{E}_5 = [\mathrm{e}_{51}, \mathrm{e}_{52}, \cdots, \mathrm{e}_{511}]$

$\lambda_6 \rightarrow \mathrm{E}_6 = [\mathrm{e}_{61}, \mathrm{e}_{62}, \cdots, \mathrm{e}_{611}]$

$\lambda_7 \rightarrow \mathrm{E}_7 = [\mathrm{e}_{71}, \mathrm{e}_{72}, \dots, \mathrm{e}_{711}]$

$\lambda_8 \rightarrow \mathrm{E}_8 = [\mathrm{e}_{81}, \mathrm{e}_{82}, \dots, \mathrm{e}_{811}]$

$\lambda_9 \rightarrow \mathrm{E}_8 = [\mathrm{e}_{91}, \mathrm{e}_{92}, \dots, \mathrm{e}_{911}]$

$\lambda_{10} \rightarrow E_{10} = [e_{101}, e_{102}, \dots, e_{1011}]$

$\lambda_{11} \rightarrow E_{11} = [e_{111}, e_{112}, \dots, e_{1111}]$

5) Reduce dataset dimension by keeping eigenvectors with highest eigenvalues

The last step to have the final dataset reduced dimension consists of focusing on the Eigenvectors with the highest Eigenvalues since those containing the smallest Eigenvalues do not portray helpful information of the original dataset. Our PM framework's dimensionality reduction aims to reduce the dataset size from p=11 to 2, corresponding to the two channels we would like to have on our MTS input side. Therefore, we only chose the first two Eigenvalues having higher values and their corresponding Eigenvectors. Assuming that λ_1 and λ_2 are the two highest Eigenvalues with $\lambda_1 > \lambda_2$, we can compute a new matrix (5.27) of their corresponding Eigenvectors as:

$$G = \begin{pmatrix} e_{11} & e_{12} & \dots & e_{111} \\ e_{21} & e_{22} & \dots & e_{211} \end{pmatrix} \tag{5.27}$$

In order to obtain the reduced dataset dimension from the matrix G, we compute the mathematical expression (5.28) as follows:

$$Z = DG^T \tag{5.28}$$

where G^T represents the transpose of matrix G and can be defined by (5.29):

$$G = \begin{pmatrix} e_{11} & e_{21} \\ e_{12} & e_{22} \\ \vdots & \vdots \\ e_{111} & e_{211} \end{pmatrix} \tag{5.29}$$

By multiplying the original matrix D from (120) of size 15 000 x 11 to the transpose matrix of G, G^T, a matrix of size 11 x 2, we acquire a new matrix of size 15 000 x 2. The number 2 represents the number of columns of the reduced dataset.

5.2.2 Conveyor system data for the CNN model

We test the effectiveness of our PM framework with data from a small experimental conveyor system in a manufacturing plant. The conveyor system has three main components: a conveyor AC motor, a variable speed drive (VSD), and a conveyor belt structure. Based on the recorded vibration speed, the factory could initiate a predictive maintenance schedule knowing issues rising from excessive motor vibration speeds. We illustrate the vibration level in machinery in Figure 5-8, with vibration thresholds represented as normal, warning, or alarm. These states depend on the motor size.

A vibration speed can be represented as a sine wave with its period, frequency, and amplitude attributes. We illustrate in Figure 5-9 the sine wave of a vibration signal. The speed of the vibration signal is the first derivative of the vibration amplitude, also referred to as the displacement computed over a certain amount of time. The analysis of motor vibrations on their own does not offer accurate results on the motor's health status. Several approaches that define the impact of vibration speeds based on the motor size exist. We utilize one of these approaches in Table 5-2. Several factories utilize special vibration sensors mounted in motors to record the vibration speed values and save them in controllers for computation purposes.

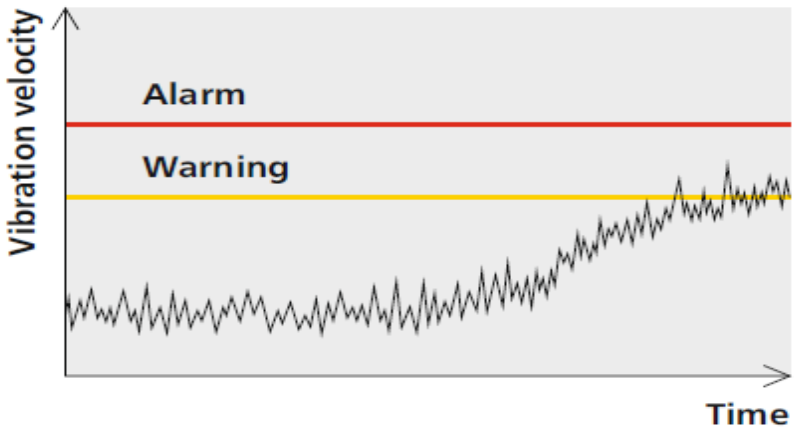

Figure 5-8: Machine vibration trend according to ISO 10816 (Ifm electronics 2013).

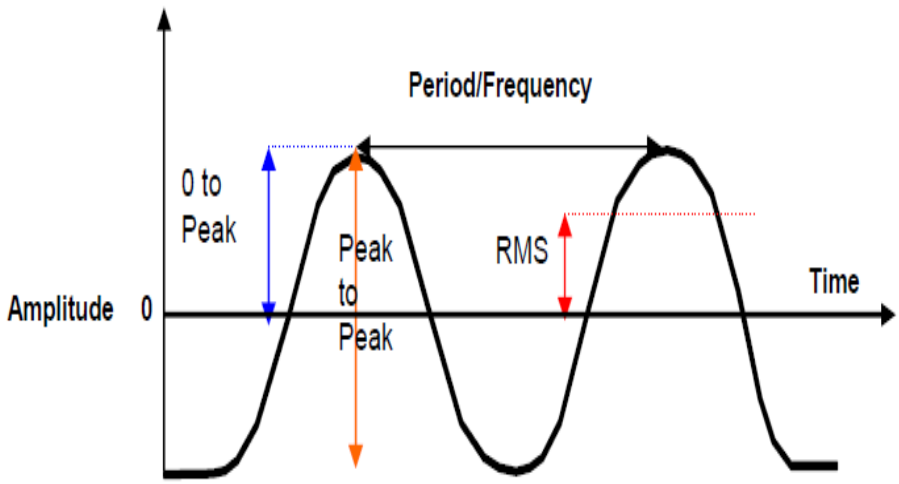

Figure 5-9: Vibration as a sine wave (Sanders 2011)

One of the reasons why a conveyor system could break is an excess of deformation, which is nothing else but an amplitude variation that repeatedly arises at a specific frequency run. A machine can experience high amplitude changes at a given frequency (usually a higher frequency). The velocity being a function of displacement and frequency can become an excellent gauge of the vibration severity in a motor. Alsalaet (2012) states that for machines running at frequencies ranging between 10Hz and 1000Hz, the vibration velocity is a good measure of the vibration severity. For heavy machines operating at frequencies above 1000Hz, the acceleration is a better indicator of the vibration severity.

The motor sizes and types for vibration severity classification in Table 5-2 can be dissected into the four classes as follows (Kiangala & Wang 2020 a):

Class I is for Small-sized equipment: for motor power ranging between 0 and 15KW.

Class II is for Medium-sized equipment: for motor power ranging between 15 and 75KW.

Class III is for Large-sized equipment (for motor power over 75KW) mounted on "Rigid Support" structures and foundations.

Class IV is for Large-sized machines (for motor power over 75KW) mounted on "Flexible Support" structures (Kiangala & Wang 2018).

We use a medium-sized equipment motor of class II for our experimental PM framework, widely installed in manufacturing SMEs factories. Considering class II Medium-sized

equipment, we can observe in Table 5-2 that vibration velocity values above **4.5mm/s** are dangerous and demonstrate a high severity. However, the plant supervisors noticed that not all high vibration speeds would result in a "critical" system fault over time. A combination of the vibration speed with several other parameter values would be best to analyse to determine the best scenarios of critical fault. It is the aim of our intelligent PM framework.

Table 5-2: Vibration severity criteria based on ISO 2372 (Kiangala & Wang 2018)

RMS Overall Velocity Level in 1000 Hz Bandwidth		Vibration Severity Criteria			
mm/s	**In/s**	**Class I**	**Class II**	**Class III**	**Class IV**
0.28	0.01	Good	Good	Good	Good
0.45	0.02				
0.71	0.03				
1.12	0.04	Satis-factory			
1.8	0.07		Satis-factory		
2.8	0.11	Unsatis-factory		Satis-factory	
4.5	0.18		Unsatis-factory		Satis-factory
7.1	0.28	Unac-ceptable		Unsatis-factory	
11.2	0.44		Unac-ceptable		Unsatis-factory
18	0.71			Unac-ceptable	
28	1.10				Unac-ceptable
45	1.77				

It is essential to note that the experimental conveyor system from which the PM framework is built and tested runs in three main speeds regulated by the VSD: f1=15Hz, f2=30Hz, and f3=50Hz. The sampling frequency (fs) utilized in the assessment is fs=2.56f3 that is about fs= 128Hz. The conveyor system generated three primary kinds of faults:

- **A misalignment fault:** This fault is caused by the continuous motion of the conveyor belt that gets the machine's shaft out of line and results in a misalignment fault that occasions system vibrations.

- **A looseness fault:** A conveyor system's structure should be rigid and stiff in everyday operations to produce acceptable outcomes. Whenever the conveyor motor structure's stiffness decreases, vibrations arise in the system. These vibrations are usually not very severe, and the conveyor belt can run with them for a long time without being noticed. Proper monitoring equipment is required to record these types of vibrations, especially in the beginning. In the small manufacturing factories, the looseness fault is often the least severe of all vibration faults. Machine supervisors noticed that the misalignment and the looseness fault individually produce less severe faults in the conveyor system. Their vibration velocity values from Table 5-2 are relatively low, nearing the unsatisfactory limit of **4.5 mm/s.** However, when these two faults co-occur, they produce high vibrations velocities in the unsatisfactory spectrum in Table 5-2.

- **An imbalance fault:** Unattended broken parts often generate an imbalance fault in the conveyor system that slowly causes vibrations. When running at low speed, the imbalance fault can remain unnoticed. Its vibrations become very visible when operating from a frequency of 30Hz and worst for frequencies above 50Hz. Vibration velocities at these frequencies are all in the unsatisfactory range. The effect of excessive vibrations considerably reduces the effectiveness of a conveyor system. A good example is the spillage of beverages while bottling them through a vibrating conveyor system in a bottling plant.

5.2.3 Evaluation of ML classification models

After building our CNN classification model, we need some metrics to evaluate the reliability of our results. We choose the following metrics for our classification models:

- **The accuracy:** The accuracy of a classification model represents the percentage of correct predictions done by the model over the total number of prototypes utilized in the prediction process. Assuming that the variable CP represents the number of correct predictions of the classification model and n the total number of samples in the prediction process, we can illustrate an expression of the accuracy by (5.30).

$$accuracy = \frac{CP}{n} 100\% \tag{5.30}$$

- **The precision**: The precision metric represents the percentage of correct prediction of each category over the overall number of specimens predicted for those classes. Our PM framework has three classes: the no-fault (NF) class, the minor fault (MF) class, and the critical fault (CF) class. We present in (5.31) an expression of the precision metric.

$$precision = \frac{CP_m}{P_m} 100\% \,, \quad m \in \mathbb{N} \tag{5.31}$$

where m represents the count of categories or classes of the dataset, CP_m represents the count of correct predictions for each category, and p_m is the total count of samples (correct and incorrect predictions) predicted for that class.

- **The Recall**: The recall metric represents the percentage of correctly predicted instances of a category. It is the ratio of correct predictions in a class over the correct and incorrect predictions for that category. (5.32) is a mathematical expression of the recall.

$$recall = \frac{CP_m}{(CP_m + IP_m)} 100\% \,, \quad m \in \mathbb{N} \tag{5.32}$$

where m represents the count of categories or classes of the dataset, CP_m is the count of correct predictions instances for each category (true positives), and IP_m represents the count of incorrect predicted samples for that category (false negatives).

The accuracy is a metric that evaluates the overall model with all classes included. The precision and the recall assess individual classes or categories and provide good insights into each of them. An essential tool utilized in classification model evaluation models is the confusion matrix (Visa *et al.* 2011). It contains results of predicted labels versus actual ones for classification models. It also facilitates the computation of metrics such as precision, recall, and accuracy.

Note: The above described metrics are applicable to any ML classification model.

5.2.4 Assumptions

We assume a programmed script automatically collects the time-series data and loads them into the image encoding module. An image scanner is connected to the CNN classification model from the image encoding module to determine the type of fault detected from time-series images automatically.

5.3 Implementation of an intelligent PM framework using time-series data imaging and CNN

5.3.1 Data collection and analysis

- **Time-series data** are recorded from a conveyor motor operation to detect abnormalities and create a PM model using ML algorithms. We utilized the collected data for training the system's PM model to detect possible failures from future data automatically. We gathered eleven motor parameters, known as variables, observed over a period. Our PM model aims to determine which parameters combinations values are safe (no-fault), warnings (minor fault), or dangerous (critical fault). The parameters are MTS independent variables for the deep learning ML model. We present the eleven

parameters (from var1 to var11) attributes in Table 5-3 and a graphical representation of the time-series data in Figure 5-10. We choose to display a small portion of the data for visibility purposes. It is worth mentioning that, usually, these parameter values are scaled in controllers and SCADA to exhibit an understandable value range for operators.

Table 5-3: Time-series variables attributes (Kiangala & Wang 2020 a)

	Parameters	**Units**
Var 1	Vibration speed	m/s
Var 2	Motor Torque	Nm
Var 3	Acceleration	mm^2/s
Var 4	Motor Speed	Hz/s
Var 5	Air Pressure	bar
Var 6	Product Weight	kg
Var 7	Deceleration	mm^2/s
Var 8	Current	A (Amps)
Var 9	Belt tension	N/m
Var 10	Motor tension	N/m
Var 11	Temperature	*C

- **Knowledge-based data** from previous manual product configuration via SCADA are analysed and refined to develop automatic ML models that accurately predict corresponding machine and product parameters from entered input values. We used this data type to generate a self-configurable parameter SCADA system and implement a product customization platform for clients.

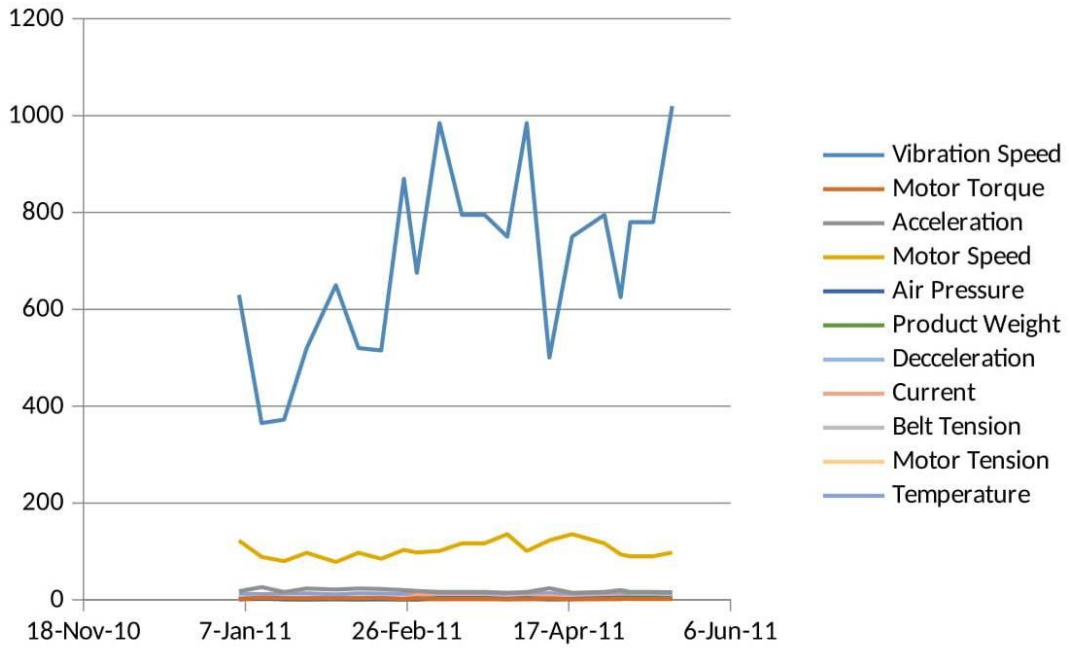

Figure 5-10: MTS independent variables (Kiangala & Wang 2020 a)

We collected some knowledge-based data and product configuration parameters from a small rubber manufacturing plant to teach an ML model how to predict these parameters automatically.

The parameters dataset contains of the following variables:

- Level of Material_A (in gram)
- Product Weight (in gram)
- Width (cm)
- Height (cm)
- Diameter (cm)
- Curing Time (seconds)
- Heating pressure (seconds)
- Number of Bumps (seconds)
- Bump delays (seconds)

The rubber plant's operation depends on the correct configuration of the above variables into the system via their SCADA. The configuration has been done manually before every product

manufacturing. Four of these parameters are critical to reaching the desired rubber composition and need supervisors' approval due to their equivalence with other product features. The four parameters are Curing Time, Heating pressure, Number of Bumps, and Bumps delays. The value of these four parameters results from the linearity or the non-linearity between Level of Material_A, Product Weight, Width, Height, and Diameter known from the manufacturing rubber product before the production. We can therefore divide the dataset into two categories:

Independent variables data:

- Level of Material_A (in gram)
- Product Weight (in gram)
- Width (cm)
- Height (cm)
- Diameter (cm);

And dependent variables data:

- Curing Time (seconds)
- Heating pressure (seconds)
- Number of Bumps (seconds)
- Bump delays (seconds)

The outcome of our intelligent PM framework is a classification model that categorizes data from several motor parameters and observations loaded as inputs of the system as No-Fault, Minor Fault, or Critical Fault. These three faults are the output of the system. Figure 5-11 illustrates the motor parameters combination as system inputs with the classified faults as outputs.

- Vibration speed (VS),
- Motor torque (MT),
- Acceleration (ACC),
- Motor speed (MS),
- Air pressure (AP),
- Product weight (PW),
- Deceleration (DEC),

- Current(CUR),
- Belt tension (BT),
- Motor tension (MT-S)
- Temperature (TMP)

5.3.2 Reducing the dataset dimension with PCA

As per our PM model structure, we apply PCA on our MTS dataset in Figure 6-1 to reduce their dimension to a maximum of two channels. We implement the PCA algorithm from the 'R' software and highlight the most critical operation settings in Table 5-4. Figure 5-12 is a graphical result of the dimensionality reduction after applying PCA.

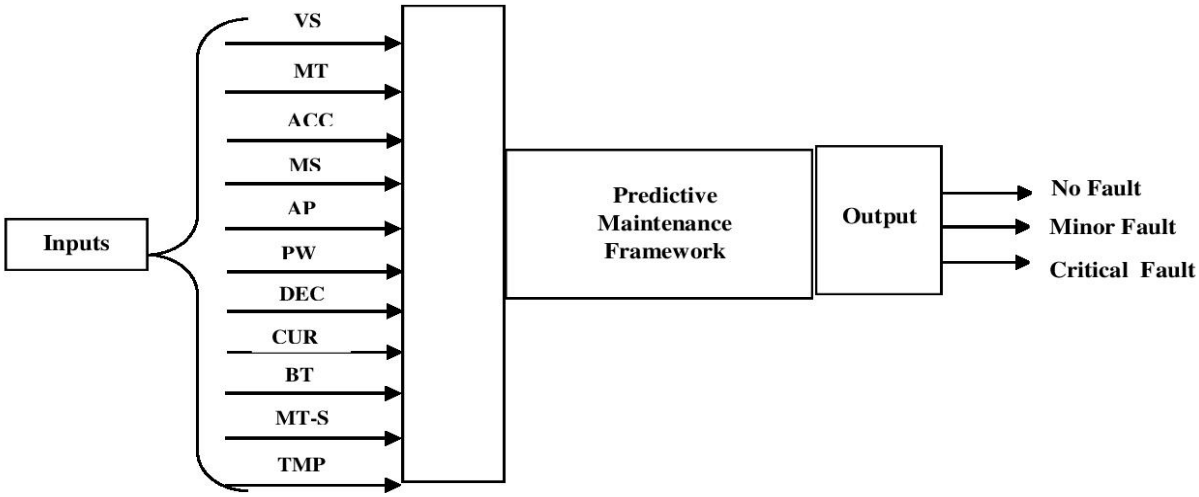

Figure 5-11: Intelligent PM Framework Inputs/Outputs architecture (Kiangala & Wang 2020 a)

Table 5-4: PCA settings (Kiangala & Wang 2020 a)

Settings	Value	Comment
SET.SEED	123	
SPLIT RATIO	0.8	Training and Test Set
METHOD	PCA	
PCACOMP	2	2 Channels used for MTS

5.3.2.1 PCA parameters study

As mentioned previously, we used the platform 'R' to compute the PCA operation. The parameters used in Table 5-4 are as follows:

- SET.SEED (123): The set.seed parameter is a random number inserted to ease reproducibility. It's not a compulsory parameter to use.

- SPLIT RATIO (0.8): The split ratio is the proportion of data allocated to the training set compared to the test (validation) set. A split ratio between 0.7 – 0.8 is recommended to achieve better performance, especially on large datasets (Rácz *et al.* 2021). The higher the training dataset ratio, the better chances are to reduce overfitting risks. The Split ratio is also sometimes represented as 2/3 in 'R'. Training our model at a split ratio value between b, $0.7 \leq b < 0.8$, did not have significant impact on the model performance (processing time or accuracy). We decided to keep to the higher recommended ratio (0.8) to reduce the risk of overfitting.
- METHOD (PCA): The method setting is the required parameter to implement PCA. We selected its value from the R library suggestion.
- PCACOMP (2): The PCACOMP setting represents the number of channels desired for the PCA reduction. We chose 2 in our model.

The PCA algorithm produces two new independent variables (the two reduced channels) that substitute the previous eleven variables in Table 5-3. We call the two variables PCAvar1 and PCAvar2. From the graph in Figure 5-12, we notice that the values of these new variables are different from the original raw data and will now be utilized in the remaining steps of our PM model.

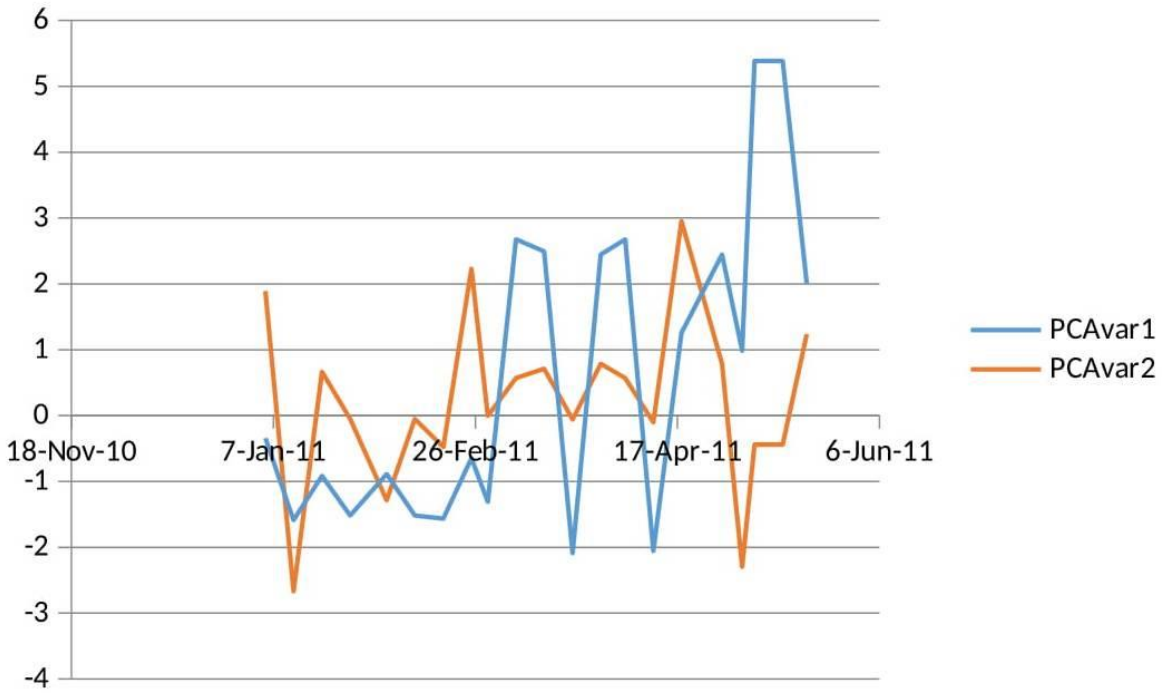

Figure 5-12: PCA reduced variables (Kiangala & Wang 2020 a)

5.3.3 Converting reduced time-series dataset into images

We load the two PCA variables values as displayed in Figure 5-12 into an image encoder to convert them into images. The image encoder section performs the following essential steps: (1) normalization of data, (2) conversion of normalized data into polar coordinates, and (3) transformation of polar coordinates into GAF images. We implemented the image encoding section in a Python IDE. Some of the most critical settings and steps for the image conversion are as follows:

- In Python, import the required GAF libraries from pyts.image
- Differentiate the various motor conditions from the PCA variables
- Load individual motor cases one at a time in the prepared GAF code
- The image_size setting is equal to 3
- Save the generated “summation” (GASF) and “difference” (GADF) images results of a dataset X from the first line item (0): X_gasf[0] to the last item (n) X_gasf[n] for the summation process; The same process is done for the “difference” process: from X_gadf[0] to the last item (n) X_gadf[n].

- The images need to be saved in separate folders for each motor condition.

We display samples of each motor condition at No-fault ('3') in Figure 5-13, Minor fault ('2') in Figure 5-14, and Critical fault ('1') in Figure 5-15, as generated by the GAF code for summation and difference schemes. Most critical faults in a conveyor motor system are caused by an imbalance when running the system at 50Hz. A combination of at least two faults could also result in a Critical state (misalignment and looseness together). Minor faults are usually a result of looseness or misalignment occurring individually.

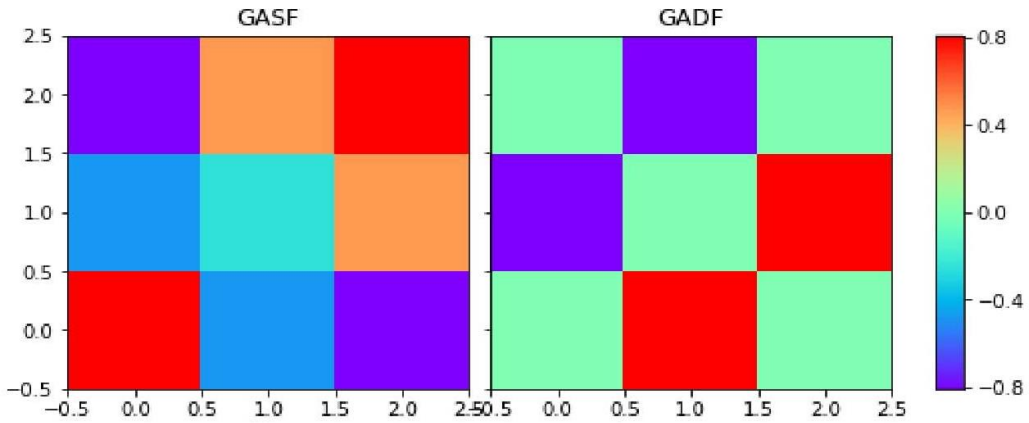

Figure 5-13: No Fault (NF)" motor condition sample on GAF images (Kiangala & Wang 2020 a)

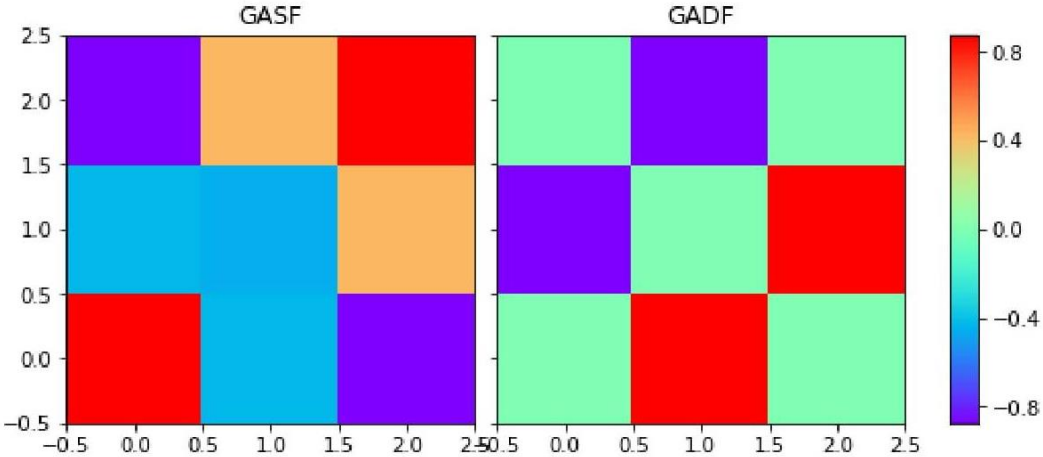

Figure 5-14: Minor Fault (MF)" motor condition sample on GAF images (Kiangala & Wang 2020 a)

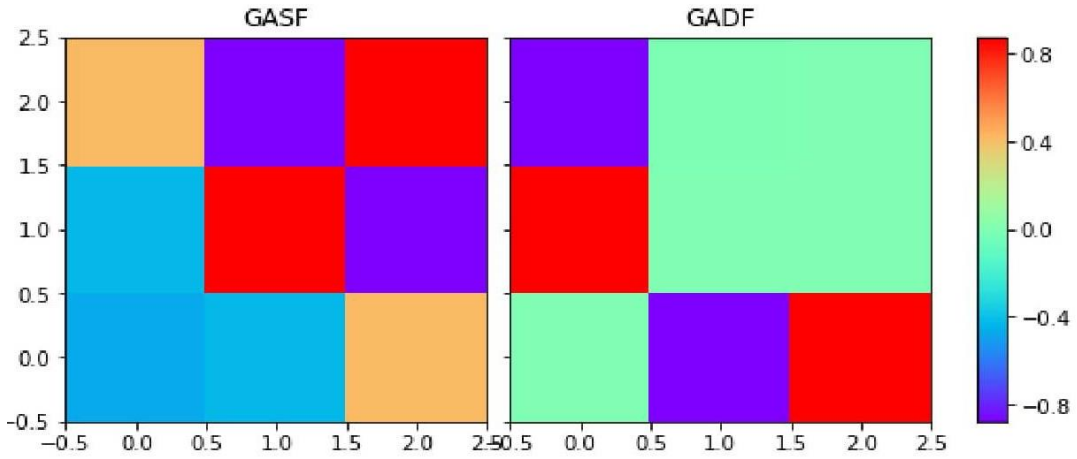

Figure 5-15: Critical Fault (CF)" motor status sample on GAF images (Kiangala & Wang 2020 a)

5.3.4 Fault classification model using CNN ML algorithm

We build a classification model using the CNN ML algorithm to categorize conveyor motor conditions as per the three states detected: No-Fault, Minor Fault, or Critical Fault. Our PM framework also offers an optional setting for more extensive networks and data by building another CNN section that applies the PReLU activation function instead of the ReLU standard one. We evaluate our results accuracy by computing three classification models:

- SVM: The inputs of this model are the two PCA variables.
- Standard CNN (using ReLU): The inputs of this model are the GAF images.
- CNN + PReLU: The inputs of this model are the GAF images.

We present in Tables 5-5 and 5-6 the essential parameters implemented for the computation of SVM and CNN ML models. The SVM classification model is created in ‘R’ and the CNN models in Python.

Table 5-5: SVM classification model settings

Settings	Value
Function	SVM
Type	C-Classification
Kernel	Linear

Table 5-6: CNN classification model settings

Parameters	Value	Comment
Split ratio	0.8	This corresponds to about 12,000 images in the training set and 3,000 images in the test or validation set
Library	Keras and tensorFLOW BACKEND	
Feature detectors size	(32, 3 , 3)	Since using a normal Central Processing Unit (CPU) instead of Graphical Processing Unit (GPU) we reduce the feature detector size to 32 to have less processing time.
Input shape	(64, 64, 3)	Since using a normal CPU instead of GPU we reduce the input shape to 64 to have less processing time.
Activation function	- ReLU (CNN + ReLU model), - PReLU (with alpha initializer = 0) for CNN+PReLU.	
Pool size under max pooler	(2, 2)	
Output dimension	128	Number of hidden nodes in the full connection section first layer
Output dimension	3	In the full connection last layer
Action function in the last layer	Softmax	
Compiling the model	Optimizer: "Adam" Loss: "categorical_crossentropy" Metrics: "Accuracy"	
Insert image data Generator to increase data size		

Extracting the training set images:	Size of the Target: (64,64) Size of the Batch: 32 Class mode: Categorical (3 classes to predict)	
Extracting the test (validation) set images:	Size of the Target: (64,64) Size of the Batch: 32 Mode of the Class: Categorical **Shuffle: False**	
Fitting the CNN model to training set and testing using the test set	Samples per epoch: 12,000; Number of Epochs: 3, number of validation samples: 3,000.	

5.3.4.1 CNN model parameters study

Selecting the appropriate hyper-parameter for the CNN model, as displayed in Table 5-6, is not a straightforward process. We achieve this selection by training our model with different parameters values and observing the model's overall performance. Going from default parameters (suggested by Python functions), we increased and decreased parameter values until we achieved the expected results. We performed our testing on a regular computer CPU (Intel R Core i3-4030U @1.90Gz – 8GB RAM). We observed the following:

- Split ratio (0.8): A split ratio between 0.7 – 0.8 is recommended to achieve better performance on large datasets (Rácz *et al.* 2021). The higher the training dataset ratio (compared to the validation set), the better chances are to reduce overfitting risks. We design our CNN model with a large dataset of at least 15,000 images. Training our model at a split ratio value between b, $0.7 \leq b < 0.8$ did not significantly impact the model performance (processing time or accuracy). We decided to keep to the higher recommended ratio (0.8) to reduce the risk of overfitting.

- Library (Keras and tensorFLOW BACKEND): These are the CNN python libraries for binary and classification models.

- Feature detectors size (32, 3, 3): The feature detector section has 3 parameters. The first one (32 in our model) is the number of filters available in the convolutional layer. The filter's role is to reduce the original image size for easy processing by keeping the most relevant features from the original image. The higher the number of filters, the longer the processing time. With a filter of size higher than 32 (tested for 64, 128, and 256),

our processing time went from about 3,431 seconds (for 32 filters and lower) to 4002 seconds (64), 4521 seconds (128), and 7532 seconds (256). We also observed that the higher number of filters did not impact the model accuracy. Few filters (tested for 8 and 16) do not significantly affect the processing time or accuracy. However, the number of filters of 1 reduced the accuracy considerably (to up to 33% only).

The second and third parameters of the feature detector are the size of the filter matrix, which for CNN can be a matrix of size 3x3, 5x5, or 7x7. Like some other researchers (Sun, Z. *et al.* 2016), our CNN model produces slightly better accuracy with a 3x3 filter matrix than any other one. We obtained an accuracy of 98.4% with 5x5 and 98.1% with 7x7. The change in matrix sizes did not affect the processing time.

- Input shape (64, 64, 3): The first two parameters in "Input shape" are the image dimension or resolution (width and height). The higher the resolution, the better the image looks but, the slower the processing time. In our CNN model, from an input image of 64 x 64 and lower (tested until 32 x 32), we obtained an overall processing time of 3,431 seconds. For higher image sizes (**tested for 128 x 128 and 256 x 256),** we recorded processing times of more than 7200 seconds. The choice of the image size parameter is a balance between the image quality and the processing time.

  The third parameter is a code for the type of image used (colour or black and white): '1' is the code for black and white images and '3' is the one for colour images. The number 3 symbolises the RGB (Red, Green, and blue) colours. We used '3' in our model since loading colour images.

- Activation function (reLU and PReLU): ReLU is the recommended activation function (Glorot *et al.*, 2011) to use in convolutional layers. We have explained the difference between reLU and PReLU in previous sections.

- Pool size under max pooler (2,2): The pooling parameter is a matrix of an arbitrary size used to reduce the size of the feature map matrix (matrix generated after applying convolution of the original image and the feature detector) and preserve a level of flexibility in the original image positioning. The max-pooling matrix of size 2 x 2 is the default and most popular used one in CNN modelling. Our model performed better with

the default max-pooling matrix size 2 x 2. We also tested our model with a different max-pooling matrix of size 4 x 4, for which the accuracy dropped to 96.3%.

- Output dimension (128): This parameter is the number of hidden nodes in the full connection section first layer. It is estimated to be 1% of the input shape parameter size: 64 x 64 x 3 = 12 288 (1% of 12288 ≈ 123). We used 128. The higher the number of hidden nodes, the longer the process time. We tested our CNN with an output dimension of 60 (0.5%) and 240 (2%). We observed a slightly reduced processing time of 3401s at 60 with the same accuracy of 100. However, at 240, we lost the accuracy to 33% with a processing time of about 3980 seconds.

- Output dimension (3): This parameter represents the number of neurons at the last output layer. In our CNN model, this number equals 3 because we are classifying 3 categories. For the binary CNN model, this number is equal to 1.

- Action function in the last layer (Softmax): It is the CNN action function parameter for classification models in Python (Sigmoid is the CNN action function for binary models).

- Compiling the model (Optimizer: "Adam" Loss: "categorical_crossentropy" Metrics: "Accuracy"): These are default parameters for CNN classification models in Python. More details are available in the code section (Appendix 1A).

- Insert image data Generator to increase data size: This is a parameter to increase the training and validation dataset to avoid overfitting. More details are available in the code section (Appendix 1A).

- Extracting the training set images Size of the Target: (64,64) Size of the Batch: 32 Class mode: Categorical (3 classes to predict): The target size should correspond to the input image dimension (64 x 64). Different size in this parameter causes an error. The size of the Batch represents the number of images in each Batch. The number 32 is a default value that works perfectly for our model. We tested the model with different

batch sizes (16 and 64) and observed that a batch of 16 images has a faster processing time 2044 seconds f about but an accuracy of 33%. A batch of 64 has a longer processing time of about 7,392 seconds and an accuracy of 98.11%

- Fitting the CNN model to training set and testing using the test set (Samples per epoch: 12,000; Number of Epochs: 3, number of validation samples: 3,000). We have explained the meaning of the Epoch in previous sections. The samples per Epoch is the number of the training dataset (images) we fed in our system. The number of validation samples is the number of test data we fed into the model.

During the pre-processing phase of our models, we divide the inputs data into two categories: a training set and a test set. The training set is used to teach and build the actual model, and the test set to assess the model's accuracy. The pre-processing data phase is done manually for the CNN models by separating training and test images in separate folders. The CNN models are built in a python platform using images generated in GAF and saved in different folders. For the SVM model, this step is achieved automatically by a pre-processing data code (See Appendices 1. A.).

We display the results of the three classification models in confusion matrices: Table 5-7 for SVM, Table 5-8 for CNN+ReLu, and Table 5-9 for CNN+PreLu**.** Using the confusion matrices results, we compute three other evaluation metrics: precision, recall, and accuracy for each model.

Table 5-7: SVM model confusion matrix results

	CF	**MF**	**NF**
CF	612	169	224
MF	149	181	706
NF	0	96	864

Table 5-8: CNN+ReLu model confusion matrix results

	CF	**MF**	**NF**
CF	1000	0	0
MF	0	1000	0

NF	0	0	1000

Table 5-9: CNN+PReLu model confusion matrix results

	CF	**MF**	**NF**
CF	1000	0	0
MF	0	1000	0
NF	0	0	1000

The green cells on each confusion matrix are the number of correct predictions achieved by the model for each motor condition: (1) CF: critical fault, (2) MF: minor fault, and (3) NF: no-fault. The remaining cells are the number of incorrect predictions. We summarize the evaluation metrics of all three models in Table 5-10.

Table 5-10: Classification models evaluation metrics results

Classification Models	**Label**	**Precision**	**Recall**	**Overall Accuracy**
SVM	CF	0.8042	0.6090	0.552≈ 55.2%
	MF	0.4058	0.1747	
	NF	0.4816	0.9000	
CNN+ ReLU	CF	0.1000	0.1000	0.100≈ 100.0%
	MF	0.1000	0.1000	
	NF	0.1000	0.1000	
CNN + PReLU	CF	0.1000	0.1000	0.100≈ 100.0%
	MF	0.1000	0.1000	
	NF	0.1000	0.1000	

5.4 Results interpretation and discussion

From the two confusion matrices of CNN models in Tables 5-8 and 5-9, we read impressive results of 100% positive predictions. We obtained this outstanding outcome by running up to

three epochs when training our models. An epoch expresses the number of times the CNN algorithm grasps the model features by its training dataset. The training and the testing accuracies percentages reached at the third epoch for the two CNN models are very close to just less than 1% (99, xx% - 100%). The closeness of accuracies at the three epochs for our two CNN models reflects lower chances of overfitting on our classification models.

The evaluation metrics results in Table 5-10 demonstrate that by applying CNN algorithm in a PM model, we increase the system's accuracy to about 50% more than in a PM model using a traditional ML algorithm such as SVM. We also obtain precision and recall of 100% when predicting each motor condition through our CNN PM model. A precision of 100% means that whenever our model predicts a motor's minor fault, it is always correct. The model does not predict false motors conditions (no false positives exist). Using an SVM model for which the precision is 40,58% would mean that the model can only successfully detect a motor's minor fault 40% of the time. In the remaining 60%, a minor's fault could be mistaken with a "critical fault" and cause unnecessary operation interruption or ignored for a "no-fault" condition and result in an unforeseen breakdown in the long run. We achieved a recall of 100% in the proposed CNN model, which implies that our model did not incorrectly predict any actual motor condition (no false negatives exist). On the SVM model side, recalls of 60, 17, and 90% mean that the ML model could incorrectly predict each motor's conditions (CF, MF, and NF) based on the percentage computed.

The impact of a 50% accuracy difference can be outrageous for a manufacturing plant affecting the performance and the system's availability. We illustrate the effects of the accuracies results of CNN and SVM PM models in a manufacturing plant using the Tecnomatix Siemens plant simulation software. The simulation settings are as follows:

- Computation time: 30 seconds
- CNN model machine availability 99%
- SVM model machine availability 56%

Figure 5-16 represents a simple simulated manufacturing structure where we test the impact of our PM model results. The manufacturing structure has the following sections:

- **A source**: that generates the parts being processed (represented by a brown box in Figure 5-16).

- **Three working stations**: where different manufacturing operations are done (represented by blue containers in Figure 5-16).

- **A drain**: that takes out the manufactured part from each station (represented by a green cage in Figure 5-16).

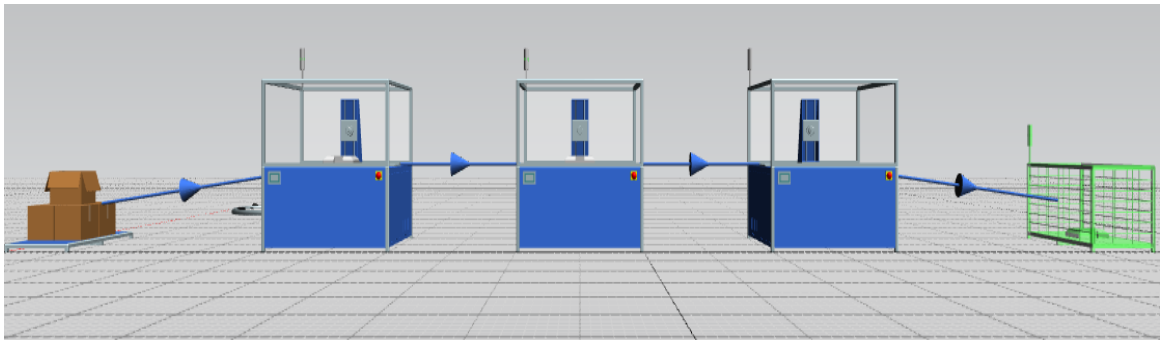

Figure 5-16: Simulated manufacturing model from Tecnomatix software

We configure the accuracy of 99% in our three stations and simulate for 30 seconds. We attain the following results:

- We operate at 73.30% at the drain section and are in a waiting mode at 29.95%. The drain fails at about 0.75%. The waiting time depends on the feeding state of the source and the performance at the three working stations. We reach a throughput of 8381,92 parts a day, 349.25 parts an hour, and 5.82 parts a minute. Figure 5-17 displays these results and some more statistics.

- We operate at 97.01% at the working station, the station is blocked for about 2%, and the station fails at 0.99%. Figure 5-18 shows the station statistical results.

- In all three working stations: Figure 5-19 summarizes the resource statistics results of all three stations using the PM with CNN, displaying the working, failed, waiting, and blocked conditions.

We configure the accuracy of 56% in our three working stations and compute the simulation for 30 seconds. We achieve the following results:

- We operate at 20.32% at the drain section and are in a waiting mode at 50.03%. The system fails at about 29.65%. The waiting time depends on the feeding state of the source and the performance at the three working stations. We reach a throughput of 1789.31 parts a day, 74.55 parts an hour, and 1.24 parts a minute. Figure 5-20 displays these results and some more statistics.

- We operate at 20.71% at the working station, the station is blocked for about 35.26%, and the station fails at 44.03%. Figure 5-21 shows the station statistical results.

In all three working stations: Figure 5-22 summarizes the resource statistics results of all three stations using the PM model with SVM ML, displaying the working, failed, waiting, and blocked conditions.

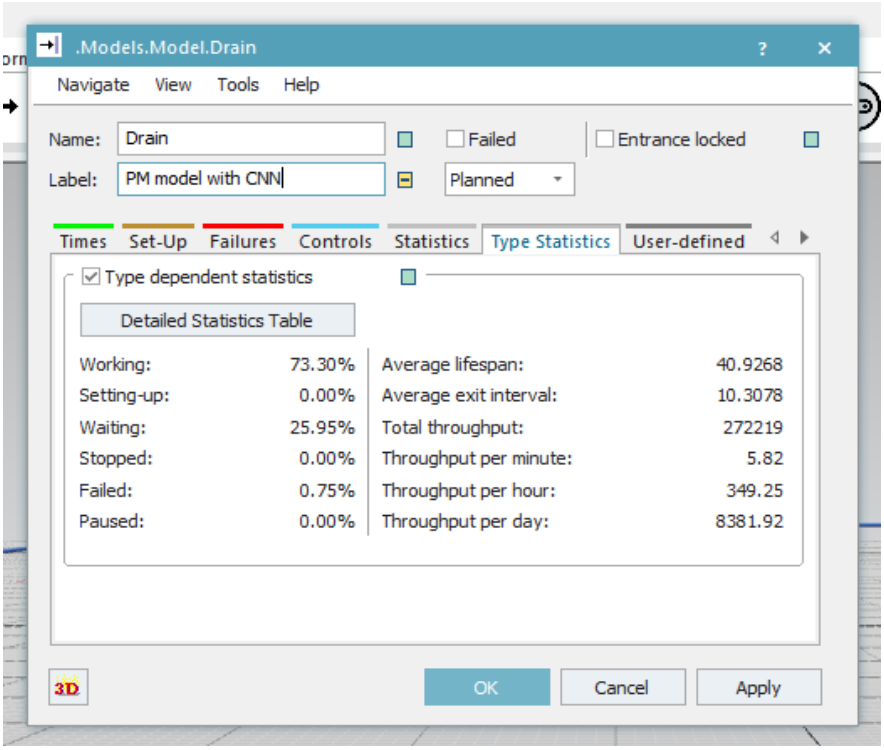

Figure 5-17: Drain section statistical results using PM model with CNN

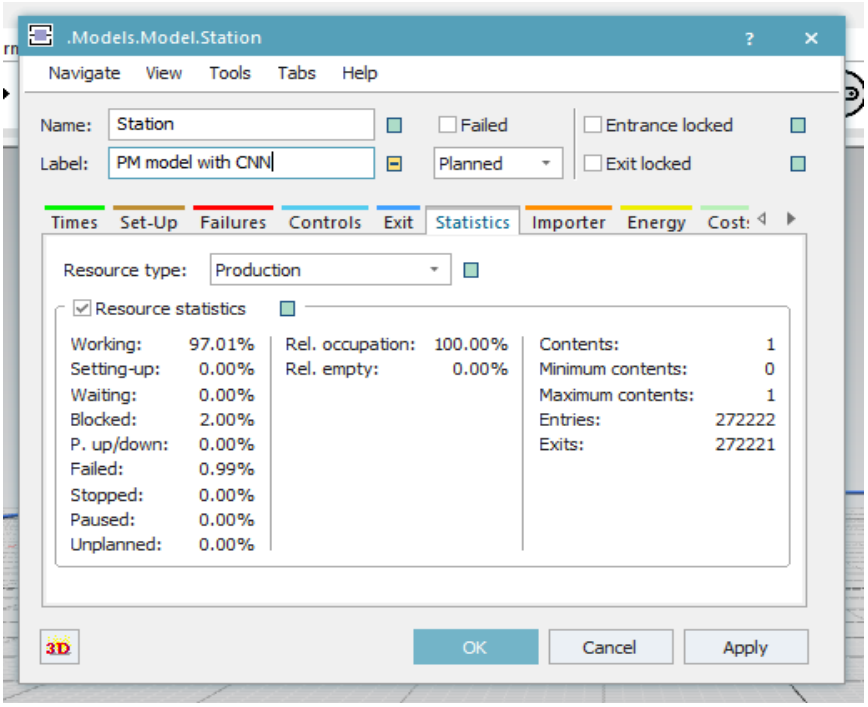

Figure 5-18: Working station statistical results using PM model with CNN

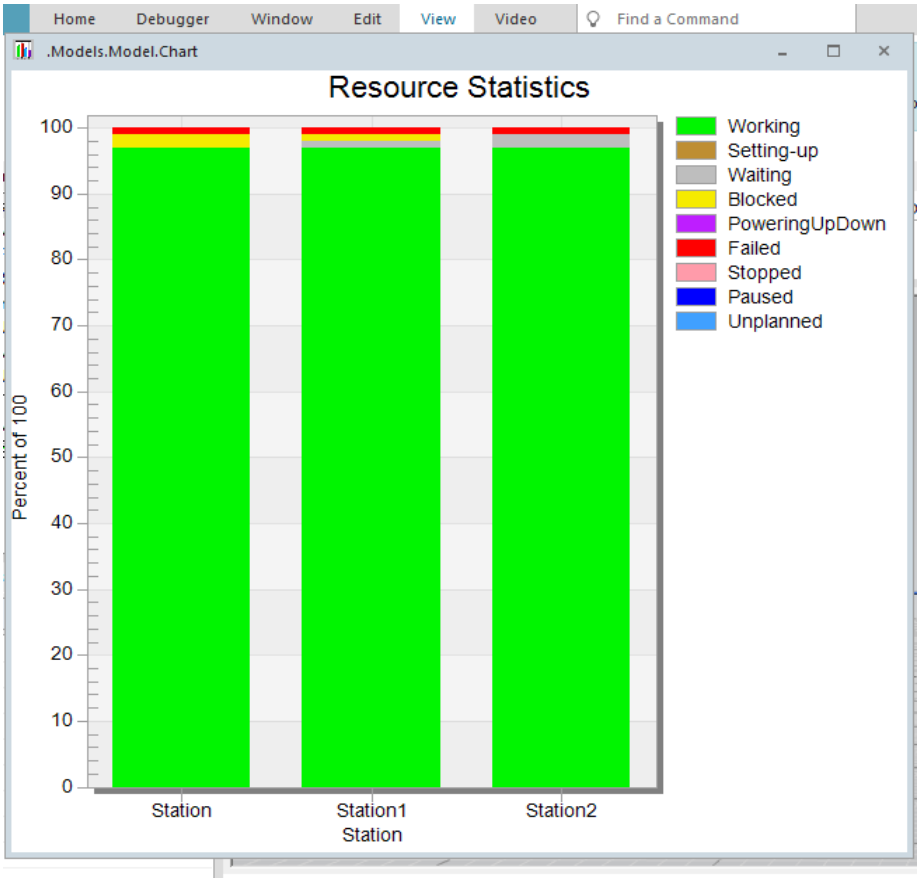

Figure 5-19: Resource statistics results using PM model with CNN

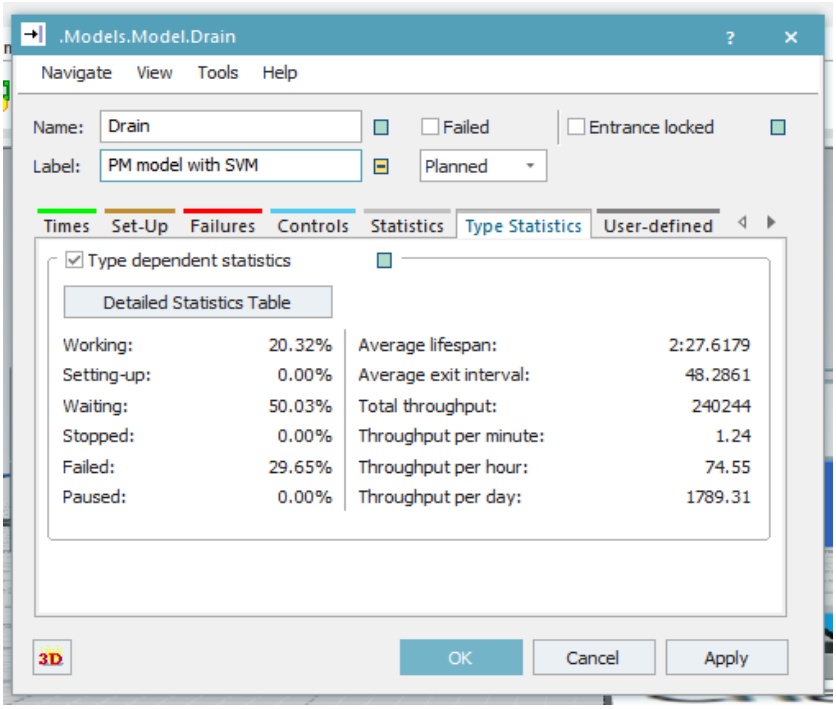

Figure 5-20: Drain section statistical results using PM model with SVM

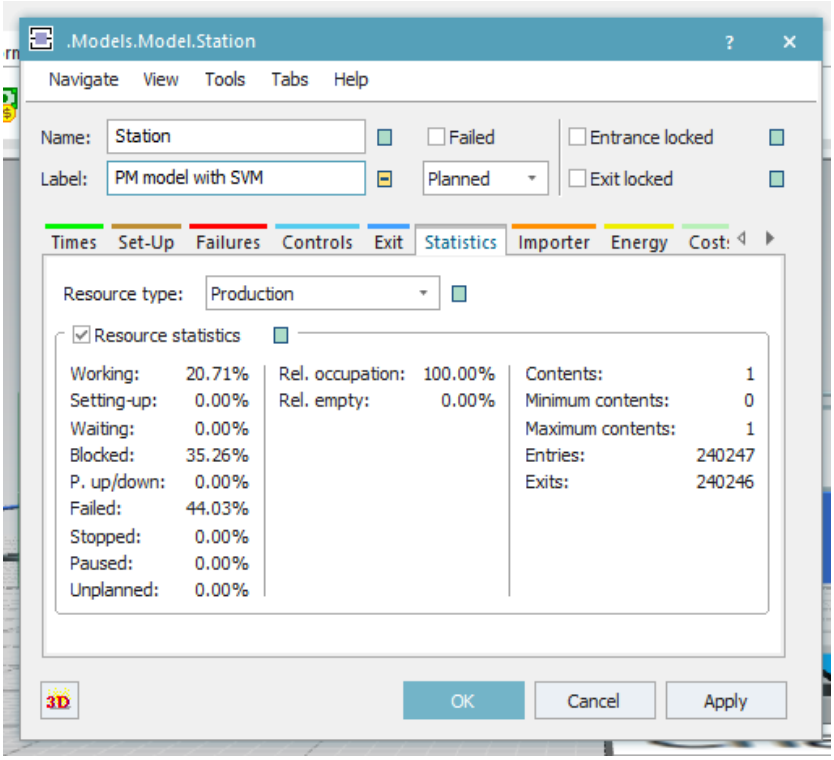

Figure 5-21: Working station statistical results using PM model with SVM

Limitations and user's guidance: The preparation and data pre-processing steps of a predictive maintenance system using our CNN method are tedious and demanding. It involves

collecting and converting data into images, as explained in previous sections, and the manual separation of input images into training and test folders. Once this most complex section is complete, the remaining tasks will be tuning ML parameters and loading new observations for the algorithm to adjust its outcomes and improve its performance. As guidance for users, we suggest that the operators or supervisors in charge perform the above activities during their maintenance or shutdown schedule, depending on the plant activities. We obtained similar accuracy on both CNN models (CNN+ReLU and CNN+PReLU) because we tested relatively small datasets. The difference would be noticeable on larger datasets where using the CNN+PReLU model would appear more advantageous. This option is future-proofed since upcoming manufacturing areas, especially in the I40 environment, are expected to deal with a more significant amount of data in their premises.

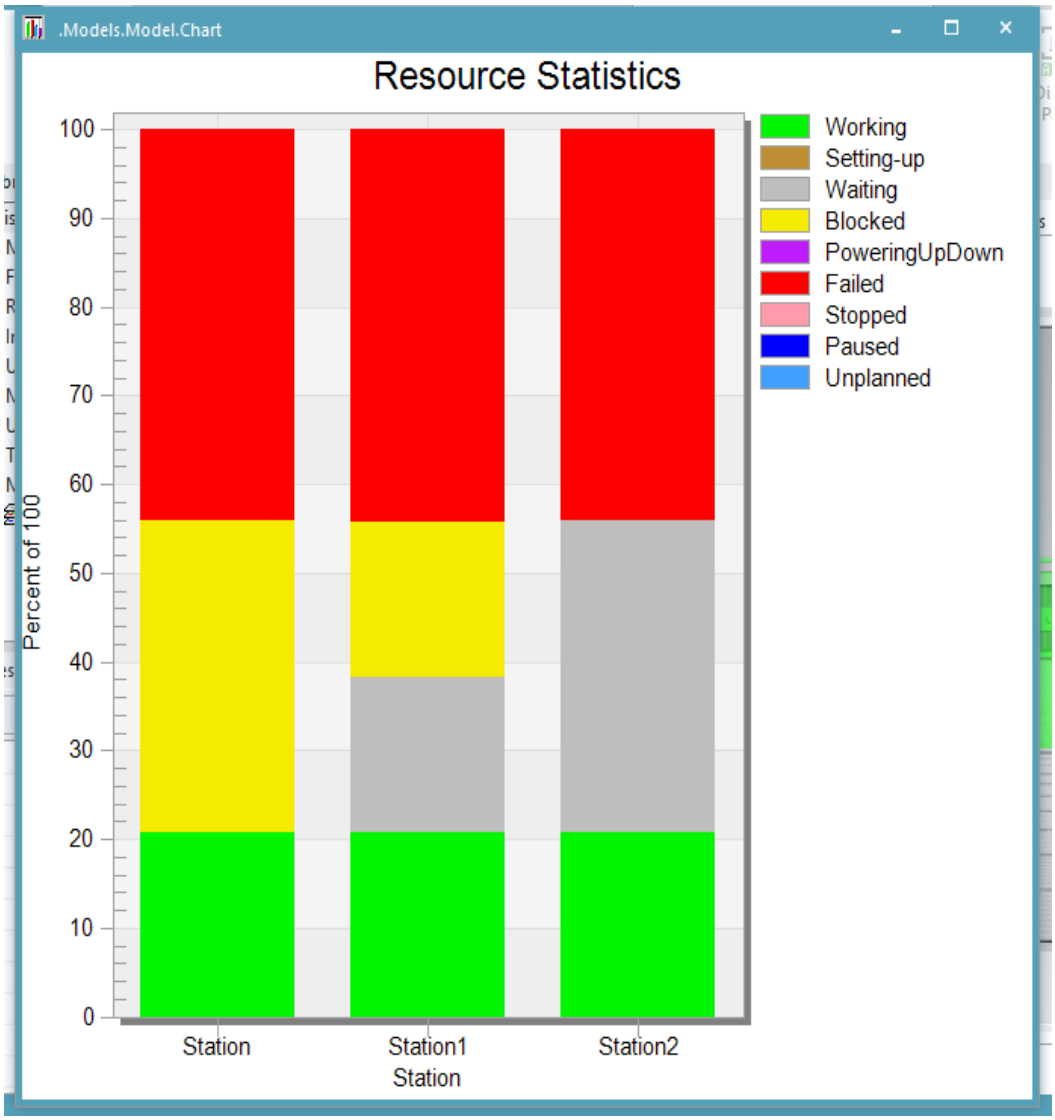

Figure 5-22: Resource statistics results using PM model with SVM

5.5 Chapter Summary

This chapter described, designed, and implemented an intelligent predictive maintenance framework for a conveyor motor in a small manufacturing environment using CNN. We started with a theoretical overview of CNN and all required algorithms and techniques (PCA and time-series imaging) required to design our intelligent PM framework. We modelled our PM system using the theoretical knowledge gained in the previous section. We implemented the modelled PM framework with different ML models designed in programming software like Python and R and tested its results. We concluded by analysing the significance of our intelligent PM platform results, in Tecnomatix Siemens plant simulation software, compared to a PM system developed with other ML algorithms such as SVM.

Chapter 6 : AN IMPROVED NETWORK INFRASTRUCTURE WITH ADVANCED COMMUNICATION TECHNIQUES AND REDUNDANCY PROTOCOLS

Although various preventives procedures and actions such as predictive maintenance are applied to ensure network availability and to lessen production downtime, unforeseen faults of network components like cabling and switches are almost inevitable. In order to palliate this concern, we design a communication prototype with specific network topologies that allows the application of zero-loss redundancy protocols to create back up communication channels for data in case of unplanned faults. The zero-loss redundancy protocols also provide fast recovery time with the most negligible downtime. Our prototype composition includes some state-of-the-art communication computing technologies in the likes of TSN and edge computing to achieve a more sustainable industrial network with fewer communication delays and more determinism. Our communication prototype is suitable for IIoT time-critical applications. The communication theories from which we build our network prototypes are TSN, edge computing, and network communication topologies.

6.1 Network concepts and infrastructure theoretical review

6.1.1 Time-Sensitive networking (TSN)

The name TSN refers to an IEEE 802.1 task group (TG) designated by the IEEE to develop several new industrial communication standards that will solve some of the current Industrial Ethernet limitations and be suitable for IIoT communication requirements in an I40 environment (TTTech. 2015). TSN ensures determinism in data communication and fixes issues of non-reliable transmission by applying principles such as bandwidth reservation, traffic shaping, and precise clock synchronization (Tian & Hu 2019 and Fu *et al*. 2018). TSN has a group of operational standards (Gutiérrez *et al*. 2018); we present some in Figure 6-1. TSN intends to become a standardized networking technology that substitutes existing industrial Ethernet protocols used in the traditional automation pyramid.

We design our industrial network communication prototypes with switches supporting TSN standards to achieve a delay-free data transmission for time-critical applications. Figures 6-2

and 6-3 represent the frame transmission of time-critical data in two successive transmission cycles without and with TSN-capable network devices (Fu *et al*. 2018).

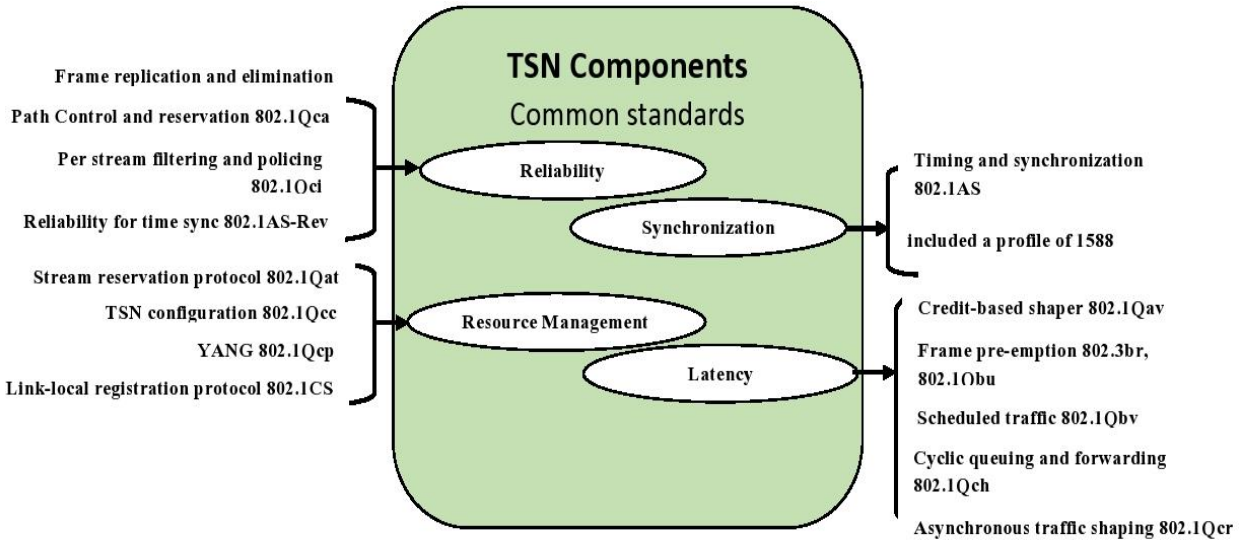

Figure 6-1: Some finalized TSN standards

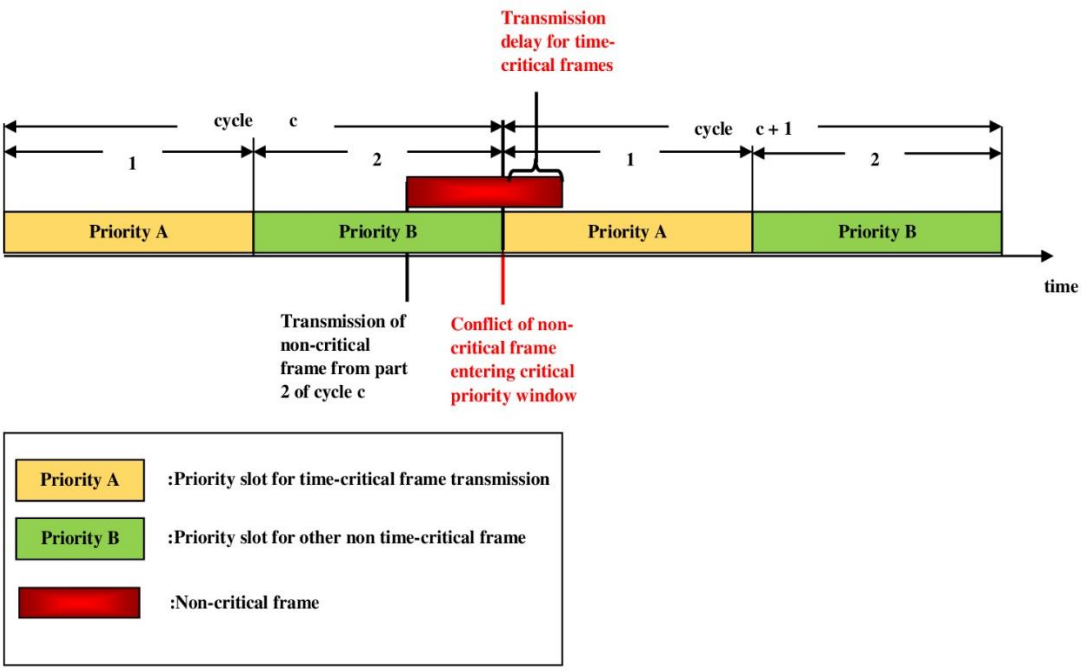

Figure 6-2: Frame transmission in network devices with no TSN capabilities

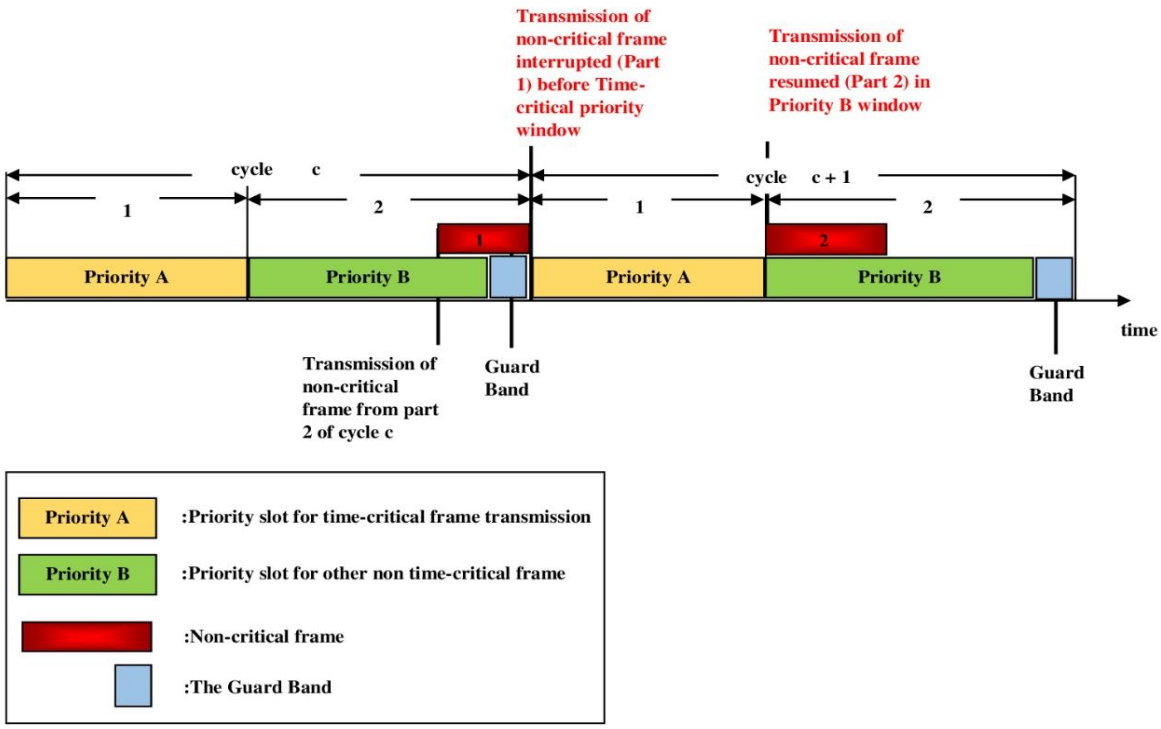

Figure 6-3: Frame transmission in TSN-capable network devices

In standard Ethernet data communication, frames have different priorities from 0 to 7, the importance of a frame during its transmission rest on its priority. However, network devices with no TSN capabilities have no mechanisms to stop the transmission of non-time-critical data during a time-critical priority window. In standard Ethernet communication, data forwarding is not interrupted unless interrupted at the physical communication layer. From Figure 6-2, we can see that the transmission of a non-critical frame in a priority window can create unnecessary delays for time-critical frames waiting in the network devices buffer queues (Fu *et al*. 2018 and Craciunas *et al.* 2016).

Figure 6-3 represents a frame transmission in a TSN-capable device using mechanisms such as the IEEE 802.1Qbu Frame Pre-emption and the IEEE 802.1Qbv Guard Bands Mechanism; Non-time-critical frames do not intrude into the priority window. The guard band halts a non-critical frame and retransmits it only after the time-critical window. Hence, time-critical frames will travel at ease and will not experience a transmission delay.

6.1.2 Edge computing

The ascent of the 4IR promises the creation of more intelligent, responsive, interconnected, and self-optimizing production systems via several innovative techniques (Lohan *et al*. 2018). Manufacturing factories should utilize more machine-type devices (MTDs). MTDs are autonomous devices performing several duties like monitoring, billing, and protection (Ali *et al.* 2018 and Musaddiq *et al.* 2018) on their own with minor human intervention. Tasks they perform that implies exchanging a more considerable amount of data between themselves and with more advanced structures such as cloud systems. MTDs contains a networking connection, an application section, and a sensing area.

Current cloud computing technologies performing advanced duties such as data analytics are in remote locations far away from MTDs, controllers, and end devices in manufacturing systems. The amount of data exchanged between remote clouds and factories often causes non-reliable connections, network congestion, and unacceptable latencies (Islam *et al.* 2014) that are unacceptable in the I40 environment. The concept of Edge computing brings solutions to some of these shortcomings by allowing intelligent services, data processing, and analytics closer to the manufacturing floor in edge servers. Edge servers have more considerable storage capabilities than MTDs. They also offer networking and computing abilities, which means they can participate in IIoT communication networks and applications. Edge computing facilitates agile connection, data analytics at edge nodes, and privacy strategies (Chen *et al*., 2018). Edge servers become the bridge between manufacturing factories and cloud servers for specific tasks requiring advanced cloud services. The edge computing technology relieves MTDs from limited computational operations or storage requirements and provides a future-proofed solution for large industrial networks in an I40 era (Liao *et al.* 2020). We present in Figure 6-4 a graphical representation of the edge computing operational principle.

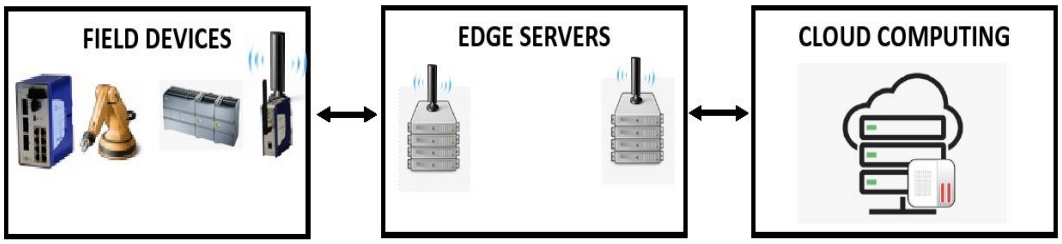

Figure 6-4: Edge computing operational principle (Qi & Tao 2019)

6.1.3 Network communication topologies

The implementation of adequate network infrastructure is essential to achieving an effective industrial manufacturing environment. It facilitates guaranteed data exchange between all production stakeholders. We describe in the following lines different network topologies for industrial communication networks (Vitturi *et al.* 2019):

- **A line or bus network topology**: This network topology consists of network devices such as switches connected in a line one device after another. We display in Figure 6-5 a representation of a bus network topology.

Figure 6-5: Bus (line) topology

- **A ring network topology**: The ring topology is a circle-looking connection of network devices. This topology is intensively implemented in industrial networks. The ring topology is a line topology connected by the first and last network devices. We present a ring network topology in Figure 6-6.

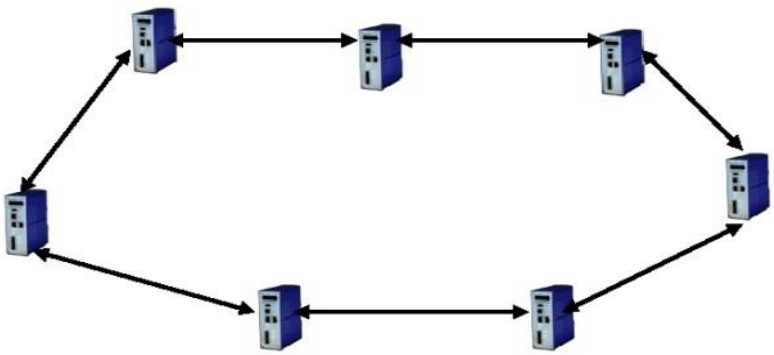

Figure 6-6: Ring topology

- **A mesh network topology**: The mesh network topology consists of multiple connection links between network devices. Each network device has one or more connections from one device to another. Figure 6-7 is a graphical representation of the mesh network topology.

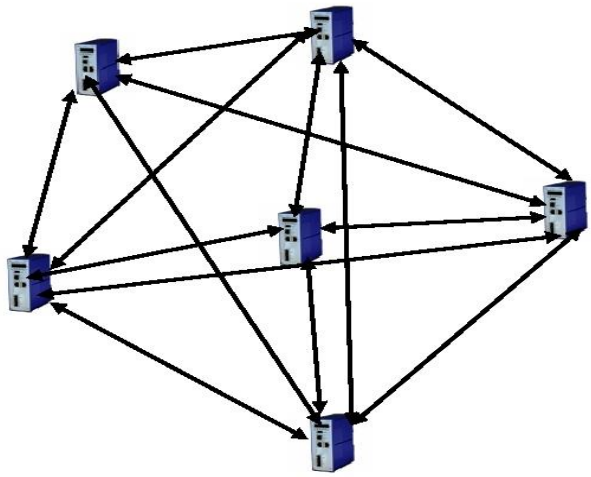

Figure 6-7: Mesh topology

- **A star network topology**: A star network topology usually has a single main switch to which all other end devices connect to interact with each other. The main switch is the single point of contact of all other network devices. This network topology is very popular in small traditional IT networks. Figure 6-8 illustrates a star network topology.

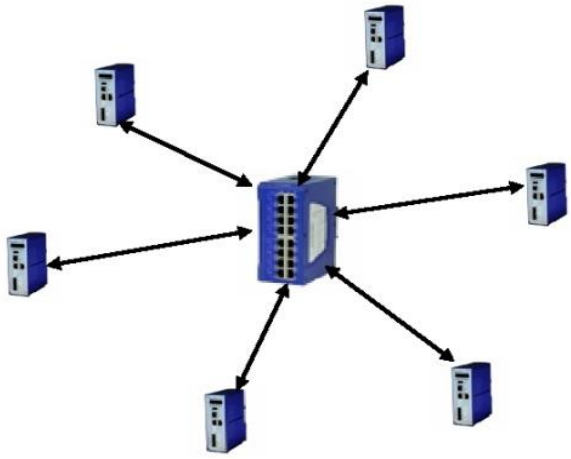

Figure 6-8: Start topology

It is crucial to note that some network topologies such as the mesh and the ring topologies require specific protocols activation in network devices (network switches) to operate effectively. Without the appropriate protocol, the communication network will create a loop with no possible communication between devices. One of the oldest well-known protocols created to avoid network loops is the spanning tree protocol (STP) (Khoshnevisan *et al.* 2019).

6.2 Modelling of an improved network infrastructure using advanced communication concepts and zero-loss network redundancy protocols

Network components failures cause communication downtime that is inadmissible in time-critical industrial manufacturing applications. The implementation of redundancy protocols is one of the responses to this issue to lessening the impact of delays due to network components breakdown. The selection of the appropriate redundancy protocol rest on the network topology applied (Prinz *et al.* 2018). Some popular network redundancy protocols in industrial networking are:

- **Rapid Spanning Tree Protocol (RSTP):** RSTP is a redundancy protocol developed to produce better outcomes than the previous spanning tree protocol (STP). It offers an improved recovery time after network failures and allows more network devices in the topology. RSTP is also known for its ability to prevent loops in Ethernet networks. The RSTP is quite flexible in terms of the kind of network topologies it supports. However, its recovery time relies on the network device position: the furthest the network device, the longer the recovery time. Therefore, the RSTP is not very reliable for time-critical industrial applications that do not tolerate communication delays.

- **Media Redundancy Protocol (MRP)**: MRP is another industrial redundancy protocol mainly design for ring topologies. It provides better recovery time than the RSTP in ring topologies and is a high-availability redundancy protocol for Industrial Ethernet networks. When correctly implemented, the MRP can handle up to fifty network devices (switches) in a ring and should achieve a recovery time of 500ms maximum in a worst-case scenario. In an MRP ring regular operation, one of the communication

links called the redundant link stays in a “blocked” state until a fault occurs in the network. The redundant link changes from the “blocked” state to an “active” state and can forward packets. One switch is permanently configured in an MRP ring as a ring manager to control and monitor the network redundancy process. The ring manager controls the activation and deactivation of the redundant link.

Even though the MRP recovery time is usually acceptable for most industrial network applications, it still appears unsatisfactory for time-critical applications that do not allow communication delays or downtime. For these kinds of applications, researchers developed zero-loss redundancy protocols (Hirschmann 2014).

- **Parallel Redundancy Protocol (PRP):** PRP is a zero-loss redundancy protocol developed in the International Electrotechnical Commission-IEC 62439-3 standard. The network device supporting PRP should directly connect to the network topology's time-critical end devices. The other network switches can have different types of redundancy protocols running in them. The PRP operation process consists of forwarding duplicated packets or frames from a PRP-capable device to a destination (PRP-capable equipment) via the two independent sides of the network (LAN A and LAN B). End-devices supporting PRP are Double-Attached Node supporting PRP (DAN P), and switches that support the PRP are called Redundancy Box (RedBox). The destination PRP-capable device processes the frame that arrives first and rejects the second one. In the network topology, LAN A and LAN B utilize different redundancy protocols like MRP or RSTP. The advantage of sending duplicated frames in two independent networks is to maximize the chances of at least one frame reaching its destination in case of a network breakdown**.** We display in Figure 6-9 an illustration of PRP operation (Araujo *et al.* 2015).

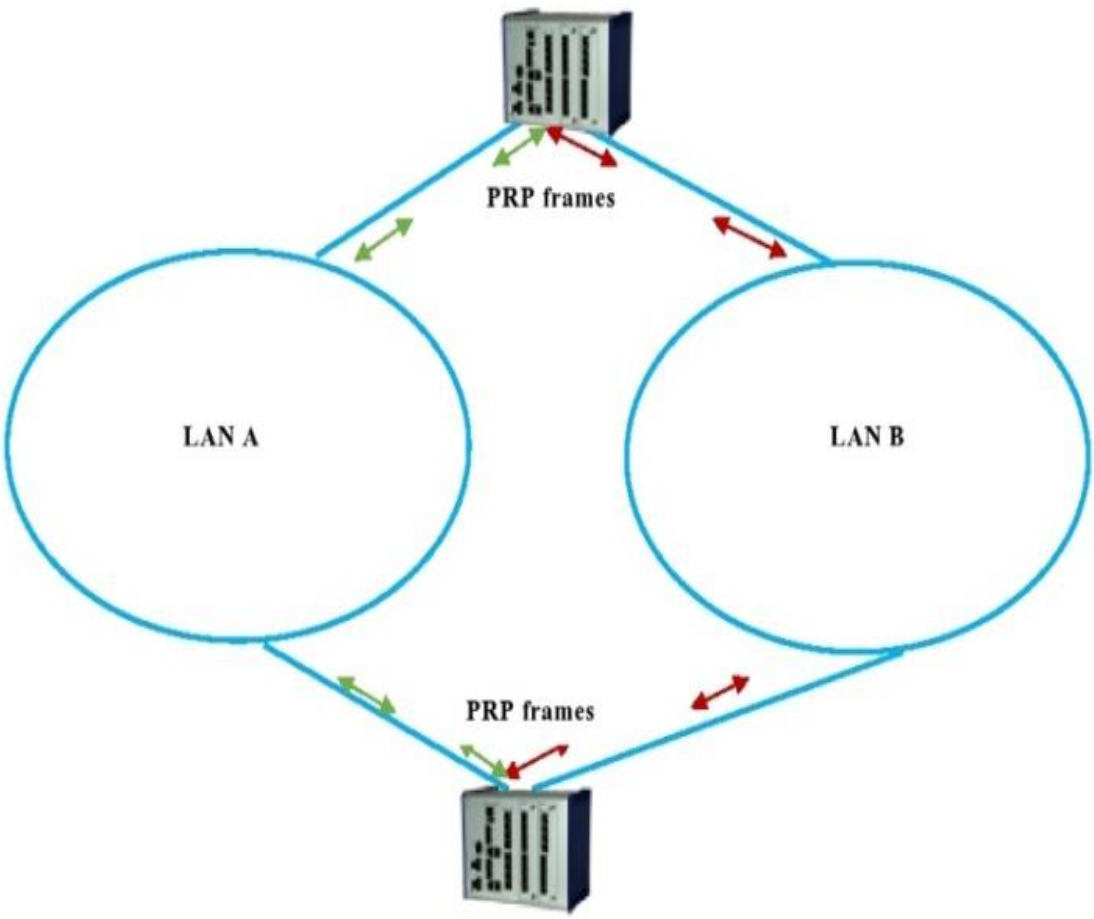

Figure 6-9: PRP network operation

- **High-Availability Seamless Redundancy Protocol (HSRP):** HSR is another zero-loss redundancy protocol in the IEC 62439-3 standard. Like PRP, its operation lies in forwarding two identical frames through the network to achieve a swift delivery of packets to the destination, especially in case of network components failures. The main difference between PRP and HSRP is the network topology in which they operate. The HSR protocol is only suitable for a ring topology supporting a maximum of 512 network devices that should all be HSR-capable devices. The same frame processing principle applies in HRS whereby the HRS-capable device accepts the first arriving packet and discards the second one. Devices supporting the HRSP are Double Attached Node supporting HSR (DAN H) (Hirschmann, 2014 and Araujo *et al.* 2015). We present in Figure 6-10 an illustration of HSRP operations.

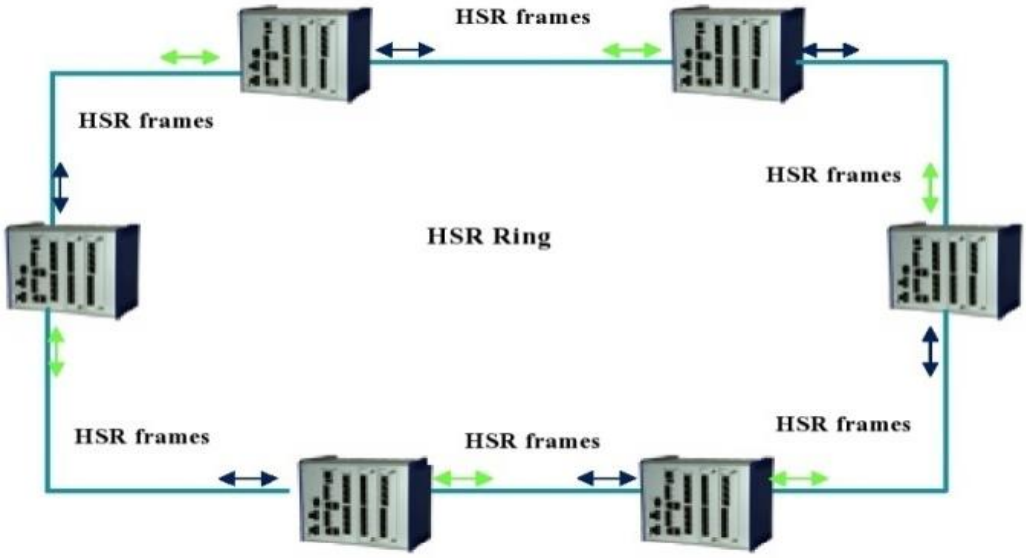

Figure 6-10: HSR network operation

Using the knowledge of zero-loss redundancy protocols and their topologies, we design two network communication prototypes where we consider network devices (switches) enabled for advanced communication concepts such as TSN. We also incorporate in our prototypes design the notion of edge computing to ensure low-latency communication in the factory when using intelligent facilities like the cloud.

We summarize in Table 6-1 different redundancy protocols and their principal attributes for better network design.

Table 6-1: Network redundancy protocols attributes

Redundancy Protocol	**Network topology**	**Max. Number of Switches**	**Recovery time**
RSTP	Ring	40	> 2 s
RSTP	Mesh, Start, Any other	Infinite	> 2 s
MRP	Ring	50	500 ms
PRP	Double networks	Infinite	0 ms

6.2.1 Assumptions

We assume that all network switches utilized in the communication prototype support the TSN protocol. They are TSN-capable devices.

6.3 Implementation of effective industrial communication prototypes

We design two network communication prototypes based on a few of the current powerful physical and software communication methods that favour reliable communication systems.

6.3.1 Physical communication methods

We implement two redundancy protocols called “zero-loss” because of their ability to perform communication recovery and enable backup routes at 0ms recovery time. Hence, it reduces the risk of communication delays and latencies in case of hardware components failure. The two zero-loss redundancy protocols are PRP and HSR. We present a communication prototype built based on the PRP protocol in Figure 6-11 and another one based on the HSR protocol in Figure 6-12.

From the two communication prototypes, we notice that network equipment supporting the PRP and HSR protocols (switches) transmits duplicated data (frames) at two different sides of the network from two separates communication ports. By doing so, the forwarded information has more chances to reach its destination in a failure in one part of the network.

- **The PRP communication prototype:** The PRP network contains two PRP switches (devices A and B) that forward and receive identical frames that travel throughout the network. The duplicated frames go through two separates ring topologies in which we apply the MRP protocol (MRP ring 1 and MRP 2). Each MRP ring has three switches, and one of the switches is configured as the RM to control the link status between each switch. Whenever one of the MRP rings experience a fault, the other duplicated frame

will still proceed independently to the destination. We summarize the functions and attributes of PRP network switches in Table 6-9**.** Time-critical devices requiring minor communication delays should be connected directly to PRP switches A and B.

- **The HSR Communication prototype:** The HSR network is quite simple, it is connected in a ring topology, and every ring device is HSR capable of transmitting and receiving duplicated frames. The non-HSR devices are connected as end devices to HSR switches.

Table 6-2: Switches attributes in PRP network prototype

Switches	Redundancy protocol	Other Attributes
A	PRP	TSN capable
B	PRP	TSN capable
C	MRP (ring 2)	TSN capable
D	MRP (ring 2)	TSN capable
E	MRP (ring 2)	TSN capable, Ring Manager
F	MRP (ring 1)	TSN capable, Ring Manager
G	MRP (ring 1)	TSN capable
H	MRP (ring 1)	TSN capable

MRP is one of the preferred ring redundancy protocols since it offers a better recovery time than STP or RSTP. We tested the recovery time of an RSTP ring in Figure 6-13 versus an MRP ring in Figure 6-15. From Figure 6-13 in the RSTP ring network, the redundant link is displayed by dotted lines between switches with IP addresses 172.16.4.3 and 172.16.4.4. This link remains blocked in everyday operations to avoid network loops and gets activated when the other network links are down. We monitored the recovery time of the ring using the MPING LCD tool. The tool is set up to measure the recovery time between the hosting PC with IP address 172.16.4.205 and the switch 172.16.4.6. We disconnect one of the network links to simulate a fault in Figure 6-14. The redundant link becomes a solid line, which means that the link is active. We measure a recovery time of about 1s:940ms from the MPING LCD tool. The recovery time depends on the size of the network and the type of switch processor. A network with fewer switches will experience a lower delay than on a more extensive network.

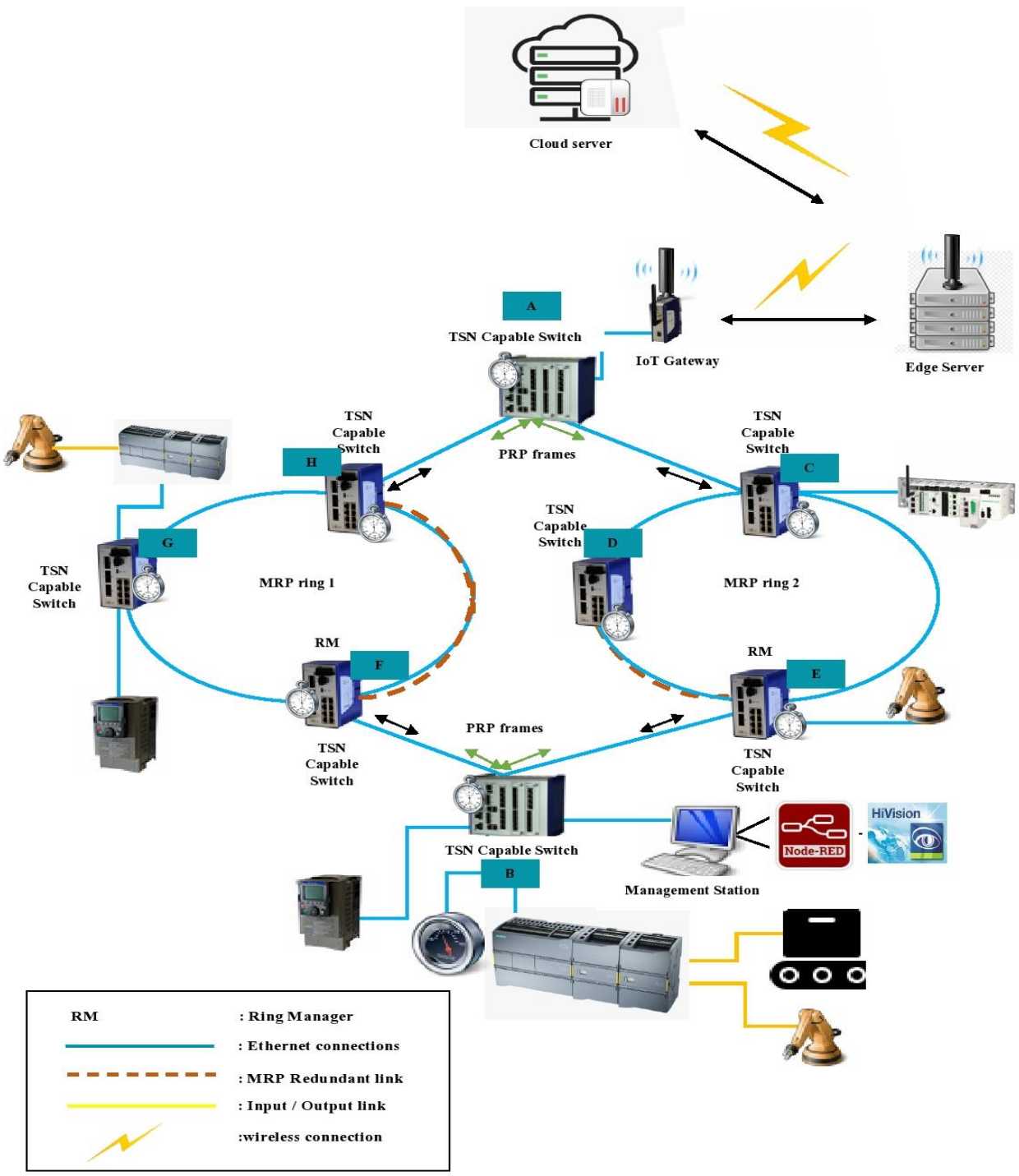

Figure 6-11: Industrial network communication prototype based on PRP

From the MRP ring network in Figure 6-15**,** we observe the redundant link in dotted lines. In an MRP ring, the redundant link is always next to the RM (switch with IP address 172.16.4.1). The exact recovery process as in an RSTP ring applies to MRP. When one of the links is down, the redundant connection becomes active and takes over the data transmission. We display, in Figure 6-16, the recovery time recorded in the MRP ring after a faulty link. The recovery time is about 40ms for a small ring network.

From the MRP ring network in Figure 6-15, we can estimate the frame travelling time using TSN and non-TSN-capable devices. For a frame of size 800 bytes travelling from the switch 172.16.4.1 to the destination 172.16.4.6 via the devices 172.16.4.2, 172.16.4.3, 172.16.4.4, and 172.16.4.5, the frame forwarding time in TSN enabled switches can be calculated as follows:

$$\delta = \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} = 5\frac{800\times8}{100\times10^6} = 320\mu s$$

Considering non-TSN capable switches, the same transmission time for the same frame size can be estimated as:

$$\delta = \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \frac{800\times8}{100\times10^6} + \delta_{mry1} + \delta_{mry2} + \delta_{mry3} + \delta_{mry4} + \delta_{mry5}$$

$$= 5\frac{800\times8}{100\times10^6} + \delta_{mry1} + \delta_{mry2} + \delta_{mry3} + \delta_{mry4} + \delta_{mry5}$$

$$= 320\mu s + \delta_{mry1} + \delta_{mry2} + \delta_{mry3} + \delta_{mry4} + \delta_{mry5}$$

where δ_{mry} is the delay of frames in each switch memory defined in (2.7).

The frame transmission time in non-TSN enabled switches depends on the number and the size of frames waiting in switches' memories during the transmission.

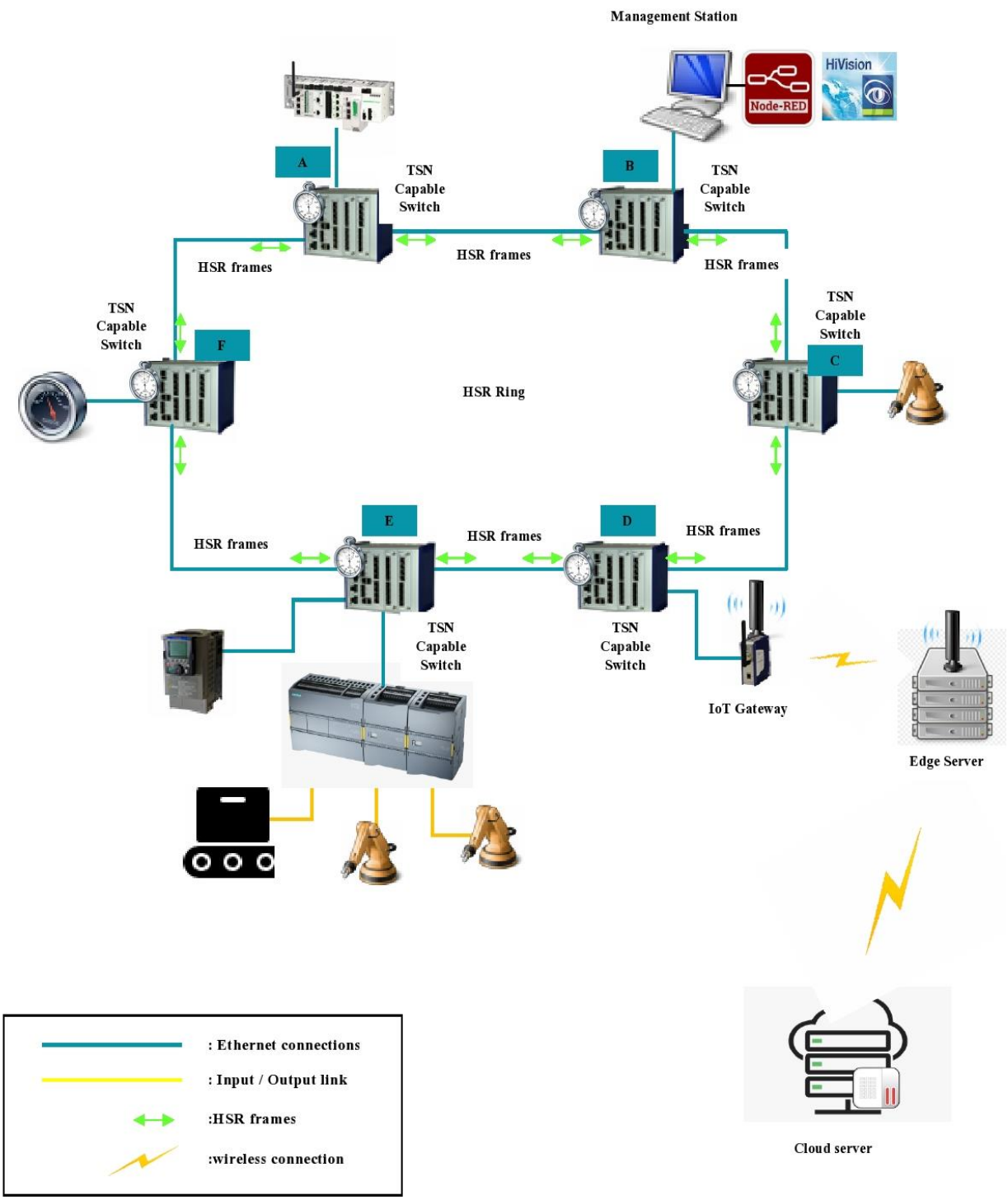

Figure 6-12: Industrial network communication prototype based on HSR

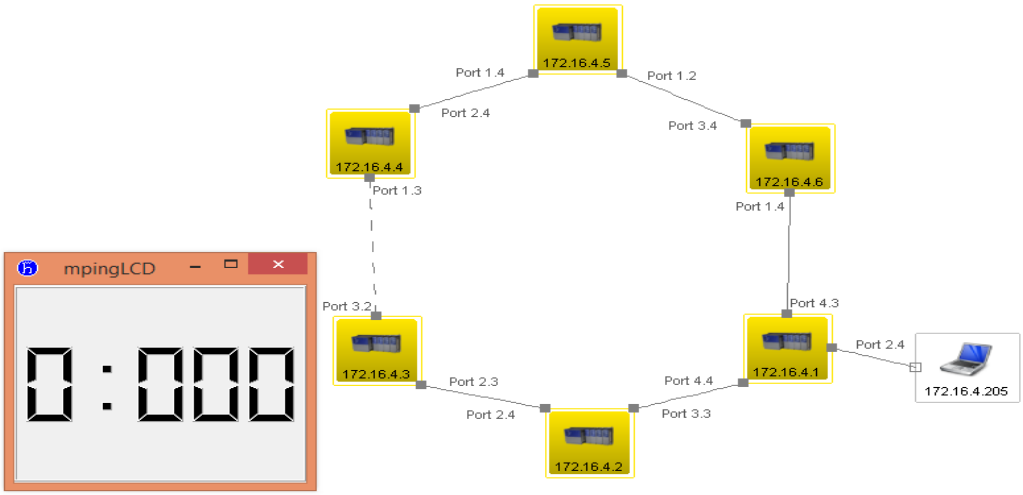

Figure 6-13: RSTP ring network with no link failure

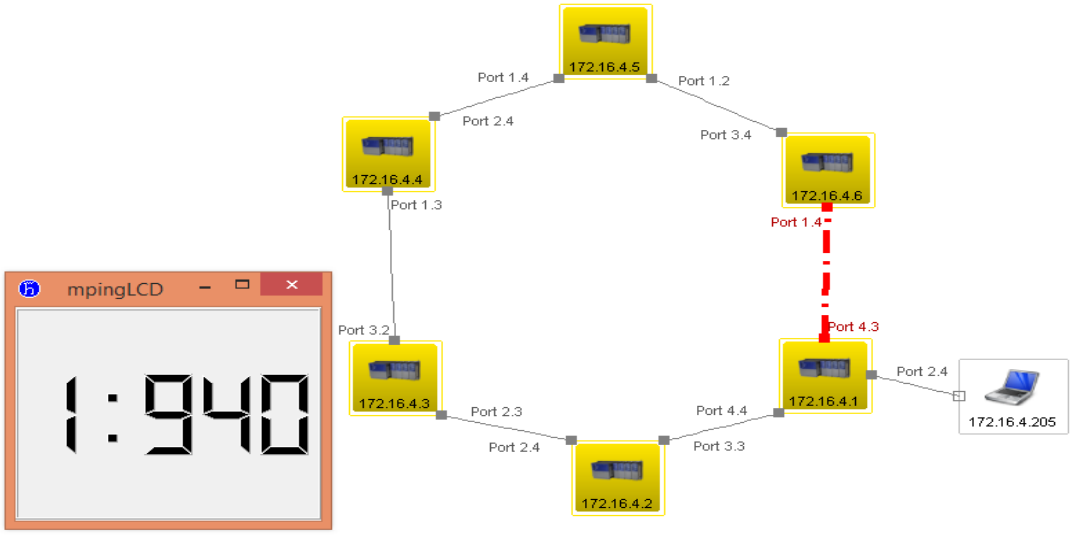

Figure 6-14: RSTP ring network with link failure

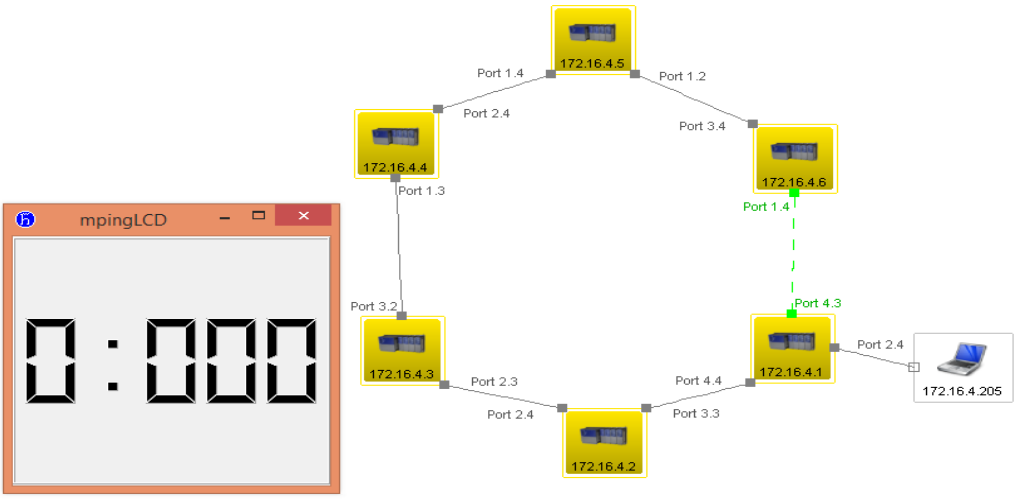

Figure 6-15: MRP ring network with no link failure

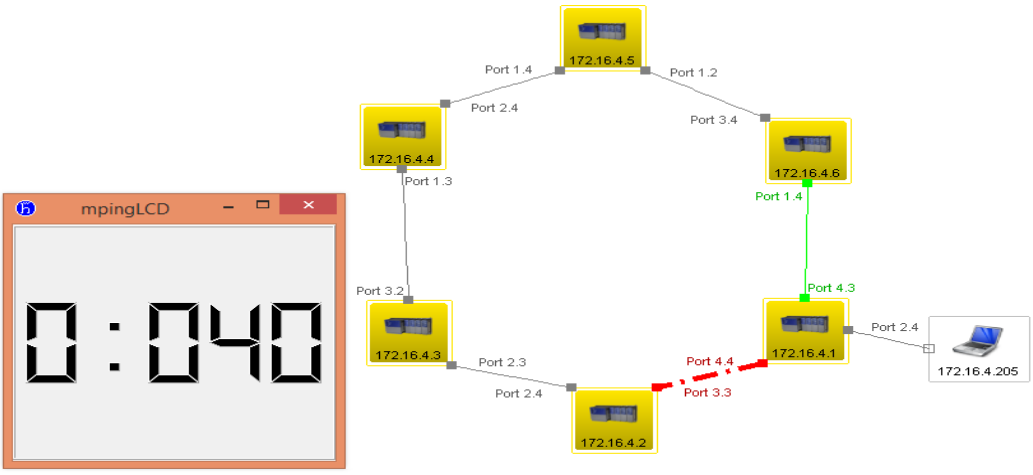

Figure 6-16: MRP ring network with link failure

6.3.2 Software communication methods

From software communication methods, we incorporate the edge computing technology to reduce the risks of network latencies and network bandwidth usage due to the massive size of

data transmitted from the physical network (at the manufacturing plant by controllers, IIoT devices, and field devices) to remote cloud facilities for further data processing. From our two communication prototypes in Figures 6-11 and 6-12, we integrated an edge server closer to the manufacturing network to exchange data and process required operations. The edge server has another communication channel at a higher level for interaction with cloud devices scheduled at hours with less communication traffic, with fewer production activities (saving the bandwidth), or for actions of non-time-critical responses. In a communication network, not all frames have the same priority. Our communication prototype uses TSN capable synchronized switches to guarantee delay-free communication for time-critical data.

6.4 Results summary and discussion

We proposed two industrial communication prototypes to improve the network infrastructure of time-critical IIoT applications. We design our communication prototypes to offer more determinism and low communication latency in industrial networks by implementing software and hardware communication concepts to create reliable communication systems. We applied concepts like TSN and edge computing to produce effective data transmission between network devices on the software side. TSN is responsible for accomplishing a deterministic communication where time-critical frames are given priority and delivered to their destination without unnecessary delays. The edge computing technology lessens communication latency with relation to information exchange by providing advanced computing services closer to the manufacturing factory, unlike cloud services located far away. On the hardware side, we integrated zero-loss redundancy protocols to ensure communication backup channels in case of failures at the main communication routes with the most negligible recovery and reconfiguration time. The zero-loss redundancy protocols we incorporate are PRP and HSR. These two redundancy protocols stand out from standard redundancy protocols by their ability to transmit duplicated frames in their communication networks to increase the chances of delivering the information at the destination if one side of the network encountered an unforeseen outage. The receiving device considers the first arriving frame and rejects the second duplicated one.

6.5 Chapter Summary

In this chapter, we presented the design of two industrial communication prototypes to achieve a robust network communication system using advanced communication techniques and redundancy protocols. We discussed the theory behind some network communication theories, such as TSN, edge computing, and network communication theologies utilized to develop our improved communication network prototypes. We outlined different popular industrial network redundancy protocols implemented in communication networks. We stressed two zero-loss redundancy protocols: PRP and HSRP, incorporated in the design of our proposed improved network communication prototypes. Our proposed industrial network communication prototypes improve the network infrastructure of a manufacturing factory to guarantee a reliable, deterministic, and low-latency communication network for time-critical IIoT applications. We achieved this goal by combining various state-of-the-art communication technologies such as TSN, edge computing, and zero-loss redundancy protocols in the same network infrastructure.

Chapter 7 : IMPLEMENTATION OF AUTOMATIC PARAMETER CONFIGURATION FOR A SCADA SYSTEM USING ML TECHNIQUES.

ML customized solutions and applications promise to transform traditional equipment into intelligent devices, increase flexibility, and boost production systems. Our innovative technique converts a traditional SCADA system into an intelligent and self-configurable device by merging ML algorithms (MLR and DT) and concepts. The SCADA platform that was configured manually for each product parameter should, after successful implementation of our adaptive technique, predict the best corresponding parameters for each product by scanning some of them. This process saves configuration time and contributes to starting production on schedule without depending on several approvals.

7.1 Decision Tree (DT) and Multiple Linear Regression (MLR) ML theoretical overview

7.1.1 Decision Tree (DT) machine learning algorithm for regression models and its limitations

We generate a DT regression model to predict numerical values that are product parameters entered in the SCADA system. We build the DT model by learning from existing product parameters saved from previous products configurations. The DT algorithm operates based on a splitting principle. The splitting principle computes the best possible splits for the learning process and generates several data areas from the best splits. The prediction results of the DT model corresponds to the average data value in each area (Breiman *et al.* 1984). We present a mathematical expression (7.1) of the residual sum of square (RSS) that correlates to the splitting principle (Romeo *et al.* 2018):

$$RSS = \sum_{i=1}^{k}(Y_i - f(X_i))^2 \tag{7.1}$$

where Yi is the input variable from which the prediction is made at a value of i and f(Xi) is the outcome, the prediction value of Yi.

The DT regression algorithm relies on few other important factors such as:

- **The entropy**: The entropy refers to the degree of randomness in a specific dataset. We present in (7.2) its mathematical expression.

$$Entropy = \sum_{i=1}^{k} X_i . \log_2 X_i \quad (7.2)$$

where k represents the data count and X(i) is a specific data type percentage over the whole dataset.

- **The information gain**: The information gain is the difference between two entropy values (a higher and lower degree) after splitting the data. (7.3) is an equation of the information gain.

$$\text{Gain} = \text{E(x)} - \text{E(x} - 1) \quad (7.3)$$

where E(x) and E(x-1) are two entropy levels at consecutive splits.

Some limitations of the DT algorithm: DT models structures can be very unreliable when dealing with unstable datasets. A slight variation of the dataset produces high instability for the model. The processing time of its modelling training phase takes longer to complete. It needs more memory in the processing hardware.

We display in Figure 7-1 a diagram of the DT algorithm operation.

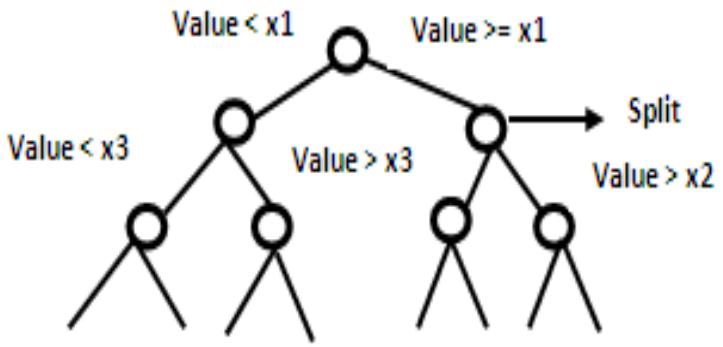

Figure 7-1: DT Algorithm operation diagram (Kiangala & Wang 2020 b)

7.1.2 Multiple Linear Regression (MLR) algorithm

MLR is a reasonably simple ML algorithm invented to predict numerical parameters whose values depend on more than one independent variable. We can mathematically compare a simple linear regression equation to an MLR one in (7.4) and (7.5):

$$y = a_0 + a_1 x_1 \tag{7.4}$$

$$y = a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_n x_n \tag{7.5}$$

where y is the dependent variable value that we predict from the regression models, x is the independent variable to which y depends or that has effects on y, and a_0 is a constant that represents the coefficient of proportion change in the regression models.

Figure 7-2 is a graph representing the linear regression algorithms.

Some limitations of the MLR algorithm: MLR algorithms assume that independent and dependent variables of the data utilized to train the ML model have a linear correlation. Non-linear data produces an inaccurate model. The algorithm is sensitive to data exceptions. For example, an observation that seems out of range compared to the remaining dataset can cause a volatile model.

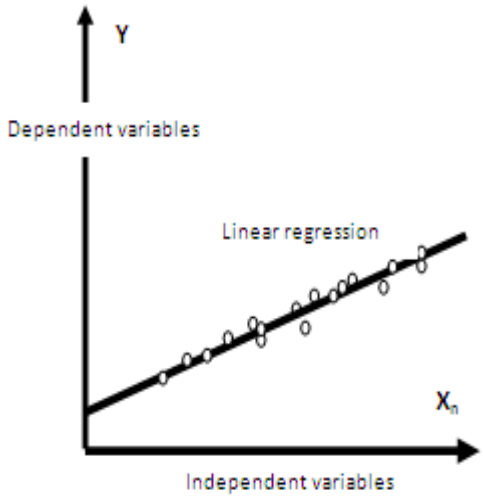

Figure 7-2: Linear regression algorithm graph (Kiangala & Wang 2020 b)

7.2 Modelling of an automatic parameter configuration method for a SCADA system using AI methods and innovative techniques

The heart of our automatic parameter configuration method is the execution and the combination of ML techniques: MLR and DT. We implement these two ML techniques for regression tasks. MLR and DT are data-driven approaches implemented to predict numerical data (the SCADA system parameters). Figure 7-3 is a summary of the automatic parameter configuration method combining two ML regression techniques.

Figure 7-3 contains various information on DT and MLR from our theoretical modelling section on the automatic parameter configuration framework.

We design our automatic parameter configuration framework using the following entities and variables (Kiangala & Wang 2020 b):

B : Dataset;

ρ : Dependent Variables;

ε : Independent Variables;

Φ : Linear Variables;

Ψ : Non-Linear Variables;

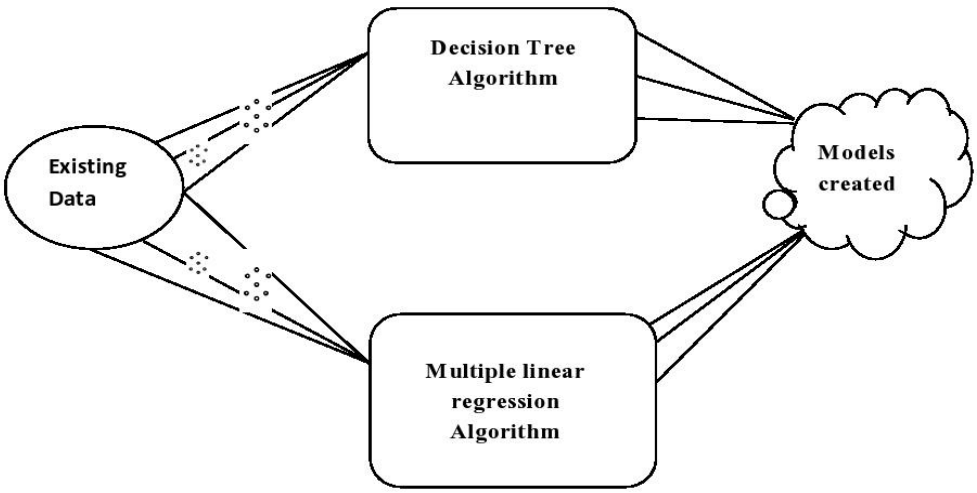

Figure 7-3: Automatic parameter configuration framework operation summary (Kiangala & Wang 2020 b)

Ω : Machine learning Algorithms (used in this framework);

δ : Decision Tree;

λ : Multiple Linear Regression;

E : Equivalence;

η : Predicted Variables

The following mathematical expressions (7.6) and (7.7) describe the methods executed to build our final automatic parameter configuration framework. They utilize the above variables and entities.

$$\mathrm{B} = \{\rho, \varepsilon\} \tag{7.6}$$

$$\rightarrow \{(\rho, \varepsilon) \in \Phi\} \vee \{(\rho, \varepsilon) \in \Psi\}$$

$$\Omega = \{\delta, \lambda\} \tag{7.7}$$

$$If\,(\rho, \varepsilon) \in \Phi \Rightarrow \lambda$$

$$If\,(\rho, \varepsilon) \in \Psi \Rightarrow \delta$$

The ultimate goal of our design is to predict new variables η by reading information from variables input ε. We utilize the composition of function principle to model our framework final prediction algorithm (7.8). Figure 7-4 presents the modeling diagram of this framework using the composition of two functions. From Figure 7-4, we notice that the ρ dataset is transparent during the design process.

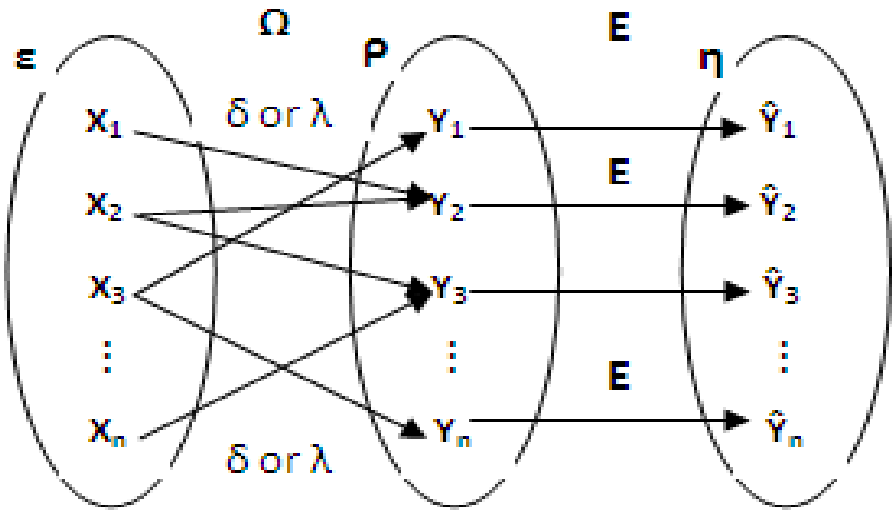

Figure 7-4: Automatic parameter configuration model diagram (Kiangala & Wang 2020 b)

$$\Omega : \varepsilon \mapsto \rho$$

$$\mathrm{E} : \rho \mapsto \eta$$

$$\mathrm{E} \circ \Omega : \varepsilon \mapsto \eta$$

$$(\mathrm{E} \circ \Omega)_{(x)} = \mathrm{E}(\Omega_{(x)}) = \mathrm{E}(Y) = \hat{Y} \tag{7.8}$$

7.2.1 Assumptions

We assume that the SCADA system can automatically read predicted parameters from the ML model using an interconnection via an application programming interface (API) in the automatic parameter configuration method.

7.2.2 Automatic parameter configuration architecture and summary

Implementing our automatic parameter configuration framework aims to improve the manufacturing system production throughput and lessen system configuration inaccuracy. Our parameter prediction model framework is essential for the successful manufacturing of quality products (using the correct configuration parameters) and for the efficient personalization of past, present, and future goods.

By implementing our automatic parameter configuration framework, we are able to achieve the following:

- Eradicate factories operators' repetitive (robotic) functions that result in unnecessary operational delays, such as designated users or supervisors dependence for parameters configuration during the production processes;

- Redirect factories operators' tasks in an innovative manufacturing environment, moving from automatic or repetitive functions to more strategic, operational roles.

- Ameliorate the overall production process accuracy by reducing the risks of human factor error when configuring parameters.

We illustrate in Figures 7-5 and 7-6 a manufacturing system operation before and after applying our automatic parameter configuration system. We notice from these two graphical representations how the role of operators and supervisors has shifted.

7.2.1 Evaluation of ML regression models

We evaluate the reliability of our regression models by computing a metric called the mean square error (MSE). (7.9) is the mathematical expression of the MSE (Wang, Xia *et al*. 2018). A good assessment indication of a regression model using the MSE is that lower values of the MSE mean a more accurate model than higher values.

$$MSE = \frac{1}{p}\sum_{n=0}^{p}(B_n - \widetilde{B_n})^2 \tag{7.9}$$

where p represents the total count of predictions, B_n is an instance of recorded observations values, and $\widetilde{B_n}$ represents an instance of predicted values.

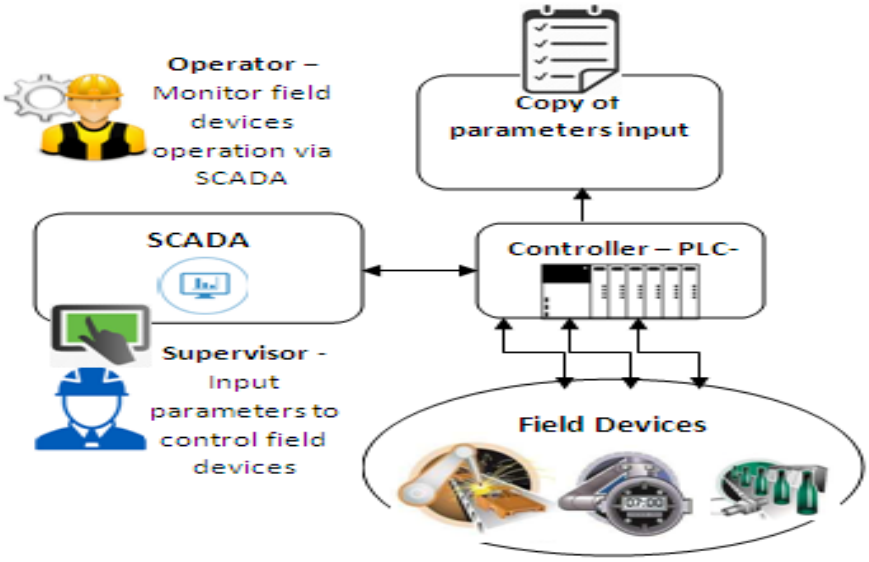

Figure 7-5: Manufacturing system with SCADA platform before automatic parameter configuration method (Kiangala & Wang 2020 b)

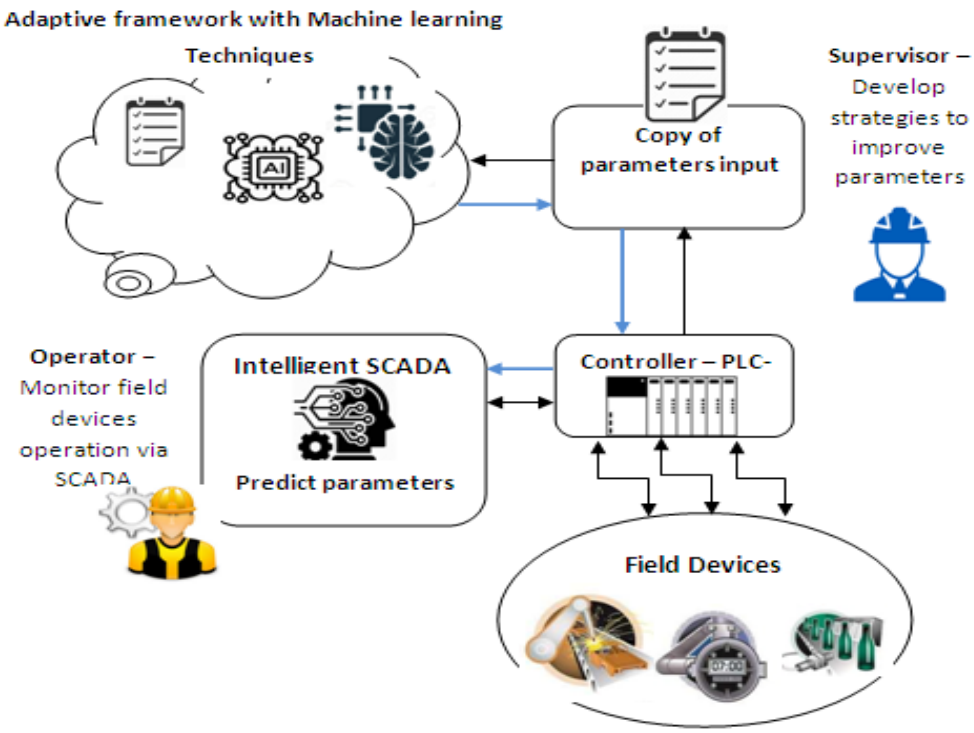

Figure 7-6: Manufacturing system with SCADA platform after implementing automatic parameter configuration method (Kiangala & Wang 2020 b)

7.3 Implementation of a SCADA system automatic parameters configuration

In order to free manufacturing plant supervisors from the repetitive action of inputting several product parameters in a SCADA system to start operations, we merge two ML algorithms: MLR and DT, to create a model that automatically predicts the corresponding parameters by scanning product data such as the width, height, diameter, and weight. We classify knowledge-based data for product parameters configuration in two groups: linear data and non-linear data. We predict linear parameters using the MLR algorithm and non-linear ones using the DT algorithm.

7.3.1 Linear data prediction using MLR algorithm

We combine the linear data into two sets of predictable data, each having independent and dependent variables. The independent variables come from the raw rubber material: width, diameter, and height. Our MLR model should predict the dependent variables (heating pressure and curing time) by scanning the independent variables. Tables 7-1 and 7-2 display the two sets of linear data combination variables.

Table 7-1: Set 1 - linear combination variables

Independent Variable	Dependent Variable
Width	Heating Pressure
Height	
Diameter	

Table 7-2: Set 2 - linear combination variables

Independent Variable	Dependent Variable
Width	Curing Time
Diameter	
Height	

We apply the MLR algorithm in the R IDE to generate our predictive model. We implement the same process for both data sets one at a time. In this section, we illustrate results with data set

1. The essential steps and parameters utilized to create the ML models in R for both data sets are as follows:

- Loading the whole dataset in the R software: The dataset is saved in a csv format. We display a portion of the loaded dataset in Table 7-3.

- Splitting the loaded dataset into a training set and validation set: We give a higher proportion to the training set than the validation set to feed as much information as possible to our ML model during the learning or training phase (The proportion are of 80% of the dataset for training set and 20% of the dataset for the validation set).

- Generation the ML model by fitting the MLR algorithm (from the R software libraries) to the training set.

- Testing the reliability of the created model by loading the validation data (20%) into the prediction model and comparing the outcome to the initial loaded data.

Table 7-4 shows the results of the Heating pressure variables values predicted by our model. We can visually compare some of its values (rows 11 and 18) to the original data loaded in Table 7-3.

A quick comparison of the predicted values versus the original ones demonstrates that the predicted data is very similar to the original one and based on error tolerance. This model can be safely used for future prediction of product parameters. We generate a statistical report on our MLR predictive model in Table 7-5 to support the viability of our model. These statistics also give us more insight into the significance of our independent variables. Only the width and the height of the raw products bear more meaning for the design of the predictive model. The diameter values are negligible, and ignoring them would not affect the performance of the ML model.

Table 7-3: A portion of the initial set 1 non-linear data loaded in R

Item	**Width**	**Height**	**Diameter**	**Heating Pressure**
10	60	40	40	55

11	100	50	50	95
12	100	80	40	110
13	80	60	45	90
14	40	30	40	45
15	80	40	45	72
16	60	80	45	87
17	60	80	45	90
18	30	40	30	45

Table 7-4: Predicted Heating pressure values by the MLR model

y_pred								
4	**6**	**11**	**18**	**19**	**21**	**42**	**67**	**77**
45.7	83.8	91.4	45.1	62.7	51	107.3	83.9	62.7

Table 7-5: Statistical results summary of set 1 MLR model

Coefficients:					
	Estimate	**Std.Error**	**t Value**	**Pr>|t|**	
Intercept	6.232	2.099	2.969	0.004	**
Width	0.584	0.027	21.33	$< 2.e^{-16}$	***
Height	0.529	0.034	15.70	$< 2.e^{-16}$	***
Diameter	0.006	0.041	0.136	0.892	
Sign. code	0 '*** ' , 0.001 '**' , 0.01 '*' , 0.05 '.', 0.1 ' '				
Multiple R^2: 0.9509, Adjusted R^2: 0.949					
p-value: $< 2.2e^{-16}$					

7.3.2 Non-Linear data prediction using DT algorithm

We perform the prediction of non-linear data using the DT ML algorithm. In order to train the ML model, we divide the non-linear variables into two sets of independent and dependent variables, as illustrated in Tables 7-6 and 7-7**.** Each set is loaded individually in the DT ML model and divided into a training set and a validation set.

Table 7-6: Set 1 – Non-linear data combination for DT ML

Independent Variable	**Dependent Variable**
Product Weight	Number of Bumps

Table 7-7: Set 2 – Non-linear data combination for DT ML

Independent Variable	**Dependent Variable**

Level of Material_A	Bump delays

We create the DT ML model using variables in Table 7-6, but the same procedure is applicable for Table 7-7.

Below are the important steps and parameters to create the DT ML model in R:

- Loading the complete csv format dataset (a portion displayed in Table 7-8) into the R platform.
- Dividing the loaded dataset into a training set (of 80% of the overall data) and a validation set (of 20% of the overall data).
- Generating the DT prediction model by laying the training set into the decision tree algorithm (the DT algorithm exist in R libraries).
- Experimenting the viability of the DT model created by loading the validation set into the model and comparing its results to the original data.

The DT algorithm results depend on a splitting criterion that determines the outcome of each decision branch. The choice of the correct number of splits is crucial to generate an accurate prediction. For the non-linear data set1 variables in Table 7-6, we input a minsplit of value '10'. We present our predictions for the Number of Bumps using the DT model in Table 7-9**.**

Table 7-8: Portion of non-linear dataset loaded in R

Item	**Product Weight**	**Number of Bumps**
11	950	23
12	950	23
13	950	24
14	950	22
15	950	22
16	950	22

17	600	10
18	400	5
19	400	4
20	750	15
21	400	4

Table 7-9: Predicted number of bumps with test data set

y_pred								
14	**17**	**18**	**21**	**22**	**25**	**30**	**33**	**34**
23.22	9.56	3.64	3.64	23.22	23.22	23.22	23.22	16.22

We observe that the values are accurate by comparing the predicted values of the number of bumps in line items 14, 17, 18, and 21. Based on the error tolerance level, the created DT model can successfully be implemented to predict the desired number of bumps for any product weight. We display a plot of our DT model for the predicted values with the average of each split interval in Figure 7-7, and in Figure 7-8, we represent the graphical tree of the DT model with different splits.

Figure 7-7: DT model plot for data set 1 – Number of bumps versus Product weight (Kiangala & Wang 2020 b)

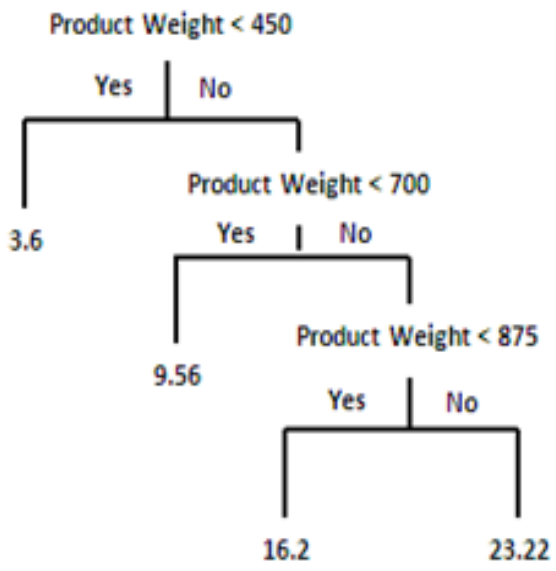

Figure 7-8: Graphical Tree of DT model with different splits values (Kiangala & Wang 2020 b)

7.4 Results summary and discussion

In this innovation area, we change a traditional Human Machine Interface (HMI) containing a SCADA system to an I40 intelligent, self-configurable device through which manufactured products can be configured faster and efficiently using parameters predictions. In order to achieve this task, we combined two ML algorithms MLR and DT, and created a predictive system using the knowledge-based data from previous products configured in the old SCADA system. By scanning some of the product's available features, such as the height, diameter, and weight, the SCADA system generates all required parameters instantaneously to initiate the manufacturing. The new self-configurable SCADA shifts the role of the supervisor or the operator initially in charge of manually configuring the product parameters away from the repetitive action of product configuration to more strategic duties about planning and improvement of parameters quality. Figures 7-5 and 7-6 summarize the impact of the new SCADA system with operators' and supervisors' roles. Thanks to the new SCADA system, the factory operation regarding product configuration does not longer relies on a supervisor's competence to set up critical parameters that control production. Therefore, our innovation technique contributes to time-saving and improves the overall production process' efficiency.

7.5 Chapter Summary

In this chapter, we developed an automatic parameter configuration scheme for a SCADA system using two ML regressions models: DT and MLR. We started the design by exploring applicable theories behind DT and MLR ML algorithms to create regression models implemented in the design of the automatic parameter configuration method. We highlighted the design architecture for combining the two ML algorithms used in our final system structure: MLR and DT for regression models. We presented the final system algorithm generated by merging the two ML algorithms. We also displayed the impact of our automatic parameter configuration method in a small manufacturing environment graphically before and after implementing the automatic parameter configuration system. Our innovative system transformed a traditional SCADA system into a self-configurable and intelligent I40 device. The traditional SCADA that depended on manual product parameters inputs from a supervisor is equipped with a prediction structure to automatically generate several parameters after scanning some of the known product's features, such as its dimension and weight.

Chapter 8 : AN ADAPTIVE PRODUCT CUSTOMIZATION PLATFORM FOR CUSTOMERS INTERACTIONS WITH PRODUCTION SYSTEM

We design an adaptive product customization platform, suitable for a SME, which permits customers to amend a product composition, within guided limits, as per their preferences by changing a single product parameter. We intend to alleviate customers' tasks to input several complex parameters to match their desired product structure by predicting the corresponding parameters for each entry and by computing for them instantaneously the final product result. In order to reach this goal, we exploit the advantages of some powerful ML algorithms like XGBoost and RF ensemble learning that we should apply in the customization platform's backend.

8.1 XGBoost and RF ML algorithms theoretical overview

8.1.1 Extreme Gradient Boosting (XGBoost) algorithm for regression tasks and its limitations

XGBoost is an efficient ML algorithm initially designed by Chen & Guestrin (2016) to optimize the performance of existing DT algorithms. The XGBoost algorithm operates in a serial data training procedure where the system combines results of each previous weak to create a more robust predictor. The result of an XGBoost model is a forest with multiple DT. Some studies conducted by Yang et al. (2017), Zhao et al. (2018), and Wang et al. 2019 prove that XGBoost algorithms produce more accurate outcomes than single DT algorithms, SVM, and Gradient Boosting Decision Tree (GBDT). The XGBoost algorithm has several other merits, such as:

- Reducing the risks of overfitting in the modelling process by making use of a regularization factor.

- Using effective memory processing resources when computing the ML models that produces faster processing results (Mo *et al.* 2019).

- Substituting missing data during the training process. The algorithm takes care of empty values in the training dataset. This feature is called "Spare Aware" capability (Reinstein 2017).

- Possessing highly flexible, efficient, and portable programming libraries (Yue *et al.* 2021). The algorithm belongs to the Distributed Machine Learning Community (DMLC).

The XGBoost algorithm can perform three boosting methods: Gradient boosting, Stochastic boosting, and Regularized boosting. The XGBoost algorithm is suitable for regression and classification problems.

We present in (8.1) a dataset, B, containing a set of independent variables, (g_i), and dependent variables (h_i).

$$B = \{g_i, h_i\} \tag{8.1}$$

where *i* represents the *i-th* sample of the dataset, $i \in \{0 \ldots . k\}$ with *k* is the number of all samples in the entire dataset B.

Considering that the count of DT in the generated model is equal to *G*, we can compute an expression of predicted values $\widetilde{h_i}$ at a sample *i* in (8.2) as:

$$\widetilde{h_i} = \sum_p^G f_p(g_i) \tag{8.2}$$

where $f_p(g_i)$ is the count of the predicted values of the instance *i* for the *p-th* tree.

The XGBoost algorithm optimizes its models' results by integrating an objective function that lessens the loss function of previous predictors' results. As mentioned previously, the XGBoost algorithm also incorporates a regularization to reduce the risks of overfitting. The regularization function also expresses the degree of complexity of the model. We present in (8.3) the objective function, in (8.4) the loss function $\mathrm{l}(\mathrm{h_i}, \tilde{\mathrm{h}}_\mathrm{i})$ and, in (8.5) the regularization function $\Omega(\mathrm{f})$.

$$K = \sum_{i=1}^{n} l(h_i, \tilde{h}_i) + \sum_{p=1}^{G} \Omega(f_p) \tag{8.3}$$

where $l(h_i, \tilde{h}_i)$ represents the loss function that is the variation between the dependent variable (h_i) and its predicted value $\widetilde{h_i}$, n is the data count processed at the *p-th* tree, G is the model tree count, and i represents the *i-th* instance of the dataset.

$$l(h_i, \tilde{h}_i) = \sum_{i=1}^{n} (h_i - \tilde{h}_i)^2 \tag{8.4}$$

$$\Omega(f) = \theta T + \frac{1}{2}\eta \sum_{b=1}^{T} (X_b{}^2) \tag{8.5}$$

Where θ and η represent the weight parameters for overfitting risks in the XGBoost model (Jiang *et al.* 2020 and Suo *et al.* 2019) demonstrates that higher values of the two weight parameters suggests a simple DT structure with less chances of overfitting. *T* is the count of lead nodes in the model, X is the weight of a node in the entire leaf, and *b* is an index that identifies every leaf in a node.

We can compute the objective function (8.3) at an iteration *s* as (8.6). (8.6) can then be optimized by applying a second order Taylor approximation method and defined as (8.7).

$$K^{(s)} = \sum_{i=1}^{n} l(h_i, \tilde{h}_i{}^{s-1} + f_s(g_i)) + \Omega(f_s) \tag{8.6}$$

$$K^{(s)} = \sum_{i=1}^{n} [l\left(h_i, \tilde{h}_i{}^{s-1}\right) + m_i f_s(g_i) + \frac{1}{2} n_i\, f^2(g_i)] + \Omega(f_s) \tag{8.7}$$

where m_i and n_i gradient statistics parameters of the loss function $l(h_i, \tilde{h}_i)$. (8.8) and (8.9) are mathematical expressions of the two gradient statistics parameters.

$$m_i = \partial_{\widetilde{h_i}^{(s-1)}} l(h_i, \tilde{h}_i{}^{(s-1)}) \tag{8.8}$$

$$n_i = \partial^2 \quad {}_{\widetilde{h_i}^{(s-1)}} l(h_i, \tilde{h}_i{}^{(s-1)}) \tag{8.9}$$

We can compute an optimal expression of the loss function at a leaf *t* as presented in (8.10) by replacing the expressions of the regularization function (8.5), the two gradient statistics parameters (8.8) and (8.9) in the optimized objective function (8.7) and pulling out its derivative. The smaller the value of the optimal loss function (8.10), the better the tree

composition. The weight of a leaf t is expressed in (8.11) (Chen & Guestrin (2016) and Jiang et al. 2020).

$$K^* = -\frac{1}{2}\sum_{t=0}^{T}\frac{(\sum m_i)^2}{\sum n_i+\eta} + \theta T \tag{8.10}$$

$$X_t{}^* = -\frac{\sum m_i}{\sum n_i+\eta} \tag{8.11}$$

Some limitations of the XGBoost algorithm: Although the XGBoost algorithm presents several advantages, it produces less accurate results when training imbalanced label data in classification models. The data imbalance occurs when one or more categories in the training dataset have lower proportions than the others (Wang *et al.* 2019).

In our adaptive customization platform, we implement the XGBoost algorithm to build a regression model that should predict the corresponding product parameter to each customers' entry. We illustrate in Figure 8-1 the operation workflow of the XGBoost algorithm.

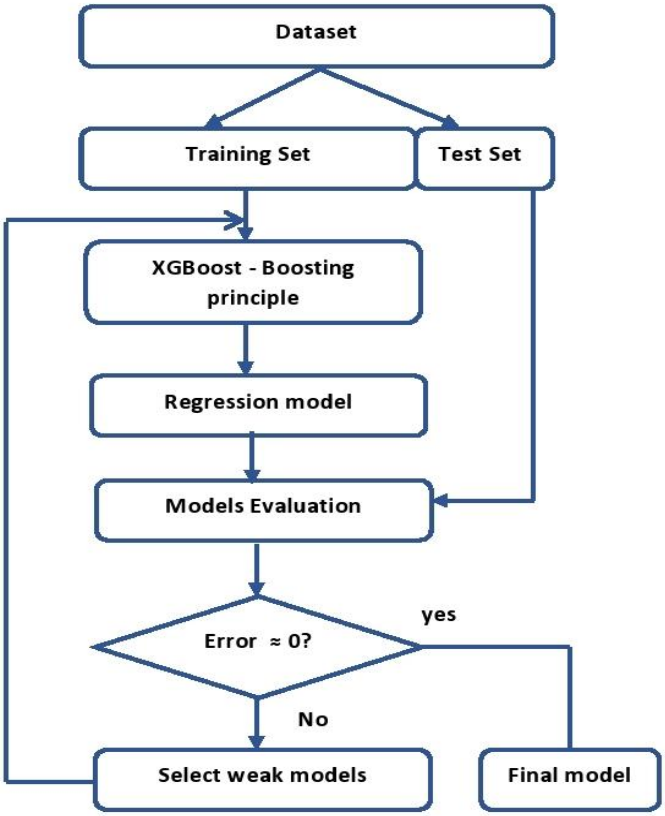

Figure 8-1: XGBoost algorithm operation work flow (Kiangala & Wang 2021 a)

8.1.2 Random Forest (RF) algorithm for classification tasks and its limitations

Breiman (2001) first introduced the RF algorithm known as an ensemble learning algorithm because of its composition of different groups of DTs. The RF algorithm principle came from the originated bagging algorithm (Breiman 1996). It consists of several parallels and independent DT models (Liaw & Wiener 2002). The RF algorithm generates several random mini-training datasets from the initial training set using bootstrap sampling. The DT algorithm operations are then applied to each sub-training set (Hefner *et al.*, 2014). The result of the RF model is the average of each DT model's results. Therefore, the RF algorithm has higher chances to produce better results than single DT algorithms. We can summarize the RF model operation in two essential stages: the forest generation and decision-making stages. Various applications utilize the RF algorithm (Tyralis *et al.* 2019; Diaz *et al.* 2021 and Xu & Luo 2021). Benali et al. (2019) conducted research that demonstrated that the RF algorithms produce better outcomes and accuracy in high-dimensional nonlinear problems. The RF algorithm works for regression and classification cases.

Let us consider (8.12) the expression of a training dataset *G* with its elements *g*:

$$G = \{g_1, g_2, \dots g_p\}\,; \qquad (8.12)$$
$$G \in R^{m \times r}$$

where g_i is the index location of a data sample *i in G*, $R^{m \times r}$ is the dataset space containing all elements of G, *m* is the samples count in the dataset G, *r* represents the corresponding number of features of each sample.

We present in (8.13), the ensemble RF algorithm expression for a dataset element *g* in a tree *s*.

$$\mathrm{f_s(g) = f(g, \theta_s)} \qquad (8.13)$$

where θ_s represents a random vector that splits the training dataset of each corresponding element vector *g*.

(8.14) is a mathematical expression of the prediction probability of a given class *v* in relation with a data sample *g* for the RF algorithm. The margin function is an essential metric that determines the average level of votes of correct predicted classes versus other classes' votes in the RF classification process. (8.15) is an expression of the margin function for RF classification. (8.16) represents the actual decision function of the RF algorithm (Dong *et al.* 2015).

$$\mathrm{P(v|g)} = \frac{1}{\mathrm{S}}\sum_{\mathrm{s=1}}^{\mathrm{S}} \mathrm{P_s(v|g)} \tag{8.14}$$

where $\mathrm{P(v|g)}$ is the estimated density of class *v* labels for a specific sample data *g* in a tree *s*. S represents the total count of participating trees in the forest.

$$\mathrm{mg(g,v)} = \mathrm{P(v|g)} - \mathrm{maxP(j|g)} \tag{8.15}$$

The margin function equation results in (8.15) convey some insights on the classifier's results reliability. The higher the value of the margin function, the more reliable the classifier's results.

$$D(g) = argmaxP(j|g) \tag{8.16}$$

The RF algorithm has the merit of reducing the risks of generalization errors in classifiers models. It lessens this error by its ability to train several classifiers using a random sub-training dataset. (8.17) expresses the generalization error (Wang, Xia, *et al.* 2018).

$$\mathrm{GE^*} = \mathrm{P_{g,v}(mg(g,v))} < 0) \tag{8.17}$$

Some limitations of the RF algorithm: The outstanding performance of the RF algorithm depends highly on the amount of labeled data in the training set (Gislason *et al*. 2006)**.** The higher the amount of labeled data, the more accurate the results. Results produced by an RF model trained with less labeled data are not reliable.

In our research, we utilize the RF algorithm to design a classifier model that categorizes customized products in the customization platform backend and the XGBoost regression model. We position the RF algorithm to receive the XGBoost regression output as an input of the RF algorithm. Figure 8-2 is a graphical representation of the RF operational principle, and Figure 8-3 summarizes its workflow.

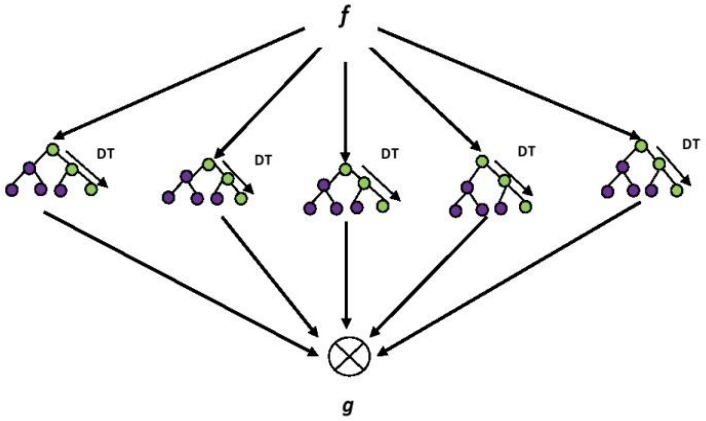

Figure 8-2: RF algorithm graphical operation principle (Kiangala & Wang 2021 a)

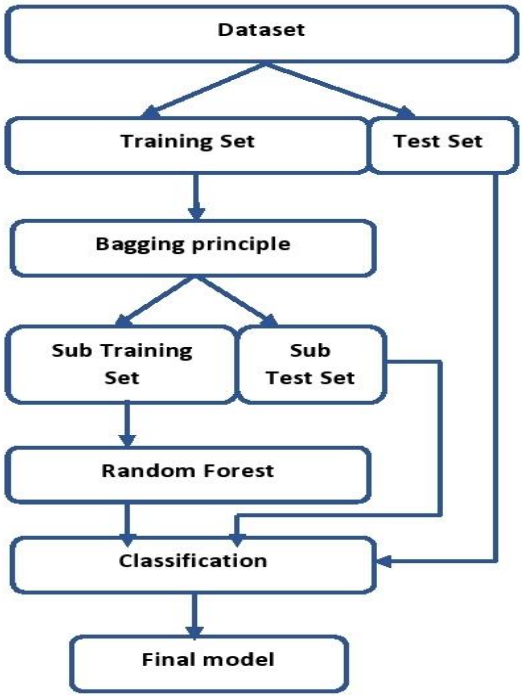

Figure 8-3: RF algorithm operation work flow (Kiangala & Wang 2021 a)

In summary, our adaptive product customization platform combines a regression model built with the XGBoost algorithm and a classification model trained on an RF algorithm. These two ML models run in the customization platform backend. We present in (8.18) a simple mathematical expression of our customization platform with the following assumptions:

- We represent the final customization algorithm of our platform by $\boldsymbol{\rho_{custom}}$
- The symbol of the XGBoost regression model function is $\boldsymbol{f}$
- The symbol of the RF classification mode is $\boldsymbol{g}$
- We represent the customization parameters that customers input in the system by $\mathbf{R_1, R_2,...,R_n}$; with $\boldsymbol{n}$ the total count of customization parameters customers are allowed to enter for the personalization of their products.(For testing of our experimental platform we assume n=1, the input parameter is R_1)
- We represent the parameters predicted by the XGBoost model *f* after the input by $\mathbf{B_1, B_2,...,B_n}$; with $\boldsymbol{n}$ the total count of predicted parameters corresponding to the input parameters $R_1, R_2,...,R_n$ (For testing of our experimental platform we assume n=1, the predicted parameter is B_1)
- We symbolize the product categories predicted by the RF classification model $\boldsymbol{g}$ as $\mathbf{cl_1, cl_2,...,cl_p}$**;** where p means the total count of predicted product categories.

$$f_n: \ R_1, R_2, \dots, R_n \ \mapsto \ B_1, B_2, \dots, B_{n\,;} \quad B_i \ \in \mathbb{R}; \ n \in \mathbb{N}$$

$$g_n: x = \{ (R_1, R_2, \dots, R_n \ )U(B_1, B_2, \dots, B_{n\,;}) \} \mapsto \ y$$

$$y \in \{cl_1, cl_2, \dots, cl_p\}; \quad p \in \mathbb{N}/\{0,1\}$$

$$\rho_{custom} = f_n \ U \ g_n \tag{8.18}$$

8.2 Modelling of an adaptive customization framework using XGBoost and RF ML algorithms

Our client adaptive customization framework consists of three principal segments: the customization framework front-end presented in Figure 8-4**,** the customization framework

back-end, and the framework bridge segment. The customization framework front-end ensures the interaction platform between the clients and the factory production system. As displayed in Figure 8-4**,** it is a GUI with instructions for customers' navigations and system feedback on the customized product. The customization framework back-end carries the ML algorithms and models we developed for product customization. This segment is the brain of the overall system. The last customization framework bridge section is responsible for transferring the details of the personalized goods to the manufacturing plant once finalized by the back-end. Our study focuses mainly on the design of the second segment: the customization framework back-end. We display in Figure 8-5 the operational workflow of our adaptive customization framework. Figure 8-6 is a summary of the framework architecture presenting the three principal segments.

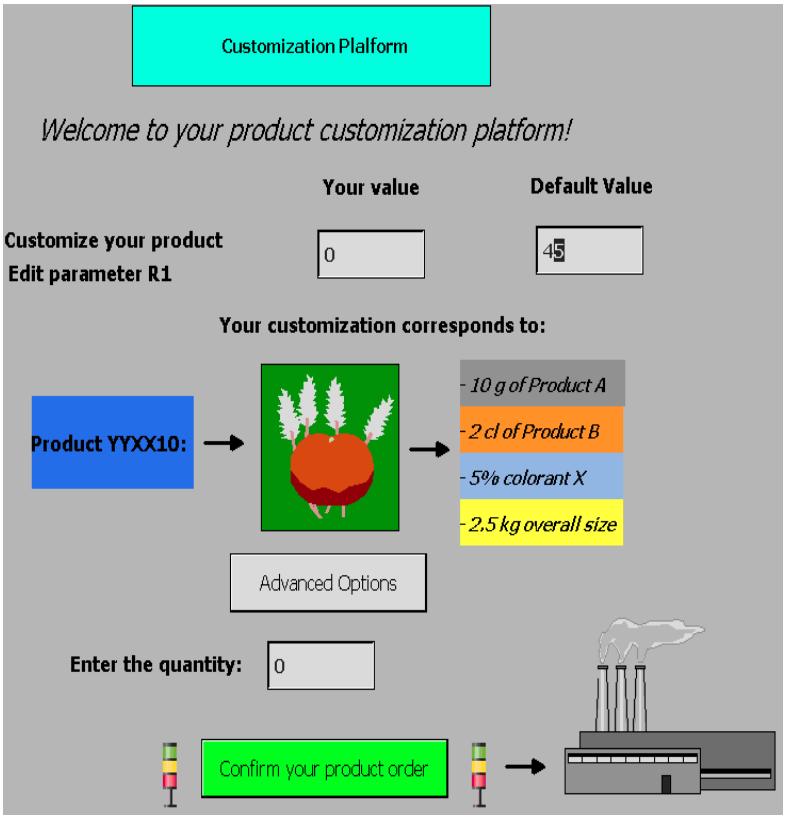

Figure 8-4: Customization framework front-end template (Kiangala & Wang 2021 a)

From Figure 8-5 displaying the product customization framework operational workflow, we notice that the customization process depends on the customer personalized input to start operating. The customer entry is transferred into the XGBoost regression model to predict the

internal parameter corresponding to the customer entry. The predicted internal parameter is later utilized as the second input of the RF classification model (with the original customer entry as the first input) to classify the personalized product. The product customization framework system displays the final product category on the system front-end GUI in Figure 8-4 and forwards instructions to the manufacturing factory to start producing goods based on the customized information.

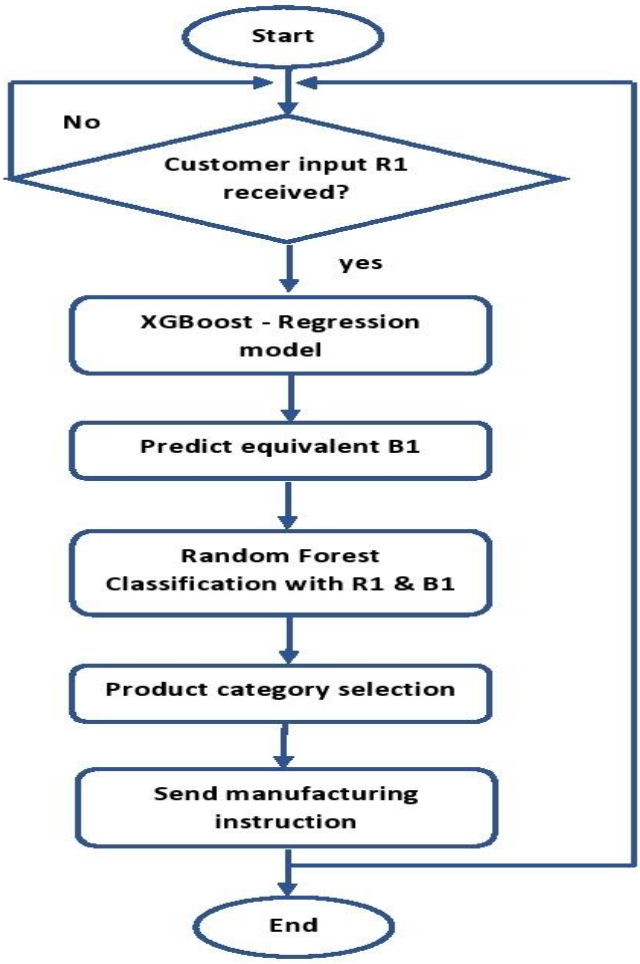

Figure 8-5: Customization framework operation workflow (Kiangala & Wang 2021 a)

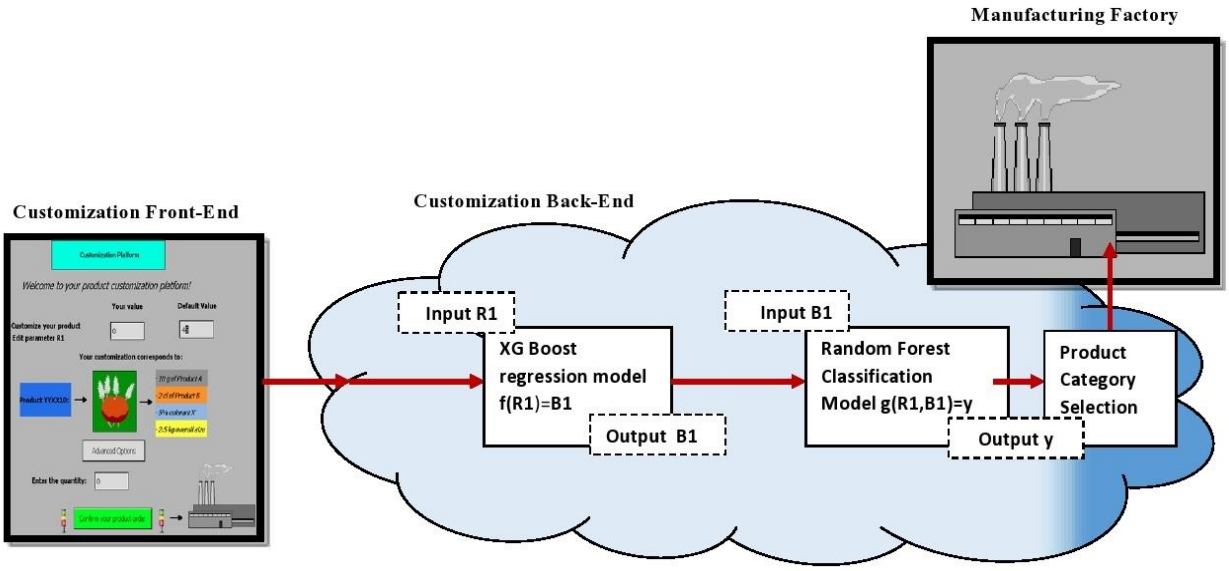

Figure 8-6: Customization framework overall architecture summary (Kiangala & Wang 2021 a)

8.2.1 Assumptions

For the design of our adaptive customization framework, we assume that the factory is a robust internet communication network linking the customers, the customization front and backend, the factory, and the suppliers. We also presume that the communication network is securely protected by devices such as firewalls and other robust security mechanisms.

8.3 Implementation of an adaptive customization platform using XGBoost and RF ensemble learning

8.3.1 Experiment background and challenges

A small manufacturing factory would like to implement product customization to improve their customers' experience and become more competitive. Being relatively new with the customization process, they chose the adaptive customization option to amend products within specific limits. During the customization operation, customers can personalize the content of the product ingredients by inputting three parameters: (i) product parameter 1, (ii) product parameter 2, and (iii) the product category as displayed in Table 8-1. The two product parameters are linked, and their combination produces the final product category. In order to

manufacture the final product, the plant requires a value for all three parameters. On their first trial, the small manufacturing plant made the following observations:

- It was difficult for some customers to match the correct product category with the corresponding product parameters. The instructions provided by the plant were not as easy to understand for everyone. The incorrect entries and selection were constantly rejected and resulted in many frustrations.

- Understanding how to use the system and inputting all three parameters without errors appeared to be time-consuming for some customers. The small manufacturing factory relies on a static database for its customization system in which the mapping between all parameters and product categories is done.

- The static customization database does not automatically create tables for new entries. The new product parameters update when the system is offline. Only the known parameters pre-existing in the data are authorized. The system declines the rest. This action is a safe setting for the adaptive customization guided limits.

- The plant requires specialized personnel to load and approve the new parameters entries in the customization database.

By developing our adaptive customization framework with ML algorithms, we offer the following solutions to the above challenges:

- We design a customization platform that receives input from customers and automatically matches them to the corresponding product categories. We reach this goal by creating an ML model that learns the ingredients parameters correlations with intelligent ML algorithms such as XGBoost and RF. We made use of the existing database information (knowledge-based data) to extract the correlation. This solution significantly reduces the rate of parameter rejection and incorrect parameter inputs.

- Our customization platform spares the factory the need for a workforce to update the database on new parameters since the ML models generated automatically determine the relationship between all three customization parameters and forecast corresponding observations for each input.

- Our customization model reduces the number of parameters required to personalize products from 3 inputs to a single input. The backend of the customization platform maps the single customer input to the related product parameter and the product category. This solution saves customers' time when personalizing their products.

The dataset we utilize to build our ML models in the customization platform backend comprises three variables: two product parameters describing a product's ingredient composition and a product category variable. In small manufacturing plants, the amount of processing data is relatively low. For this experiment, the index n in (8.18) is equal to 1, and p is equal to 3. The same procedure would apply to a larger dataset. We present a composition customization parameter in Table 8-1.

Table 8-1: Experimental variables structure

Product Parameter 1	**Product Parameter 2**	**Product Category**
R_1	B_1	Product A
		Product B
		Product C

We build our experimental customization platform based on an adaptive customization model where customers personalize the required product's ingredients within boundaries. The customer inputs a single variable R1 as per the variables in Table 8-1 and instantaneously obtains the equivalent product category. The input is done via a front-end, as displayed in Figure 8-4**.** We present in Table 8-2 four different ML models we build to test the reliability of our customization platform. These four models constitute the back end of two separate customization platforms, as illustrated in Table 8-2. In this experiment, we will compare both customization platforms' results and discuss the best one.

Table 8-2: Experimental ML models & Customization platforms

Model ID	Model name	Customization platform ID
1	DT regression	1 - (R_1, B_1)
2	XGBoost regression	2 - (R_1, B_1)
3	DT classification	1 - (R_1, B_1, Product A, Product B, Product C)
4	RF classification	2- (R_1, B_1, Product A, Product B, Product C)

8.3.2 ML regression models

The regression model is the first ML prediction model of our product customization back end. It predicts the corresponding product parameter related to the customer input. We build three regression models for evaluation purposes: one using the DT algorithm, a second using the XGBoost algorithm, and a third using the RF algorithm. Previously, we implemented classification ML models in our intelligent PM to predict fault categories. The regression ML models aim to predict numerical values (product parameters). Table 8-3 is a summary of basic settings to create a DT regression model in R.

Table 8-3: DT regression model settings for R

Settings	Value	Comment
Libraries	CaTools, rpart	rpart is the library for DT
Set.seed	123	
Split ratio	2/3	Training set and Test set
minsplit	100	DT model setting

The *"minsplit"* setting in Table 8-3 refers to the number of splits the algorithm applies when fitting to the training set. This DT regression model is the first ML model of the experimental customization platform ID-1 as per Table 8-2.

We create a second regression model using the XGBoost algorithm. This model is the first ML model of the back end of the proposed adaptive customization platform ID-2, as described in Table 8-2. Table 8-4 presents the settings for the XGBoost regression model computation. We design the ML model in R.

Table 8-4: XGBoost regression model settings for R

Settings	**Value**	**Comment**
Libraries	CaTools, xgboost	Xgboost is the library for XGBoost
Set.seed	123	
Split ratio	2/3	Training set and Test set
nrounds	100	XGBoost model setting

From Table 8-4, the “nrounds” setting refers to the number of times the XGBoost model is trained to produce the desired results.

The last regression model we generate for testing purposes is the RF regression model. We display its settings in Table 8-5. We build this model to get insights into the implementation of XGBoost over RF algorithms for non-linear regression data.

Table 8-5: XGBoost regression model summary settings

Settings	**Value**	**Comment**
Libraries	CaTools, randomForest	randomForest is the library for RF
Set.seed	123	
Split ratio	2/3	Training set and Test set
ntree	1	RF number of trees

8.3.2.1 Regression models parameters study

As mentioned previously, we used the platform 'R' to compute the DT, XGBoost and RF regression models. The parameters used in Tables 8-3, 8-4 and 8-5 are as follows:

- Libraries: CaTools, rpart (these are the libraries required for DT regression models in R), CaTools, xgboost (these are the libraries required for XGBoost regression models in R), CaTools, randomForest (these are the libraries required for randomForest regression models in R).

- SET.SEED (123): The set.seed parameter is a random number inserted to ease reproducibility. It's not a compulsory parameter to use.

- SPLIT RATIO (2/3): The split ratio is the proportion of data allocated to the training set compared to the test (validation) set. As a rule, giving a higher proportion to the training dataset is recommended than the validation set to reduce the risks of overfitting (Rácz *et al.* 2021). The Split ratio can be represented as 2/3 in 'R.,' which means that 2/3 of the loaded dataset goes to the training set or as a percentage (recommended between 0.7 – 0.8).

- minsplit (100): We tested our model with different minsplit values between 1 – 1000. The default value of the minsplit parameter is 20. Minsplit values of 1 and 1000 distorts our model prediction. Any minsplit value between 20 – 800 produces stable results. We selected 100. Since dealing with smaller datasets, the minsplit value does not impact the model processing time.

- nrounds (100): We tested our model with different nround values between 1 – 100. We achieved the model best accuracy from nrounds = 78. All values higher than 78 tested up to 100 give the same accuracy. We made the results discussion below based on the least nrounds required to achieve the best accuracy (78).

- ntree (1): Our RF regress model produced its best accuracy with a ntree equals to 1. A higher number of trees does not affect the accuracy but the processing time. We sticked to 1 tree.

8.3.3 ML regression models results and discussion

We represent the generated regression models graphically in Figure 8-7 (DT regression model), in Figure 8-8 (XGBoost regression model), and in Figure 8-9 (RF regression model). When observing the three graphs, we notice that the DT regression model blue line (Figure 8-7) does not go accurately through all observations represented as red dots. It means that the DT regression model is not very accurate in representing all observations. It has a higher error rate than the XGBoost and RF models, where the regression model blue line runs through almost all red dot observations. Therefore, the XGBoost and RF regression models are more accurate and have lower error rates than the DT regression model.

An essential metric to rate regression models' accuracies is the mean square error (MSE). It has been by Wang, Y., Xia, S. et al. (2018). Their findings reported that: "the lower the MSE, the more accurate the regression model." The lower value of MSE obtained in the DT regression model is equal to 5.82% at the 235th node. On the XGBoost regression model, we achieved a lower MSE of 0.031% minimum error. Testing this impact in two product parameters: R1=0.0927613 and B1= 1.413106509, we predict a value B1 =0.943 using the DT regression model. This result represents an error of about 33.27% for the model. On the XGBoost model, we predicted a value of B1=1.41303038597, which is equivalent to an error of 0.005%. It is the best accuracy of the model with the number of nrounds equal to 78. Using the RF model, the best result we obtained for the same R1 input was B1=1.41 regardless of the number of trees. In the testing, we utilized ntree =1, and we reached an error of 0.22%. The XGboost regression model offers the best accuracy than the DT or the RF regression models.

In terms of processing speed, the RF regression model is the fastest of all three models since it is trained with the least number of trees (ntree = 1 as per the settings in Table 8-5). In this experiment, since processing a relatively smaller data size, the XGBoost regression model processing time appeared to be only 2 seconds slower than the RF regression model processing time at 78 iterations. For significant data size, the processing time could be higher and create long delays. When training non-linear data, choosing the most appropriate regression model based on the application is crucial. When requiring a model that produces the best accuracy over the fastest processing speed, the XGBoost regression model is the most appropriate model to use.

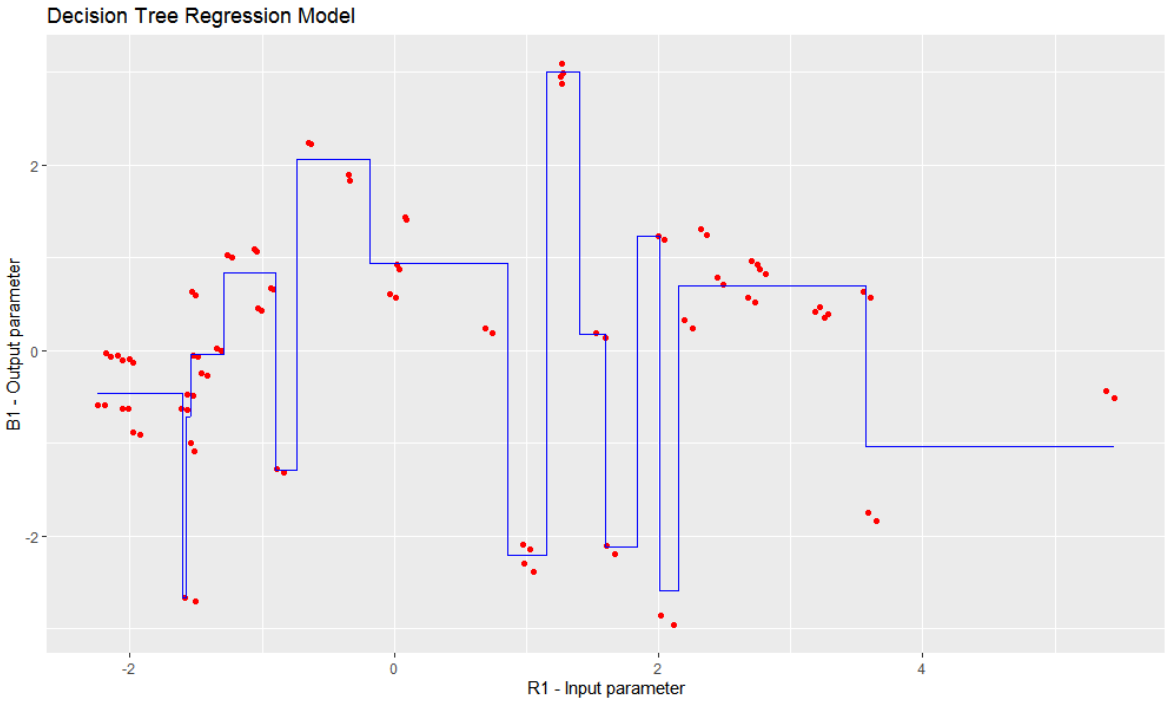

Figure 8-7: DT regression model graph (Kiangala & Wang 2021 a)

8.3.4 ML classification models

In order to determine the product category of the parameter entered by the customer, we developed a classification model coupled with the regression model in the production customization backend. For evaluation sake, we build two classification models loaded into two different experimental customization platforms back ends. We create the first ML classification model using the DT algorithm and combine this model with the previous ML DT regression model to constitute the first experimental product customization framework ID-1.

We build the second classification model utilizing the RF algorithm. This RF ML classification model is integrated into the previous XGBoost regression model to implement our proposed adaptive customization framework. Tables 8-6 and 8-7 summarize the basic settings for building the ML classification models using the DT and RF algorithms.

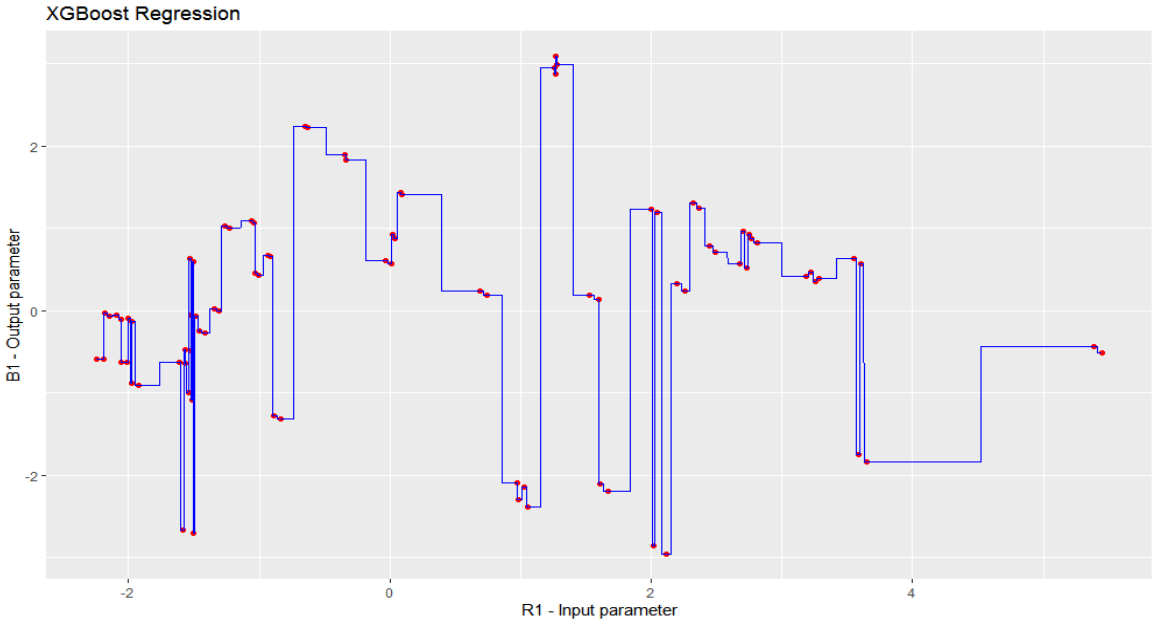

Figure 8-8: XGBoost regression model graph (Kiangala & Wang 2021 a)

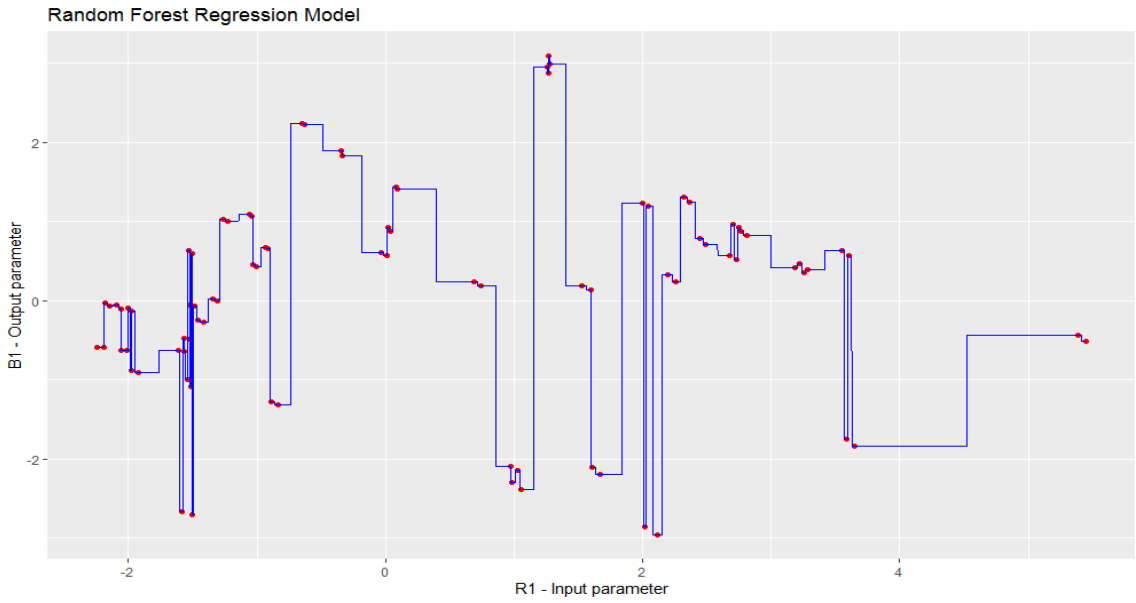

Figure 8-9: RF regression model graph (Kiangala & Wang 2021 a)

Table 8-6: DT ML classification model settings

Settings	Value	Comment
Libraries	CaTools, caret, performance analytics, rpart	rpart is the library for DT
Set.seed	123	
Split ratio	0.75	Training set and Test set
type	class	Test set prediction setting

Table 8-7: RF ML classification model settings

Settings	Value	Comment
Libraries	CaTools, caret, performance analytics, randomForest	randomForest is the library for RF
Set.seed	123	
Split ratio	0.75	Training set and Test set
ntree	500	Number of trees in the RF algorithm

8.3.4.1 Classification models parameters study

As mentioned previously, we used the platform 'R' to compute the DT and RF classification models. The parameters used in Tables 8-6 and 8-7 are as follows:

- Libraries: CaTools, rpart, caret, performance analytics (these are the libraries required to build and analyze DT classification models in R), CaTools, randomForest, caret, performance analytics (these are the libraries required to build and analyze RF classification models in R).

- SET.SEED (123): The set.seed parameter is a random number inserted to ease reproducibility. It's not a compulsory parameter to use.

- SPLIT RATIO (0.75): The split ratio is the proportion of data allocated to the training set compared to the test (validation) set. As a rule, giving a higher proportion to the training dataset is recommended than the validation set to reduce the risks of overfitting (Rácz *et al.* 2021). The Split ratio can be represented as 2/3 in 'R.,' which means that 2/3 of the loaded dataset goes to the training set or as a percentage (recommended between 0.7 – 0.8).

- Type (class): This parameter is a compulsory one for DT classification model. It needs to be set to 'class' for classification models. Any other parameter setting would not work well for the model.

- ntree (500): We tested our RF classification model with ntree values from 1 – 500. We observed the best accuracies between 100 – 500 trees. The higher the number of trees, the best the accuracy but the longer the processing time. The choice of this parameter should therefore be a balance between the best accuracy required and the processing time to achieve. We were more interested with the best possible accuracy.

8.3.5 Classification results and discussion

We evaluate our classification results using the confusion matrix tool that presents the number of correct predictions achieved by our models versus the number of incorrect ones. The confusion matrix is also valuable for generating several other evaluation parameters such as the model percentage of accuracy, the precision, and the recall. We present the confusion matrices of the DT and RF classification models in Tables 8-8 and 8-9.

Table 8-8: DT ML classification model confusion matrix

	PGA	**PGB**	**PGC**
PGA	617	4	7
PGB	120	486	32
PGC	0	43	739

Table 8-9: RF ML classification model confusion matrix

	PGA	**PGB**	**PGC**
PGA	617	11	0
PGB	0	638	0
PGC	0	0	782

PGA, PGB, and PGC are the three product category classes we intend to predict in the product customization framework: Product group A (PGA), Product Group B (PGB), and Product Group C(PGC). The green cells in Tables 8-8 and 8-9 represent the number of correct predictions for each class. The numbers in the remaining cells (uncoloured) represent the number of incorrect predictions. Table 8-11 displays a summary of performance metrics results calculated by the confusion matrices outcome. These results enable us to determine the most performing classification model.

From the classification results summary in Table 8-10, we can detect that the RF ML classification model performs better than the DT model. We achieve an overall model accuracy of 99.4% with the RF classification model. It is 10% more than the one obtained for the DT model. A 10% accuracy difference is not negligible and automatically affects the overall production performance. It could imply that a product parameter entered by a client would be incorrectly mapped to the corresponding internal one, and the final customized product would not meet the customer's requirements, therefore, causing the customer's dissatisfaction.

Our proposed RF model produced a precision of 100% when predicting product categories, A and C and a precision of 98% for product category B prediction. A precision of 100% means that whenever our model receives parameter inputs from customers and predicts a final product in categories A and C, it is always correct. The RF model does not predict false product categories for these two product categories (no false positives exist). A precision of 98% implies that the model successfully predicts that a product belongs in Category B 98% of the time. The model has a 2% probability to predict an incorrect product category B (a product A or C could be mistakenly predicted as a product B). From the DT model, we produced precisions of 83.72% for product category A, 91.18% for product category B, and 94.99% for product category C. These results mean that the DT model predicts correctly products belonging to each category based on the percentages computed (83, 91 and 95% of the times). Our RF model offers better precision for each product category than the DT model. With the

proposed RF ML model, we achieved a recall of 100% to predict product categories B and C and a recall of 98% for the prediction of product category A. A 100% recall implies that our model did not incorrectly predict any product in categories B and C that did not belong to these categories (no false negatives exist). On the SVM model side, recalls of 98, 76, and 94% mean that the ML model could incorrectly predict a product (PGA, PGB, and PGC) to not belong to its actual category while it does based on the percentage computed.

Table 8-10: Classification models summary results

Classification Models	Label	Precision	Recall	Overall Accuracy
DT	PGA	0.8372	0.9825	0.899≈ 89.9%
	PGB	0.9118	0.7618	
	PGC	0.9499	0.9450	
RF	PGA	0.1000	0.9825	0.994≈ 99.4%
	PGB	0.9831	0.1000	
	PGC	0.1000	0.1000	

8.4 Overall product customization framework results summary and discussion

As part of the innovation areas of a small manufacturing plant in an I40 environment, we developed an adaptive product customization model based on two ML algorithms: XGBoost and RF. Using the ML algorithms, we built two predictive models (a regression model with XGBoost and a classification model with RF) daisy-chained in the product customization back end (the XGBoost model outputs are fed as one of the RF model inputs).

While the XGBoost regression model receives a personalized product parameter and predicts its corresponding parameter (a numerical value), the RF classification model determines the product category to which the customized client's input belongs. By applying these two ML models, we transformed a static database into an intelligent predictive tool that can receive any input and accurately group it to a product category without manually inputting the information before the run. Our research answered one of the I40 goals to improve production processes by

effectively utilizing data accumulated during the production lifecycle and creating more "intelligent" tools through it (Tao *et al.* 2018).

We rated the performance of our ML models by building two experimental product customization frameworks, as presented in Table 8-2. The first framework (Id-1) contains DT algorithms for the regression and classification problems. The second customization framework (Id-1) that we implement for our research contains the XGBoost (for regression) and RF (for classification) algorithms. We summarize the results of these two frameworks in Table 8-11. From the results, we notice that the second customization framework outperforms the first by reaching a classification accuracy of 99.4% for the RF model and a regression prediction error of about 0.03% on the XGBoost model. The first customization framework produced 89.9% classification accuracy on the DT model and about 33% error on the regression model built with DT. Through our research, we also experienced that the choice of the XGBoost and RF algorithms to build regression models on non-linear data depends on the aim of the application. XGBoost regression models produce more accurate results than the RF models, and RF regression models have faster processing speed than XGBoost models.

Table 8-11: Experimental product customization frameworks results summary

Customization Platform ID	**Machine learning model**	**Accuracy**	**Error**
1	DT regression		Up to 33%
	DT classification	89.9%	
2	XGBoost regression		Up to 0.03%
	RF classification	99.4%	

8.5 Chapter Summary

In this chapter, we developed an adaptive product customization framework that permits customers of small manufacturing industries in an Industry 4.0 environment to interact with a factory production system by sending their personalized products requirements before

production starts. We build our customization platform by implementing two ML models: XGBoost and RF. We presented and explained the different sections of our customization system: a front-end, a backend, and a bridge section. We also developed a system operational workflow and an overall system architecture displaying the different sections. We divided our adaptive customization backend into two sections: a regression model matching the clients' inputs to a factory internal parameter and a classification model detecting the final product group corresponding to the client's input. We analysed the theoretical principle of the XGBoost algorithm for the development of regression models and the RF algorithm for classification models. Using the XGBoost ML regression model, we improved the customization platform performance with the XGBoost algorithm feature for parallel computation. We reached an error rate of 0.03% in the experimental predictions. With the RF ML classification model, we obtained an accuracy of 99.4%, an outstanding result for prediction activities.

Chapter 9 : CREATING AN ENHANCED SAFETY MECHANISM FOR OPERATORS AND AMR USING Q-LEARNING ALGORITHM AND SPEECH RECOGNITION

We utilize some theoretical knowledge on Q-learning to develop a safety induction procedure for AMR working in a manufacturing environment with human operators. The safety induction generated from a Q-learning algorithm should teach the moving robot to choose the best obstacle-free trajectory to the closest safety exit in an emergency or evacuation. We also offer an additional way for operators to trigger the plant Emergency STOP (ESTOP) by incorporating a speech recognition signal theory into the active operations of a plant PLC program. This feature is very efficient when the operators are far away from any ESTOP buttons in an emergency or need to stop the system abruptly.

9.1 Reinforcement learning (RL), Q-learning (QL) and Speech recognition theoretical overview

9.1.1 Reinforcement learning (RL) algorithm

The RL (Sutton & Barto 1998*;* Kaelbling *et al.* 1996, and Wang & Usher 2005) algorithm is an ML concept developed on the principle that an "agent," which is the object using the algorithm to acquire the desired knowledge through a learning process, applies all possible actions and decisions repetitively in its power to establish the most effective one (the best strategy to adopt) (Jiang *et al.* 2004). The RL algorithm was created in the early 1990s. The RL algorithm operation depends on five main participants: the agent, the state, the reward, the action, and the environment. The algorithm operational principle encourages the agent to execute as many interactions as possible with the environment from which it is acquiring knowledge by performing different actions resulting in a change of states in the environment that correspond to a positive reward or a penalty (negative reward*) (*Wang & Hsu 2020). The agent does not know anything about its environment and only learns the effect of its decisions by the nature of rewards received (positive or negative). Therefore, the RL algorithm falls into the unsupervised category *(*Wicaksono 2011). *The* agent actions do not depend on previous expertise or data accumulated by an intelligent system that can build a decision-making model (Ou *et al.* 2018 and Weiss 1999). The agent only learns the best decision-making scheme through trials and errors (Ribeiro 2019). Various areas adopted the RL for applications like

Surgery (Hashimoto *et al.* 2018), Dam Management (Wang & Xu 2012), Chemical reaction (Zhou et al. 2017), Traffic-light Management, Resource Management (Mao *et al.* 2016), Traffic-light Management (El-Tantawy *et al.* 2013), Robotics (Gu *et al.* 2017) and Autonomous Driving (Leurent & Mercat 2019).

The RL algorithm is divided into value-based algorithms, also known as model-free algorithms, and model-based algorithms or on-policy-based algorithms. In value-based algorithms, the agent chooses the best strategy to adopt based on values recorded during the trial-and-error interaction process. It does not use modelled policy to act. The agent operations in model-based algorithms rely on a reward and a transition function to predict the type of reward it will receive and the state it will be in by making a specific decision (Chen *et al.* 2020).

The Q-learning algorithm is a well-exploited model-free algorithm in RL, especially for robotics applications. Model-free algorithms benefit from efficiently reusing data (Nguyen & La 2019). In this research, we utilize Q-learning algorithm capabilities to teach AMR to decide on the best obstacle-free in case of emergencies Model-free algorithms are suitable for robotics problems. We present in Figure 9-1 a graphical summary of the RL algorithm the working principle.

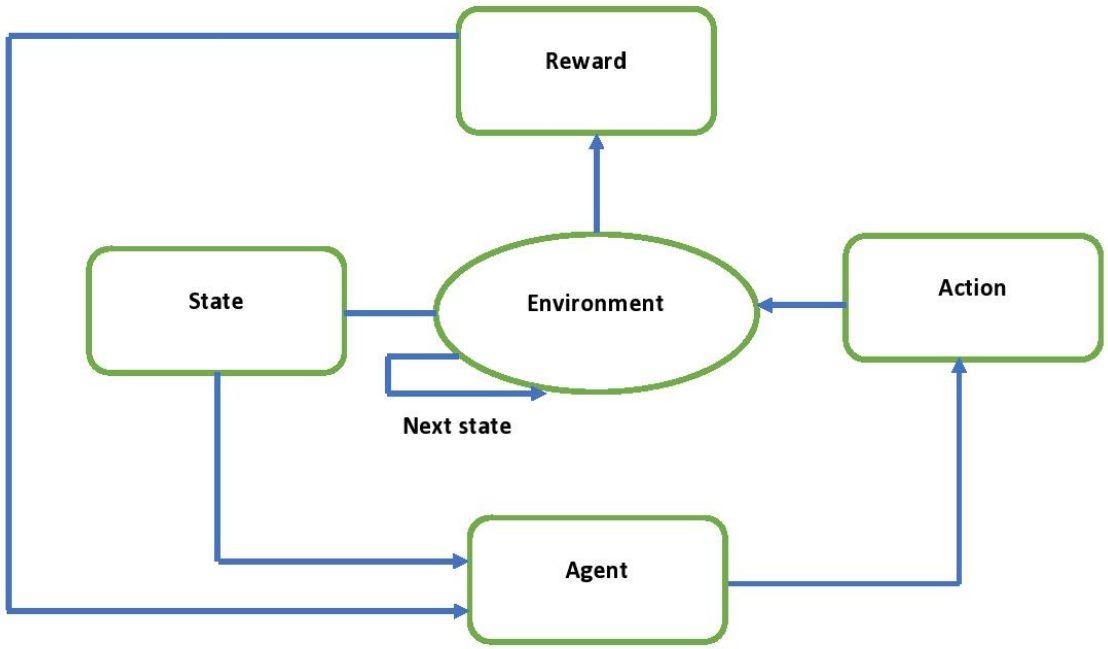

Figure 9-1: RL algorithm graphical operation summary (Su et al. 2019)

9.1.2 Q-learning (QL) algorithm to generate obstacle-free paths and its limitations

QL is a model-free RL algorithm that originated from the psychological behaviour model that encourages people with rewards for their excellent conduct or good actions and punishes them for the bad ones. The QL algorithm was initially proposed by Watkins & Dayan (1992). The algorithm presents several advantages, but the most known is its ability to start operating in new environments without previous awareness or knowledge of the environment. The agent accumulates environment knowledge through trial-and-error until it discovers the best decisions to make for each condition. Each decision and its corresponding outcomes are recorded in a Q table that becomes a reference to the agent for future actions during the learning process. The Q table format is a matrix that contains states of the environment in its rows and actions for each state in its columns. The Bellman equation (Zhao *et al.* 2020) computes values of the Q-table elements. From the populated Q-table, the agent chooses the action that offers the highest reward value. The reward value can be cumulative (Wang & Hsu 2020 and Zhang & Xu 2005). The QL algorithm provides effective solutions for Markovian decision problems (Haykin 2001) and is an incremental and dynamic programming technique.

An agent's behavior or action at a given time to change its state is often referred to as a *policy*. The policy is a function of the agent's action and state (Zhang et al. 2020). We present in (9.1) an expression of the agent policy probabilityβ:

$$\beta = a|s \tag{9.1}$$

where a represents the agent action at a specific time and s refers to the agent's state when performing the action a. We can summarize β as the probability the agent executes an action a with regards to its state s.

The Q-learning algorithm pairs all actions the agent takes to the outcomes of their equivalent state and computes values accordingly. The values representing every action and its states are called Q-values. Through the learning process, the agent performs the action that results in the largest Q-value, which entitles a higher reward. The algorithm updates the Q-table with generated Q-values until it reaches the best possible policy (Tang *et al.* 2020). The Q-learning algorithm produces Q-values using the Bellman equation utilized in a deterministic or non-deterministic environment. In a deterministic environment, we assume that the agent actions have 100% chances to produce desired results or states all the time. We display in (9.2) the Bellman equation

expression for the deterministic environment. In a non-deterministic environment, also known as a stochastic environment, we consider that the agent can experience difficulties preventing it from obtaining expected results. In other words, the agent could generate different states than expected. (9.3) is the mathematical expression of the Bellman equation for non-deterministic environments. This equation is equally known as the Markovian decision tree (Zhao *et al.* 2020).

$$G(s) = max_a(r(s,a) + \gamma G(s_n)) \tag{9.2}$$

where $G(s)$ represents the Q-value computed at a state s, $r(s,a)$ represents the reward value offered when performing an action a at a given state s, γ is the discount factor, an hyperparameter required for the computation of (9.2). The discount factor gives more weight to future rewards. $G(s_n)$ represents the next state s_n Q-value.

$$G(s) = max_a\,(r(s,a) + \gamma \sum_{s_n} P(s,a,s_n)\, G(s_n)) \tag{9.3}$$

where $P(s,a,s_n)$ is the probability an agent will successfully reach its next state s_n from a given state s when making action a. (9.4) is an expression of the final Q-learning model equation.

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha[r(s,a) + \gamma(max_{a_n} Q(s_n,a_n))] \tag{9.4}$$

where $Q(s,a)$ represents the Q-value obtained an action a and a state s of an agent, α is the learning rate, another hyperparameter like the discount factor γ, required to compute the Q-value. The learning rate establishes the effect of new information recorded versus old ones. The two hyperparameters values are between 0 and 1: $\alpha \in [0,1]$ and $\gamma \in [0,1]$ (Ribeiro *et al.* 2019), $r(s,a)$ refers to the reward value received by executing an action a at a state s, $Q(s_n,a_n)$ represents the Q-value calculated from a next action a_n at a next state s_n , and $max_{a_n} Q(s_n,a_n)$ is the highest Q value obtained from the next action and the next state.

Algorithm 1 is a summary of the Q-learning algorithm in (9.4).

Algorithm 1: Q-learning operation summary

		Initialize $Q(s,a) = 0$ and $Q(s_n,a_n)=0$
		Set hyperparameters α and γ
1	:	Record a state *s*
2	:	Choose an action *a* based on the state *s*
3	:	Examine the reward received $r(s,a)$
4	:	Record the next state S_n
5	:	Find the maximum Q value based on (s_n , a_n)
6	:	Find the value of $Q(s,a)$
7	:	Go back to 1

Some limitations of the Q-learning algorithm: The QL algorithm only produces outstanding results when dealing with discrete states and actions. We do not recommend the algorithm for continuous states. The algorithm processing speed relies on the size of the Q-table. A larger Q-table will result in a longer processing time and cause delays for the agent (Wicaksono 2011). Therefore, the Q algorithm is suitable for a relatively small-sized environment.

9.1.3 Speech recognition process

In I40 manufacturing environments, the interaction between humans and machines is substantially improving from simple machine light indications for monitoring purposes, from push buttons for operation controls, and Graphical User Interface (GUI) to more intelligent interfaces controlled by gestures and voices (Oviatt & Cohen 2000). These advanced interfaces are referred to as natural human-machine interfaces (NHMI). They utilized enhanced reality, gesture recognition, and speech recognition (Gorecky *et al.*, 2014). In this research, we focus on speech recognition technology to enhance a manufacturing plant emergency response.

The speech recognition theory identifies and recognizes information patterns present in a specific speech wave signal (Garcia *et al.* 2019). We present in (9.5) the statistical expression of the speech recognition (Li *et al.* 2016). Its goal is to predict the most accurate sequence of words B from the speech wave signal captured.

$$\hat{B} = argmax_B P_{\Psi,\lambda}(B|S) \tag{9.5}$$

where Ψ represent the acoustic model of the speech recognition and λ its language model. (9.5) can be further reshaped in (9.6):

$$\hat{B} = argmax_B p_\Psi(S|B)P_\lambda(B) \quad (9.6)$$

where $p_\Psi(S|B)$ refers to the probability of information quantity and quality coming from the acoustic model and $P_\lambda(B)$ refers to the one originating from the language model.

By considering a time sequence t in the transmitted signal wave and more observation b_t containing hidden states δ_t from the original word sequence, B, (9.6) can be transformed to a new express (9.7) using the hidden Markov models (HMMs).

$$\hat{B} = argmax_B P_\lambda(B) \sum_\delta \prod_{t=1}^{T} p_\Psi\,(b_t|\delta_t)P_\Psi(\delta_t|\delta_{t-1}) \quad (9.7)$$

A speech recognition system operation has two steps: the first, called the speech recognition front-end, produces the acoustic feature. It is a feature extraction module. The second step is the system back-end segment responsible for transferring the acoustic feature of the front-end into an acoustic and a language model. These two models will then generate the word sequence probability regarding the initial wave signal (Li *et al.* 2016).

9.2 Modelling of an enhanced safety mechanism for AMR and operators using Q-learning algorithm and speech recognition.

Our enhanced safety mechanism for AMR and operators in a small smart manufacturing environment contains two sections:

1) A section that computes obstacle-free paths for AMR to the closest safety exits of the manufacturing environment in case of emergency. We implement the Q-learning algorithm to design this first section.

2) A section that enables a Siemens S7-1200 PLC to receive a voice command from an operator to stop the factory's activities and generate an emergency signal. We apply a speech recognition process to activate this second section.

9.2.1 Assumptions

We make the following assumptions for the implementation of our enhanced safety response mechanism:

- The AMR has all required sensorial equipment such as sensors, global positioning system (GPS) modules, and cameras integrated into their housing to interact with field devices like the main plant PLC and are aware of their positioning or locations.

- The microphone installed to capture the human operators' voice command has a noise suppression module that makes it less sensitive to disturbances and has a broad voice capturing range (powerful microphone) to recognize the emergency voice command from a long distance.

- Within the factory, the AMR has its own travelling passage that does not obstruct human operators' paths.

9.2.2 Q-learning algorithm

We apply the Q-learning algorithm principles so that an AMR learns the best obstacle-free path to the closest safety exit every time an emergency requires it. The safety exit selection changes based on the AMR's current location. We develop an algorithm (Algorithm 2) that explains how the agent (the AMR) finds the best trajectory from its current location to the safety exit location. The generated path should have the location indexes of all areas through which the AMR goes through to reach its destination. The Q-learning algorithm operation also relies on Q values stored in the Q table.

Algorithm 2: Finding the obstacle free path from the current location to the safety exit

Initialize the system, set the Q table
1 : Read current location index
2 : Set the current location as *start*

```
3   :   Determine the closest safety exit location index
4   :   Set the closest safety exit as Destination
5   :   While (start! = Destination)
6   :       next location = Location [Max Q value   (start)]
7   :       route = start index + next location index
8   :       start = next location index
9   :   return route
10  :   End
```

9.2.3 Speech recognition process

We implement a speech recognition process to transform a traditional PLC into an intelligent voice-enabled controller that can use a voice command as an additional input for its programming routine. The voice command, in our research, is intended to stop the factory operation by being part of the PLC, stopping interlock. We apply the language and acoustic models from the Google API to design our speech recognition interface. The speech recognition system receives the factory operator's voice command and transfers it to the PLC as an action enabler to stop plant activities. Algorithm 3 is an outline of the speech recognition system interfacing with the factory PLC.

Algorithm 3: Voice command to PLC input via speech recognition

```
    Import all important libraries (speech recognition and PLC
    connection)

1   :   Define write and/or read functions for PLC
2   :   Establish connection with the PLC
3   :   while (True)
4   :     Listen to Microphone (sound source)
5   :     Recognize the language using Google API
6   :     if language recognized:
7   :       Go to 11
8   :     else:
9   :       print ("Please speak again")
10  :       Go to 4
11  :     if language == "emergency command":
12  :       write instruction to PLC variable
13  :     else:
14  :       Go to 4
```

```
15 : end
```

We present a summary of the overall enhanced safety mechanism, in Algorithm 4**,** as a combination of the Q-leaning algorithm method for AMR to learn obstacle-free paths to safety exists and the speech recognition system for the voice command instruction the PLC. We made the following assumptions for the algorithm's variables:

1) θ represents the evacuation or emergency signal
2) X represents the AMR
3) η represents one of the safety exits

Algorithm 4: Safety mechanism procedure summary

```
1 : Activate θ
2 : Send θ to X from controller (PLC)
3 : X initiates the safety routine
4 : X determines the closest η
5 : X finds the obstacle free path to η
6 : X reaches η
7 : end
```

Figure 9-2 is a representation of our experimental smart manufacturing location overview with a legend providing details on the manufacturing environment components. The AMR is in motion from one location index to another when performing its manufacturing tasks. When the AMR receives an evacuation signal, it should compute from its current location the obstacle-free trajectory to the closest safety exit. Our Q-learning algorithm section allows the AMR to enable this function. From Figure 9-2**,** the safety exits indexes are E, G, S, and W. We assume

that the emergency or the evacuation signal can be generated by a voice command input into a microphone and directly controlling a Siemens S7-1200 PLC program or by an ESTOP push button wired on one of the factory control panel or motor control centre (MCC).

We represent in Figure 9-3 the architecture of the voice command reception to the PLC program control. Figure 9-4 summarizes our enhanced safety response mechanism workflow when receiving the emergency signal to evacuate the factory.

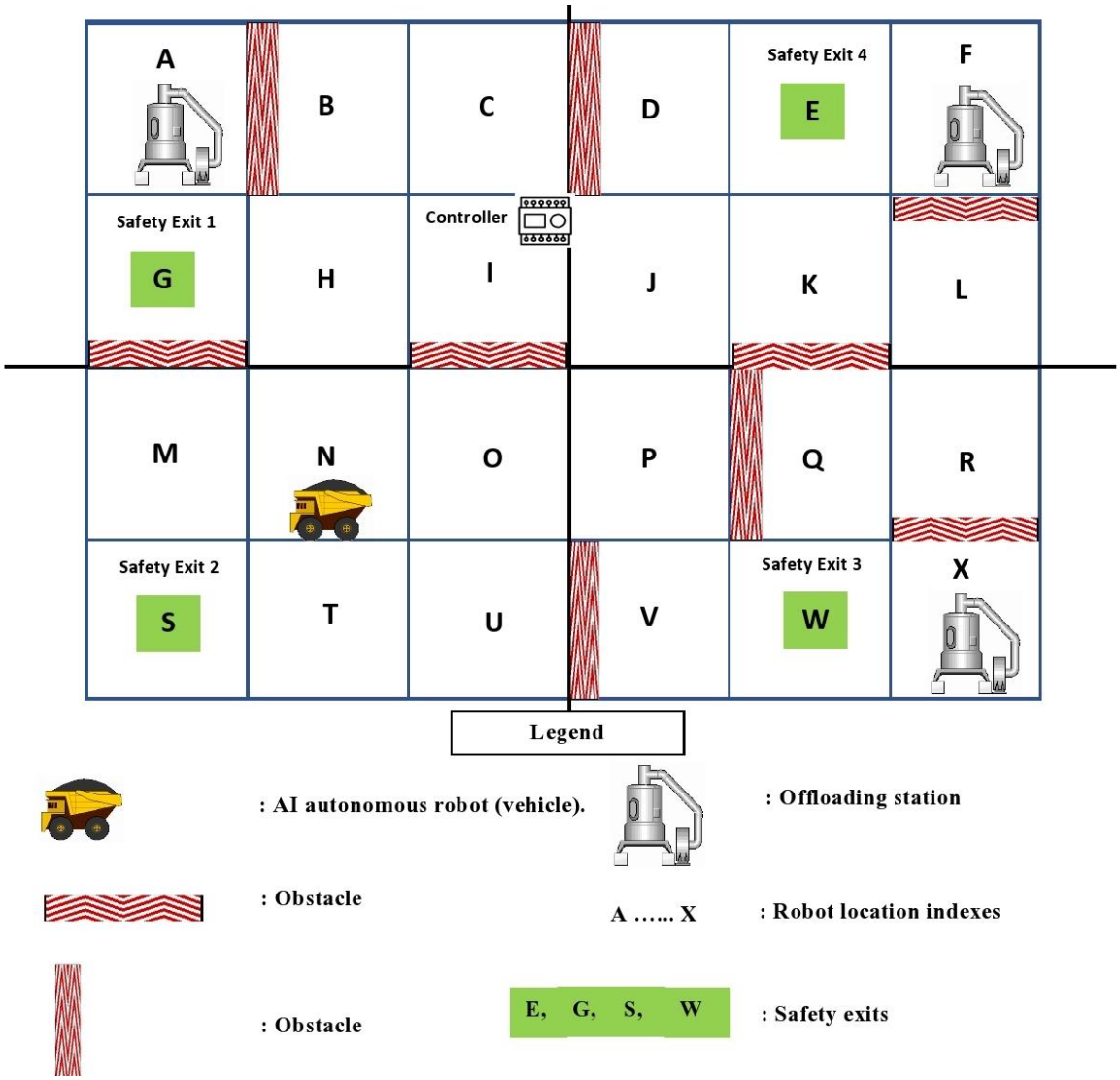

Figure 9-2: Experimental manufacturing plant location overview (Kiangala & Wang 2021 c)

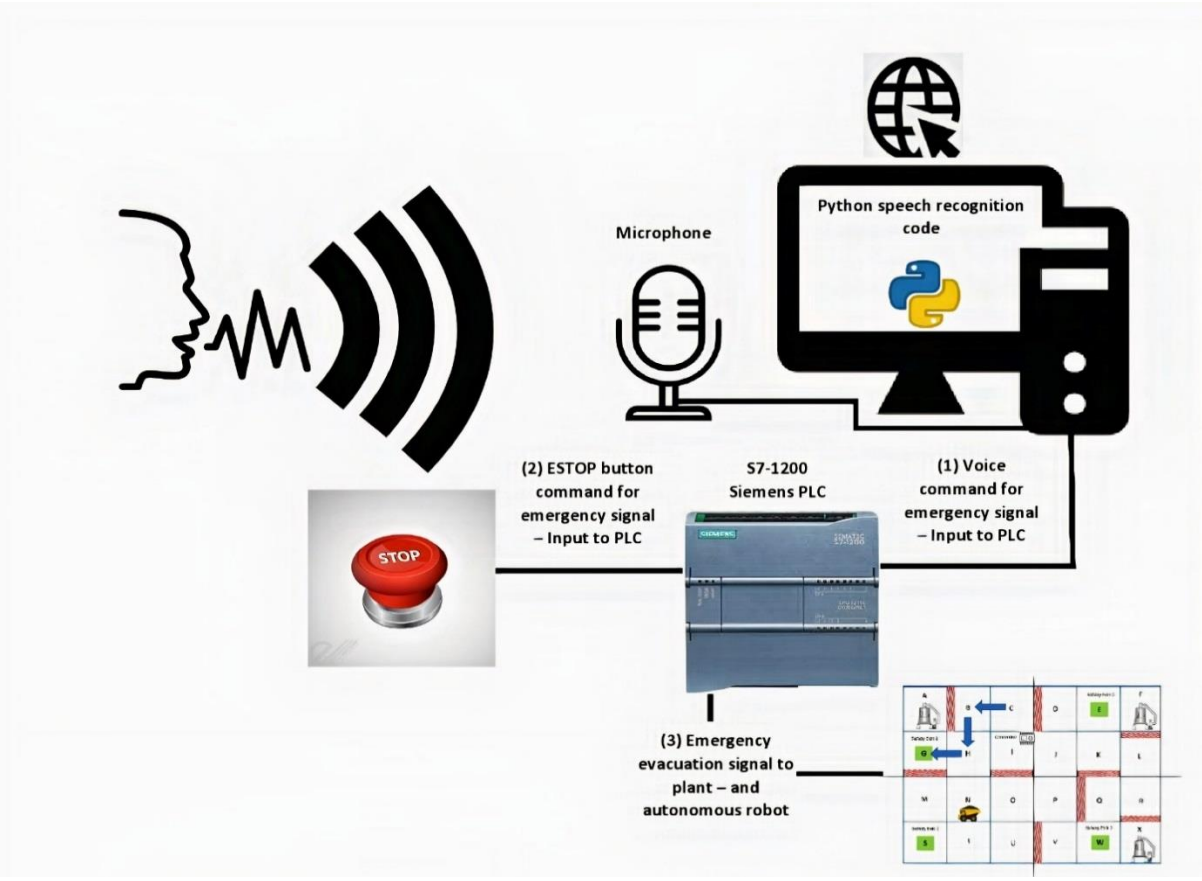

Figure 9-3: From voice command to PLC program control (Kiangala & Wang 2021 c)

9.3 Implementation of an enhanced safety mechanism for operators and AMR using Q-learning algorithm and Speech recognition

9.3.1 Experimental background

- Every industrial plant built according to strict safety standards should have implemented safety protocols to deal with emergencies and disasters. During factory operations, the emergency stop interlocks (ESTOP buttons or Stop buttons depending on the severity of the emergency event) would usually abort all ongoing activities. In some other cases, they would start an alarm for evacuation of the plant. The stop interlocks are often controllers' inputs hardwired to the stop system and programmed in the controller to reset outputs. In this experiment, we use a PLC as the plant central controller. As part of the safety protocols, the emergency stops buttons (ESTOPs) are

positioned in areas accessible (in front of controller panels) and sometimes in several strategic locations.

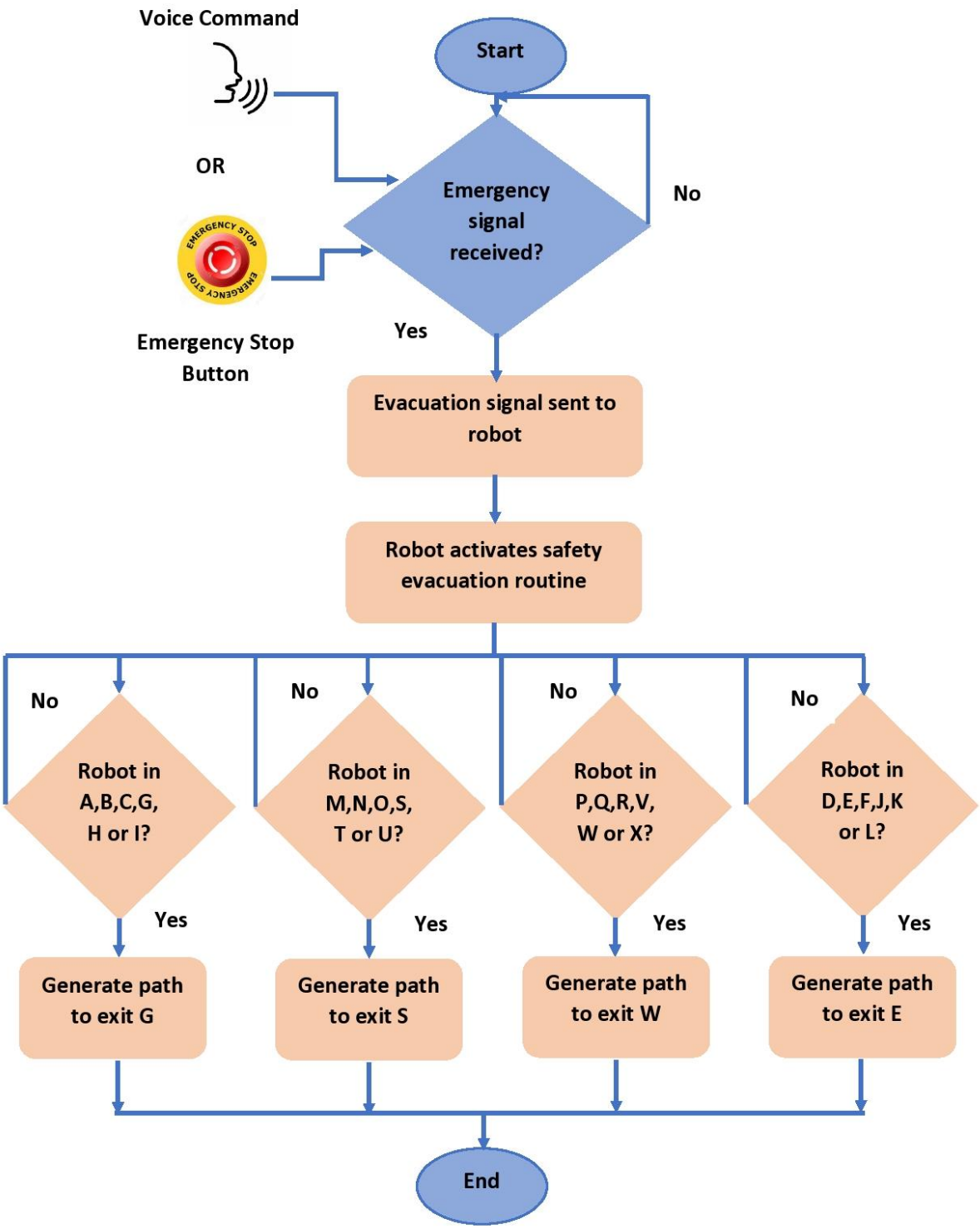

Figure 9-4: Enhanced safety response mechanism operation process flow (Kiangala & Wang 2021 c)

In order to enhance safety protocols in factories, we research methods to deal with emergencies where none of the operators can physically access the ESTOPs at the time of the disaster.

We solve this concern by integrating a voice command to the factory emergency stop interlock programmed in the PLC. The operators no longer solely depend on the physical ESTOPs to abort plant operation but can also utilize the voice command "emergency" through a microphone directly to the PLC program. We implement a speech recognition algorithm to transform a voice "emergency" command to a stopping PLC's instruction. We test the speech recognition system on a Siemens S7-1200 PLC programmed with the TIA portal software. We generate and compute the speech recognition code in python.

- Another safety standard in large modern buildings is to have various safety exits at strategic locations to ensure accessible and orderly evacuations in case of emergencies. Having a single emergency exit in a massive building could be very dangerous for people during the evacuation. Everyone would be in a panic to reach the single point of exit that could be located very far from their current location, with more risks of injuries along the way.

In the I40 environment, operators and intelligent robots work together as the factory workforce. While operators go through regular safety inductions to be aware of safety responses and safety exists, intelligent robots like AMR are usually not prepared to respond to disasters. How should AMR react to evacuation signals or emergencies when performing their assigned tasks in the plant?

We implement a safety induction procedure for AMR to learn the closest safety exists to their current locations in emergencies. We simulate a manufacturing environment with safety exits at different locations. The AMR, based on its current location, can choose to go to one of these exits. The robot finds its intelligence in a Q-learning algorithm that we apply to learn the obstacle-free trajectory from its current location to the closest exit upon receiving the evacuation signal. Like with human operators, we lessen the likelihood of undesired incidents on AMR in emergencies. We program the Q-learning safety procedure in Python.

9.3.2 Using a voice command to control the Emergency stop interlock of a S7-1200 PLC Siemens program

Every manufacturing plant usually has an ESTOP system wired to stop all running outputs and programmed in the controller's CPU to interrupt operations in an emergency needing a quick stop. In most industrial environments, the ESTOP is installed in front of the controller panel, easily accessible for the operator to stop operations whenever required. We suggest an additional stopping interlock via voice command for operators to have the flexibility of interrupting operations with the voice even when they are far from the controller panel. Most PLCs installed in factories (traditional PLCs) cannot receive voice commands as inputs. We need to implement an external mechanism to integrate the voice capability into a PLC system. Some existing researches utilize third-party controllers such as Raspberry Pi to accept the voice command and give instructions to the PLC. Our study does not use any third-party controller to manipulate the voice command. We extract the speech signal from a microphone straight to the Siemens PLC through a speech recognition program in Python. We simulate the voice command operation by interacting with an S7-1200 Siemens PLC in which we write a simple ladder logic program for the start/stop interlock as presented in Figure 9-5**.** Table 9-1 explains the function of each variable.

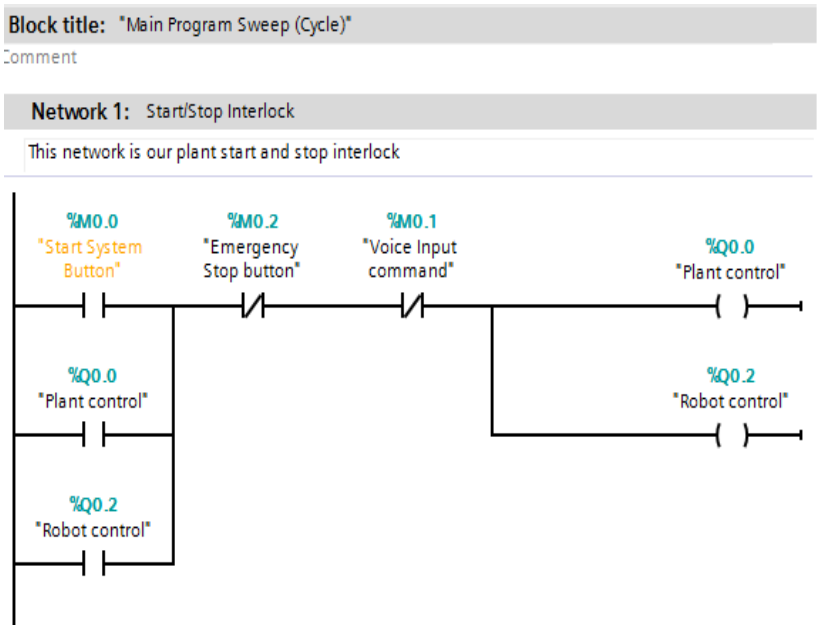

Figure 9-5: Starting and stopping interlock PLC ladder program

Table 9-1: PLC Starting and Stopping interlock variables functions

Variable address	Variable function
M0.0	This is the start signal for the system. It is usually a ON-OFF action directly wired to a PLC input IX.X. For this simulation, we use a memory bit M. When the memory bit is ON the system starts if M0.1 and M0.2 are OFF.
M0.1	This is the memory bit controlled externally by our python speech recognition code that converts the voice command to a 'ON' bit value. When M0.1 is ON the system is in stop mode.
M0.2	This is the memory bit controlled by an emergency stop signal. In real plant applications it is usually an input signal IX.X. For simulation, we use a memory bit variable M.
Q0.0	This is a physical PLC output that controls the plant.
Q0.2	This is a physical PLC output that controls the control plants robots.

We program the starting and stopping interlock of our experiment in a "latch" structure so that a single push button signal (from OFF to ON) runs the system (through the variable M0.0), and the outputs (Q0.0 and Q0.2) will remain ON (running) until one of the stop commands (M0.2 or M0.1) is enabled. Table 9-2 summarizes necessary experimental settings to interact with the Siemens PLC starting and stopping interlock code to the speech recognition program in Python.

Table 9-2: Siemens S7-1200 PLC settings for speech recognition integration

Settings	Values	comment
PLC IP address and MAC address	172.16.2.254	This is the experimental IP address; it can be adapted to any other IP address
Python PC IP address and MAC address	172.16.2.202 (with internet access)	This PC needs internet connection to connect to the Google speech API
PLC security/protect level	Permit access with PU/GET communication – No protection	This is to allow the external python code to modify the variable value
Speech recognition variable	M 0.1	This needs to be a global memory variable

In order to successfully convert an operator voice command into a PLC input signal, we apply

a speech recognition Python code (using a Google API) and a Siemens PLC library that we install in the Python IDE. This specific library facilitates the communication between the Python code and the Siemens PLC program. We install another essential library: a speech recognition library to translate the voice signal into a text syntax. We use the converted text in our code to manipulate the PLC variable M0.1 switching it ON or OFF. We present in Table 9-3 the experimental settings configured in the Python IDE and its code to successfully communicate with the S7-1200 PLC.

We display in Figure 9-6 the starting and stopping interlock program of a healthy running system in the Siemens S7-1200 PLC (when no stop signal is received) monitored live from the TIA portal software. We explain the value and state of each variable of Figure 9-6 in Table 9-4.

Table 9-3: Python IDE settings for speech recognition process with a Siemens S7-1200 PLC

Settings	**Values**	**comment**
Python version	Idle (Python 3.7 32-bit)	
Libraries	Python-snap7 speechrecognition wheel pyttsx3 pipwin pyaudio playsound gTTs	These libraries need to be installed in the Python environment used.
plc = c.Client()	plc.connect('172.16.2.254',0,1)	This is the IP address of the PLC
r.pause_threshold	0.5	The Speech recognition system listens after 0.5 seconds
Voice source	Microphone () as Source	Select the Microphone as the source of the voice
Write Command	WriteMemory(plc,0,1, S7WLBit,1)	Writing command to M0.1 variable

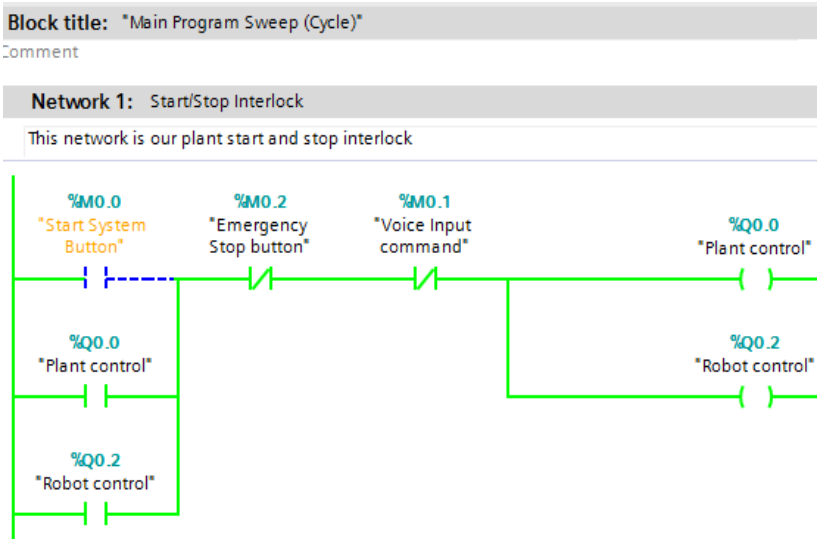

Figure 9-6: Healthy running plant program (with no stopping interlock)

Table 9-4: PLC healthy running system start-stop interlock variables values meaning

Variable	Values	comment
M0.0	'0'	Because of the latch programming format. M0.0 does not have to remain ON ('1') for the system to run continuously.
M0.1	'0'	No stop signal has been sent by the voice command.
M0.2	'0'	No stop signal has been sent by the emergency stop button
Q0.0	'1'	The plant control signal has been activated
Q0.2	'1'	The robot control signal has been activated

Whenever the operator gives the "emergency" voice command through the microphone, a signal is sent directly to the PLC to the starting and stopping interlock program to stop the system. In Appendix 1.D., we display a segment of the Python speech recognition program and its dependencies for a successful voice signal transfer to the S7-1200 PLC.

In the Siemens PLC, after a successful transmission of voice command in the Python code, a stop signal is activated via the variable M0.1**,** as displayed in Figure 9-7. The system operations now stop. We describe the meaning of variables' updated values after the reception of the stop signal in Table 9-5.

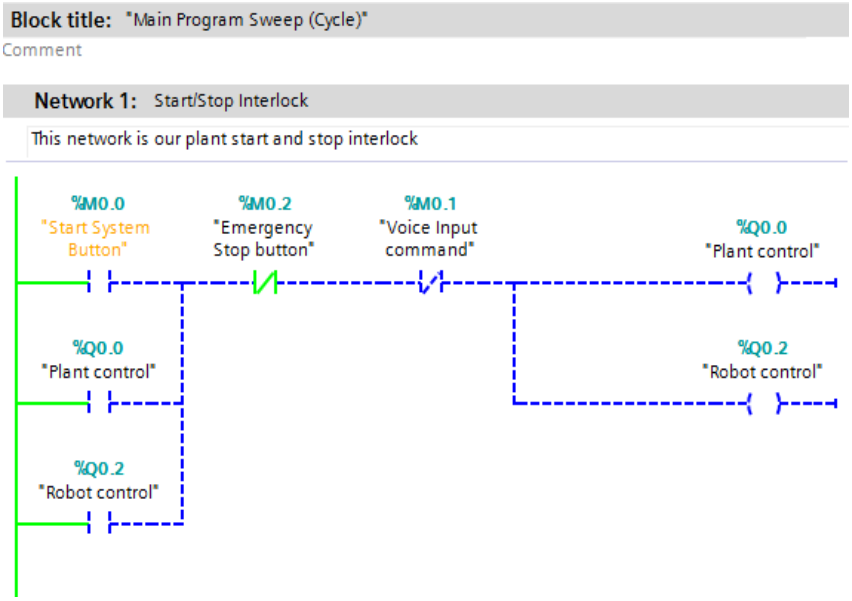

Figure 9-7: PLC system stopped after emergency voice command received

Considering that the voice input signal is also in the PLC code as an evacuation signal for the factory, the same signal will be sent to the AMR (via Q0.2 turned OFF) to activate its emergency procedure to evacuate the plant through a trajectory to the closest safety exit.

Table 9-5: PLC stopped system start-stop interlock variables values meaning

Variable	Values	comment
M0.0	‘0’	Because of the latch programming format. M0.0 does not have to remain ON (‘1’) for the system to run continuously.
M0.1	‘1’	An external voice command is received to stop the system
M0.2	‘0’	No stop signal has been sent by the emergency stop button
Q0.0	‘0’	The plant control signal is stopped.
Q0.2	‘0’	The robot control signal is stopped.

9.3.3 Finding AMR trajectories to the closest safety exits using Q-learning algorithm

We simulated an experimental manufacturing environment in Figure 9-2 where an AMR travels from one location to another, performing its assigned duties. We implement the Q-learning algorithm to the robot operation to learn and to decide on the obstacle-free path it should undergo to reach the closest safety exit based on its current location. This procedure can be considered a safety induction for AMR evacuation and is activated when the robot receives an evacuation signal from the factory. Our simulated manufacturing environment contains fixed obstacles that the AMR should avoid while finding its way to the closest exit. We assume that the AMR does not know anything about the safety exits and implement the Q-learning algorithm to make it learn with the principle of reward and punishment, the best way to a safe exit. In the design of our manufacturing environment, we split the factory into four areas with one safety exit each. We display the four areas' safety exits in Tables 9-6, 9-7, 9-8, and 9-9.

Table 9-6: Factory area 1 locations and safety exit indexes.

Locations	**Safety exit location**
A	G
B	G
C	G
G	G
H	G
I	G

Table 9-7: Factory area 2 locations and safety exit indexes.

Locations	**Safety exit location**
M	S
N	S
O	S
S	S
T	S
U	S

Table 9-8: Factory area 3 locations and safety exit indexes.

Locations	Safety exit location
P	W
Q	W
R	W
V	W
W	W
X	W

Table 9-9: Factory area 4 locations and safety exit indexes.

Locations	Safety exit location
D	E
J	E
K	E
L	E
F	E
E	E

The Q-learning algorithm does not understand the location letter indexes format. A crucial step of our design is to convert our experimental factory into a matrix R of size m x m, where m is the number of locations of the manufacturing environment. We apply the Q-learning algorithm to the new matrix format representing the simulated manufactory factory. As per our factory in Figure 9-2, the size of the matrix R (m x m) is as follows:

$$m = \{A, B, C \ldots, X\} = 24$$

$$R_{size} = 24 \times 24$$

Our initial matrix R values are mostly '0s' and '1s'. A '1' means that two locations (from the intersection of a row and a column) are neighbours, and there are no obstacles between them. A '0' represents an obstacle between two locations or two locations that do not share a border. The only value '10' designates the safety exit location for one factory area. The value that represents the safety exit can be any value higher than' 0' and '1' for the Q-learning algorithm to give more priority to the safety exit location. We display in Figure 9-8 the initial R matrix for our experimental factory (Figure 9-2) for the first area (Table 9-6) before implementing the

Q-learning algorithm. After computing the Q-learning algorithm, the matrix R in Figure 9-8 changes into another matrix displayed in Figure 9-9 with new reward values. Considering factory area 1, the reward value at the safety exit location G has increased from 10 to 33. When looking for the obstacle-free path to reach the closest safety exit, the AMR will aim for locations offering the best rewards until it gets to the safety exit with the highest reward value. Using the Q-learning algorithm in (9.4), we substitute the Hyperparameters' values as $\alpha = 1$ and $\gamma = 0.7$. Figure 9-9 is the Q-learning matrix for factory area 1. We implement a similar procedure to generate the Q-learning matrices for the three other areas. When applying the Q-learning algorithm in other factory areas matrices, the main difference is the change in safety exit locations. In factory area 2, the exit changes from location G to location S. In factory area 3, the new safety exit is location W, and in factory area 4, the safety exit location is E.

We present in Table 9-10 the rewards values between all locations of factory area 1 according to the matrix in Figure 9-9.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
A	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	1	0	0	0	0	0	10	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	0	1	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
I	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
K	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
N	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0
P	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
R	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1
X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Figure 9-8: The initial environment matrix for factory area 1 before Q-learning

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
A	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	0	10	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	13	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	0	0	0	0	6	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	0	0	0	8	0	5	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	17	0	0	0	0	0	33	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	0	13	0	0	0	0	24	0	13	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0
I	0	0	10	0	0	0	0	17	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J	0	0	0	8	0	0	0	0	13	0	8	0	0	0	0	8	0	0	0	0	0	0	0	0
K	0	0	0	0	6	0	0	0	0	10	0	6	0	0	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	5	0	0	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	8	0	0	0	0	0
N	0	0	0	0	0	0	0	17	0	0	0	0	10	0	10	0	0	0	0	10	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	8	0	0	0	0	8	0	0	0
P	0	0	0	0	0	0	0	0	0	10	0	0	0	0	10	0	0	0	0	0	0	6	0	0
Q	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	5	0
R	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	5	0	0	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	10	0	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	8	0	8	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	10	0	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	5	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	6	0	5
X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0

Figure 9-9: AMR Q-learning matrix for safety response in factory area 1

Table 9-10: Factory area 1 locations reward values

Travel locations	Reward value	Comment
From A to B or from B to A	0	The reward value is 0 because of the obstacle between the two locations
From A to G	24	
From G to A	17	The difference in reward is because going from G to A, the robot moves away from the safety location (G) but from A to G, it moves closer hence a higher reward.
From B to C	10	
From C to B	13	The difference in reward is because going from B to C, the robot moves away from the safety location (G) but from C to B, it moves closer hence a higher reward.
From C to D	0	The reward value is 0 because of the obstacle between the two locations
From C to I	13	

From I to C	10	the difference in reward is because going from I to C, the robot moves away from the safety location (G) but from C to I, it moves closer hence a higher reward.
From I to J	10	
From I to O	0	The reward value is 0 because of the obstacle between the two locations
From I to H	17	
From H to I	13	The difference in reward is because going from H to I, the robot moves away from the safety location (G) but from I to H, it moves closer hence a higher reward.
From H to N	13	
From H to G	24	
From G to H	13	The difference in reward is because going from G to H, the robot moves away from the safety location (G) but from H to G, it moves closer hence a higher reward.
From G to M	0	The reward value is 0 because of the obstacle between the two locations

The AMR selects locations with the highest rewards when learning its path to the safety exit. It cumulates rewards values until it gets to its destination. In order to display the location indexes selected by the AMR, we apply Algorithm 2 in Python. The selected path starts from the current location of the AMR is when it receives the evacuation signal. We simulate the path selection by inputting any factory locations and displaying the trajectory computed by the AMR. Figure 9-10 presents one of our simulation results. The path generated by the AMR in factory area 1 when its current location is equal to ‘C’ (variable loc = ‘C’). We also display the AMR graphical path of Figure 9-10 in Figure 9-11 with the experimental manufacturing environment.

The AMR procedure to reach the safety exit from location C can be summarized using Algorithm 2 from row C, the highest reward value is ‘13’ for locations B and I. Since the reward value is the same, the AMR can decide to go via any two locations: B or I. B is the first location in the Q table before I; it has the preference. From C, the following path will be C-B. B is now the new starting position from which the AMR intends to reach the safety exit. We follow the same procedure to determine the following location it should go through by checking on the highest reward from row B. From Figure 9-11, we notice that the maximum reward in row B is the value

‘17’ in column location H. Therefore, H is selected as the following location after B until the AMR gets to G. As per Figures 9-10 and 9-11, the final trajectory displayed is C-B-H-G.

```
#Print the route based on the selected location

if loc =='A' or loc =='B' or loc =='C' or loc =='G' or loc =='H' or loc =='I':
  print(route(loc,'G'))

elif loc =='M' or loc =='N' or loc =='O' or loc =='S' or loc =='T' or loc =='U':
  print(route(loc,'S'))

elif loc =='P' or loc =='Q' or loc =='R' or loc =='V' or loc =='W' or loc =='X':
  print(route(loc,'W'))

else:
  print(route(loc,'E'))
```

```
['C', 'B', 'H', 'G']
```

Figure 9-10: AMR obstacle free path computed from location C

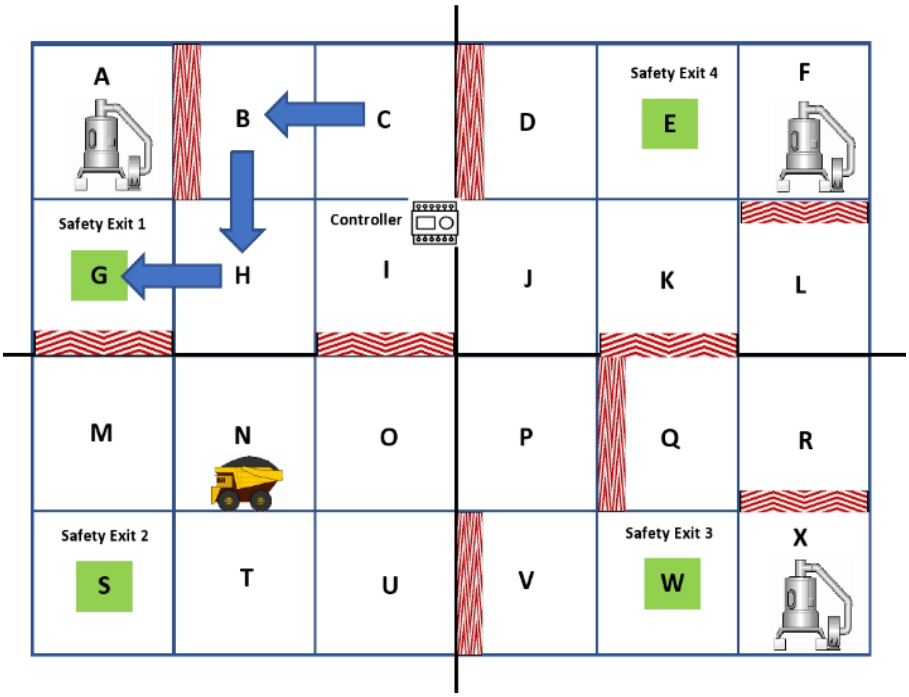

Figure 9-11: Graphical obstacle free path selected by AMR from location C (Kiangala & Wang 2021 c)

9.4 Result summary and discussion

We designed an enhanced safety procedure for the operations of a smart manufacturing plant in an I40 environment as part of its innovation areas. Every manufacturing organization aims to establish a safe working environment to smoothly run their production activities and reach their targets (Dabous, S.A. *et al.* 2021). In a smart manufacturing environment where human operators and intelligent machines such as robots work together, the concept of safety becomes broader and involves the protection of operators and robots collaborating with them. Our enhanced safety procedure provides a safety induction model for AMR working in a factory to learn, through the Q-learning algorithm, the obstacle-free path to the closest safety exit in case of plant evacuation or emergency. We also incorporated a voice command signal as an added input to the factory controller (PLC) to abort plant activities in an emergency. This last function allows operators to respond timeously to emergencies, especially far away from physical plant ESTOPs.

In order to test our system's reliability, we simulated a manufacturing environment with a total of twenty-four locations (from A to X) through with an AMR travels back and forth to accomplish its daily duties. We divided the plant into four areas and implemented the Q-learning algorithm that attributed various rewards to the locations, with the highest reward given to the safety exit location index. When learning the trajectory to the exit, the AMR accumulates the best rewards values on its way until it reaches its destination. Our Q-learning method's simulation results demonstrate that an autonomous robot, on its own, finds the obstacle-free path to the closest safety exit from any location in case of an emergency.

We improved the emergency response process of the manufacturing plant by implementing a speech recognition program that receives a voice command from an operator and behaves like a stopping button for the whole factory via its central PLC program. For simulation purposes, we used the word “emergency” to reset all outputs. The speech recognition program developed in Python is connected to a microphone to receive the signal and convert it to text. The communication between the Python code running on a PC and the plant PLC (A Siemens S7-1200) does not use any intermediate controller. We installed specific libraries in Python and configured the PLC accordingly to create a proper communication channel. Our solution reduces the risks of delayed actions in emergencies due to the lack of proximity with the factory ESTOPs buttons.

9.5 Chapter summary

In this chapter, we developed and designed an enhanced (experimental) safety response mechanism for a robot operating in a small manufacturing factory. Our safety mechanism generates a free obstacle path for the AMR to the closest exit in an emergency or evacuation. We used the Q-learning RL algorithm to compute the best path a robot could choose to the closest safety exit based on a trial-and-error principle and collect the best possible rewards along the path to the safety exit. We also added a new stopping interlock for a factory PLC (a Siemens S7-1200) via voice command to increase the operator's ability to stop the system in case of emergency without because closer to the plant emergency stop button. We capture the voice command through a computer microphone connected to the PLC network and run a Python script to process the speech recognition system without any additional equipment. We went through theories on the RL algorithm, the Q-learning algorithm, and the speech recognition system that we utilize to produce our enhanced safety response mechanism. We explained the principal enhanced safety mechanism segments and their inner compositions. We developed a process algorithm enabling AMR to learn obstacle-free paths to the closest safety exists in case of evacuation, a speech recognition process algorithm for activating an evacuation signal to the central PLC using a voice command, and a final process algorithm merging the two segments of the safety mechanism. We also designed the experimental manufacturing environment with a location overview to test the enhanced safety mechanism, a speech recognition conversion graph for our system, and an operational workflow for the AMR to determine an obstacle-free trajectory to safety exits.

Chapter 10 : CONCLUSION, RESEARCH SUMMARY AND FUTURE RECOMMENDATIONS

10.1 Research summary and discussion

From our research objectives, we achieved the following:

- The creation of a small innovative conceptual business model that transforms traditional organization structures into dynamic workplaces encouraging innovation. The designed business model is a foundation to ensure a durable and successful implementation of advanced and innovative techniques within manufacturing SMEs. Some essential features of the innovative business model are a continuous collaboration between production stakeholders (management, employees, and customers), active integration of customers' inputs to the production process, and adequate training for all production stakeholders.

- The selection of various advanced tools, techniques, and systems capable of increasing manufacturing production performances when successfully implemented. These techniques are PM, robust communication networks, automation of repetitive tasks, product customization, and enhanced safety mechanisms.

- The optimization, adaptation, and application of these processes and tools for manufacturing SMEs environment using advanced technological concepts and some of the current state-of-the-art technologies such as AI and ML. We empowered the PM framework with CNN (an AI algorithm); we implemented TSN, edge computing, and zero-loss redundancy protocols technologies to create a robust communication network infrastructure; we automated parameters configurations of a SCADA system by merging MLR and DT ML algorithms; we designed an efficient product customization scheme for a traditional manufacturing plant using XGBoost and RF ML algorithms; we implemented an enhanced safety response system for a manufacturing factory by integrating speech recognition as an additional stopping interlock for a plant PLC and by creating a safety induction scheme that teaches AMR how to evacuate their working environment (to the closest safety exit) in case of emergency using the Q-learning algorithm.

We utilized the DSRM to conduct this research through which we designed, demonstrated, and assessed several optimized AI and advanced technological trends to build innovative solutions for manufacturing SMEs. The innovative solutions we built are PM, communication network infrastructure, automation of repetitious functions, product personalization, and safety response procedures. Unlike the few existing studies that focused on a single research aspect (technological or organizational) when suggesting innovative solutions for manufacturing SMEs, our research merges organizational with technological changes to provide a more practical guide for the successful application of new technologies. We mapped our proposed optimized, innovative solutions into the designed conceptual business model.

As highlighted in chapter 1, the lack and rareness of SMEs-customized practical applications for advanced and innovative technologies slow down the embracement of new technologies by SMEs as on their own they are limited in terms of financial and HR they could invest on. It is one of the greatest motivations of this research. In this same chapter, we set up the research questions and objectives, which lead us to reach our research goal. Figure 10-1 is an illustration of the research questions and objectives and how our study addressed them. In chapter 2, we conducted a background and literature review on essential concepts that, actively or passively, contributed to building our innovative solutions: I40, AI, ML, SMEs, PM, product customization, business models, and much more. The durable implementation of new technologies in businesses depends highly on the type of organization structure in place. In chapter 3, we emphasize a new conceptual business model that encourages innovation and new technologies. We list and give a short highlight on the different innovation areas developed with our optimized AI and advanced techniques in chapter 4, focusing on data collection and analysis as the foundation of the optimized AI techniques. From Chapters 5 to 9, we describe the design, implementation, and experimental testing of our developed AI techniques and innovative methods. In chapter 5, we cover creating an intelligent PM framework for conveyor motors using CNN. We develop an improved network infrastructure for a small industrial factory by implementing technologies like TSN, edge computing, and zero-loss redundancy protocols in chapter 6. We implement an automatic parameter configuration scheme that transforms a traditional SCADA system into a self-configurable device empowered by DT and MLR ML models in chapter 7. In chapter 8, we design an adaptive product customization platform that allows customers to interact with a small factory production system by sending their personalized product requirements before production begins. We use two ML algorithms,

XGBoost and RF, as the core of the customization platform backend. In chapter 9, we introduce a safety response mechanism that provides an AMR with the safest obstacle-free trajectory to an exit in case of an emergency or evacuation. Our safety response mechanism also adds an interlock for a PLC plant by incorporating a voice command as a new way for the operator to stop the plant's operations in case of an emergency without being closer to one of the emergency stops. In each of these chapters, we start by theoretically studying the algorithms and other technical concepts required for each innovation area. We develop operational flows and procedures to implement them practically. Moreover, we evaluate the results of our optimized technique by comparing them and their impact on several other existing algorithms or solutions.

Our research contributes: to society by offering manufacturing SMEs, who constitutes the pillar of several countries' economies, a practical guide (combining organizational and technical guidelines) to sustainably implement new technologies that have the potential to improve their operations significantly; in research by augmenting additional knowledge and literature to the field of practical applications for SMEs, in need of more materials, opening paths for further researches in this regard; in theory by providing practical means to convert abstract theoretical concepts such as ML algorithms into practical solutions for businesses, linking the gap between theoretical knowledge and practice.

10.2 Future recommendations

As we can observe from the conceptual business model displayed in Figure 4-1**,** our research did not cover all innovation areas listed. Future studies should be conducted to complete the design and demonstrate innovative areas such as decentralized, remote control, and monitoring of production processes, and digitalization of HR, employee management, and financial system linked to real-time production feedback. In this same line of ideas, more innovation areas can be populated in the innovative business model to match the specific requirements of different manufacturing businesses. Our research only provided a sample that can be adjusted and adapted to various other manufacturing SMEs' specifications. Further researches should be done to apply our proposed conceptual business model and the innovative technical solutions into various manufacturing SMEs sectors such as textile, food and beverage, packaging, and much more, to collect practical outcomes, differences, and abnormalities and suggest an improvement for a more robust system.

We built most of our ML (AI) models (CNN classification, MLR regression, DT regression, XGBoost regression, and RF classification) utilizing a relatively small amount of data considering SMEs' environment. The choice of ML algorithms and the results obtained depends highly on the training dataset loaded. More researches should consider larger and different datasets types to accommodate the future growth and diverge manufacturing industry. A larger and different dataset type would affect the ML outcomes and the type of algorithms implemented. For example, when considering the CNN classification model in our intelligent PM framework, studies could consider including non-linear time-series input data type and implement Kernel PCA algorithm for dimensionality reduction instead of a standard PCA algorithm which will be irrelevant for this scenario. Several additional tuning parameters like the “Dropout” function could also be integrated into the CNN model to prevent overfitting. For larger datasets in product customization, researchers could use techniques such as Grid search to effectively deal with more extensive data sizes, for instance, when having multiple product parameters to personalize.

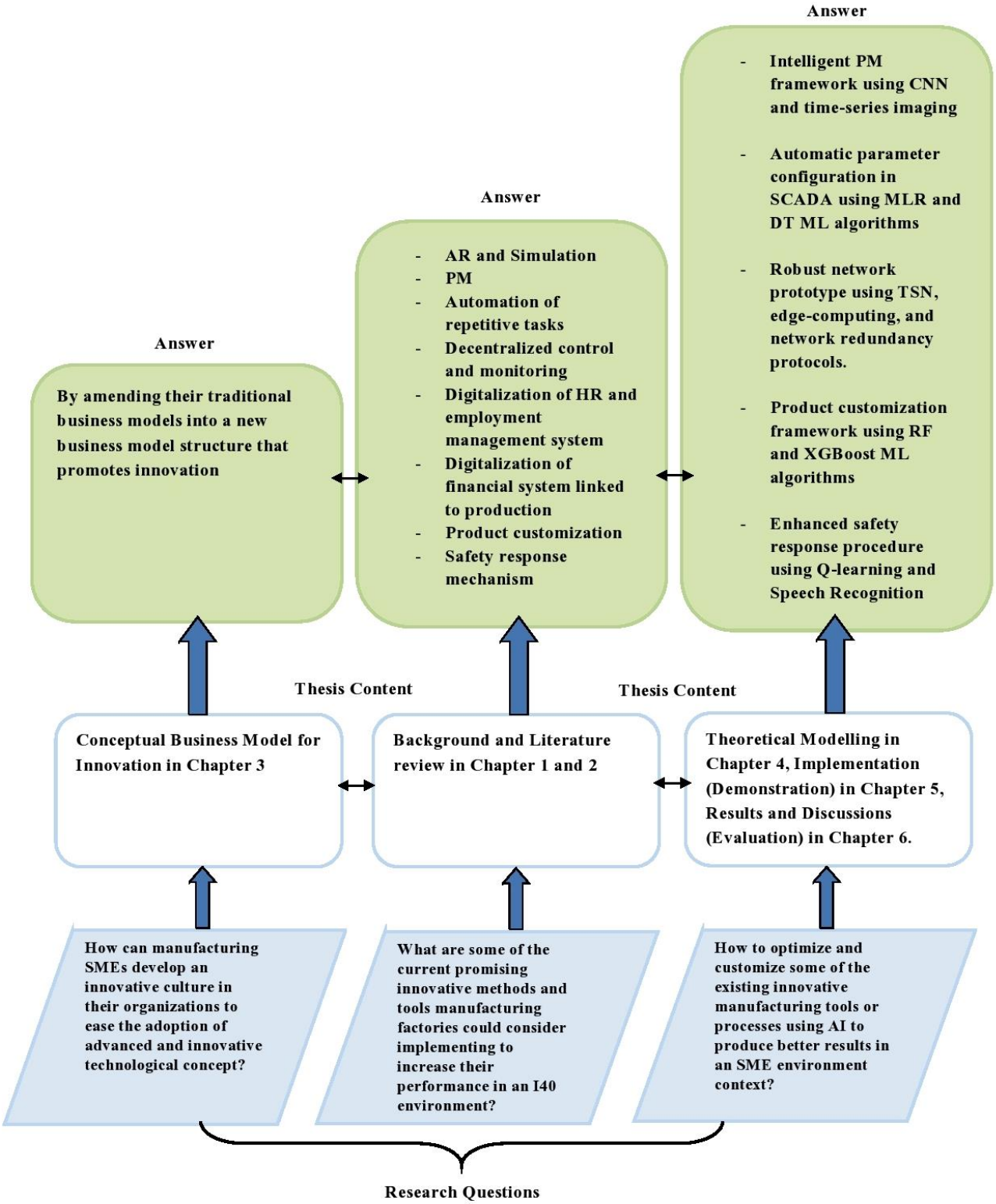

Figure 10-1: Research questions, objectives and answers

Our research did not tackle the interaction of our new proposed innovative solutions with existing factory components. Future studies should be conducted to find efficient ways to utilize legacy devices, software, and components installed in manufacturing factories with new

innovative solutions suggested in this research and progressively phase out the most outdated ones.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y. & Zheng, X. (2019) Tensorflow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pp. 265–283.

Abor, J. & Quartery, P. (2010) Issues in SME development in Ghana & South Africa. *International Research Journal of Finance and Economics*, 39, pp. 218-228.

Adhikari, R. & Agrawal, R.K. (2013) An introductory study on time series modeling and forecasting. *arXiv: 1302.6613*. Available at: https://arxiv.org/ftp/arxiv/papers/1302/1302.6613.pdf (Accessed: 03 April 2020)

Afaqui, M.S., Garcia-Villegas, E., & Lopez-Aguilera, E. (2017) IEEE 802.11ax: Challenges and requirements for future high efficiency Wi-Fi. *IEEE Wireless Communications*, 24(3), pp. 130–137.

Ali, A., Feng, L., Bashir, A., El-Sappagh, S., Ahmed, S., Iqbal, M. & Raja, G. (2018) Quality of Service Provisioning for Heterogeneous Services in Cognitive Radio-Enabled Internet of Things. *IEEE Transactions on Network Science and Engineering*, 7(1), pp. 328 – 342.

Alsalaet, J. (2012) Vibration analysis and diagnostic guide. Available at: https://www.researchgate.net/publication/311420765_Vibration_Analysis_and_Diagnostic_Guide (Accessed: 03 February 2020)

Antonelli, D., Pasquino, N. & Villa, A. (2007) *Mass-Customized Production in a SME Network.* In: Olhager J., Persson F. (eds) Advances in Production Management Systems. IFIP The International Federation for Information Processing, 246, Springer, Boston, MA.

Araujo, J.Á., Lázaro, J., Astarloa, A., Zuloaga, A. & Gárate, J.I. (2015) PRP and HSR for High Availability Networks in Power Utility Automation: A Method for Redundant Frames Discarding," in IEEE Transactions on Smart Grid, 6 (5) pp. 2325-2332,

ATCC Finance (2015) *Industry 4.0 challenges for the digital transformation and use of exponential technologies*. Deloitte. Available at: https://www2.deloitte.com/content/dam/Deloitte/ch/Documents/manufacturing/ch-en-manufacturing-industry-4-0-24102014.pdf (Accessed: 10 June 2021)

Avram, G. (2004) Empirical study on knowledge based systems. *The Electronic Journal of Information Systems Evaluation*, 8 (1), pp 11-20.

Baden-Fuller, C. & Haefliger, S. (2013) Business Models and Technological Innovation. *Long Range Planning*, 46, pp. 419–426.

Barga, R., Fontama, V., Tok, W.H., Cabrera-Cordon, L. (2015) *Predictive analytics with Microsoft Azure machine learning*, Springer. https://doi.org/10.1007/978-1-4842-0445-0

Bass, B. (2021) *The traditional hierarchy business model*. Available at: https://smallbusiness.chron.com/traditional-hierarchy-business-model-20784.html (Accessed: 24 March 2022)

Benali, L., Notton, G., Fouilloy, A., Voyant, C., & Dizene, R., (2019) Solar radiation forecasting using artificial neural network and random forest methods: Application to normal beam, horizontal diffuse and global components. *Renew. Energy*, 132, pp. 871–884.
Binding, A., Dykeman, N. & Pang, S. (2019) Machine Learning Predictive Maintenance on Data in the Wild. *In Proceedings of the IEEE 5th World Forum on Internet of Things (WF-IoT)*, pp. 507–512.

Boer, H.E.E., Nielsen, K. & Brunoe, T.D. (2018) *Can the SME Successfully Adopt Mass Customization?* In: Hankammer S., Nielsen K., Piller F., Schuh G., Wang N. (eds) Customization 4.0. Springer Proceedings in Business and Economics. Springer, Cham.

Box, G.E.P., Jenkins, G.M., Reinsel, G.C. & Ljung, G.M. (2015) *Time Series Analysis: Forecasting and Control*. 5th ed. John Wiley & Sons, New York, NY, USA.

Breiman, L. (1996) Bagging predictors. *Machine learning*, 24(2), pp. 123-140.

Breiman, L. (2001) Random forests. *Mach Learn*, 45(1), pp. 5–32.

Breiman, L., Friedman, J.H., Olshen, R.A. & Stone, C.J. (1984) Classification and Regression Trees. Monterey, CA: Wadsworth and Brooks.

Bruckner, D., StÄ.nicÄ., M.P., Blair, R., Schriegel, S., Kehrer, S., Seewald, M. & Sauter, T. (2019) An introduction to opc ua tsn for industrial communication systems. *Proceedings of the IEEE*, 107, pp. 1121–1131.

Brunoe, T.D. & Nielsen, K. (2016) Complexity Management in Mass Customization SMEs. *Procedia CIRP*, 51, pp. 38-43.

Chatfield, K., Simonyan, K., Vedaldi, A. & Zisserman, A. (2014) Return of the devil in the details: Delving deep into convolutional nets. *arXiv: 1405.3531v4*. Available at: https://arxiv.org/pdf/1405.3531.pdf (Accessed: 25 February 2020)

Chen, B., Wan, J., Celesti, A., Li, D., Abbas, H. & Zhang, Q. (2018) Edge Computing in IoT-Based Manufacturing. *IEEE Communications Magazine*, 56 (9), pp. 103-109. https://doi.org/10.1109/MCOM.2018.1701231.

Chen, M., Mao, S. & Liu, Y. (2014) Big data: A survey. *Mobile Network and Applications*, 19(2), pp. 171–209.

Chen, M., Zhang, Y., Li, Y., Mao, S. & Leung, V. (2015) EMC: Emotion-aware mobile cloud computing in 5G. IEEE Network, 29(2), pp. 32–38.

Chen, T. & Guestrin, C. (2016) Xgboost: A scalable tree boosting system. *In Proc. 22Nd ACM SIGKDD Int. Conf. Knowledge Disc. Data Mining*, pp. 785–794.

Chen, Y.J., Chang, D.K. & Zhang, C. (2020) Autonomous Tracking Using a Swarm of UAVs: A Constrained Multi-Agent Reinforcement Learning Approach. *IEEE Transactions on Vehicular Technology*, 69 (11), pp. 13702-13717.

Coraddu, A., Oneto, L., Ghio, A., Savio, S., Anguita, D. & Figari, M. (2014) Machine learning approaches for improving condition-based maintenance of naval propulsion plants. *Inst. Mech. Eng. M J. Eng. Maritime Environ.*, 230(1), pp. 136–153.

Craciunas, S.S., Oliver, R.S., Chmelík, M., et al. (2016) Scheduling real-time communication in IEEE 802.1Qbv time sensitive networks. *in 24th Int. Conf. on Real-Time Net. Syst.*, pp. 183-192.

Creswell, J.W. (2013) *Qualitative inquiry and research design: choosing among five approaches*. Kindle ed. USA: SAGE Publications.

Dabous, S.A., Ibrahim, F., Feroz, S. & Alsyouf, I. (2021) Integration of failure mode, effects, and criticality analysis with multi-criteria decision-making in engineering applications: Part I – Manufacturing industry. *Engineering Failure Analysis*, 122, https://doi.org/10.1016/j.engfailanal.2021.105264.

Das, S., Dey, A., Pal, A., & Roy, N. (2015) Applications of Artificial Intelligence in Machine learning: Review and Prospect. *International Journal of Computer Applications*, 115 (9), pp. 31-41.

Davies, R. (2015) Industry 4.0—Digitalisation for Productivity and Growth. *European Parliamentary Research Service (EPRS)*. Available at: http://www.europarl.europa.eu/RegData/etudes/BRIE/2015/568337/EPRS_ BRI(2015)568337_EN.pdf (Accessed: 31 July 2021)

Davis, S.M. (1989) From "future perfect": Mass customizing. *Planning Review*, 17(2), pp. 16-21.

Deloitte Insights (2020) The fourth industrial revolution, at the intersection of readiness and responsibility. Available at: https://www2.deloitte.com/content/dam/Deloitte/de/Documents/human-capital/Deloitte_Review_26_Fourth_Industrial_Revolution.pdf (Accessed: 10 July 2021)

Devarakonda, A., Naumov, M., & Garland, M. (2017) AdaBatch: Adaptive batch sizes for training deep neural networks. *arXiv:1712.02029*. Available at: https://arxiv.org/pdf/1712.02029.pdf (Accessed: 11 March 2020)

Dey, A. (2016) Machine Learning Algorithms: A Review. *International Journal of Computer Science and Information Technologies (IJCSIT)*, 7 (3), pp. 1174-1179

Diamond, S. & Marfatia, A. (2013) *Predictive maintenance for dummies*. Available at: https://cdn2.hubspot.net/hub/215445/file-534377792-pdf/Predictive_Maintenence_forDummies-Operations.pdf. (Accessed: 5 June 2021)

Diaz, P., Salas,J.C., Cipriano, A., & Núñez, F., (2021) Random forest model predictive control for paste thickening. *Minerals Engineering*, 163. https://doi.org/10.1016/j.mineng.2020.106760.

Dong, Y., Du, B. & Zhang, L. (2015) Target Detection Based on Random Forest Metric Learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(4), pp. 1830-1838. https://doi.org/10.1109/JSTARS.2015.2416255

Dossou, P.E. (2019) Development of a new framework for implementing Industry 4.0 in companies. *Procedia Manufacturing*, 38, pp. 573-580.

Duan, Y., Edwards, J. & Dwivedi, Y., (2019). Artificial intelligence for decision making in the era of Big Data – evolution, challenges and research agenda. *International Journal of Information Management*, 48, pp. 63-71. Available at: https://eur-lex.europa.eu/eli/reco/2003/361/oj (Accessed: 31 July 2021)

El-Tantawy, S., Abdulhai, B. & Abdelgawad, H. (2013) Multiagent Reinforcement Learning for Integrated Network of Adaptive Traffic Signal Controllers (MARLIN-ATSC): Methodology and Large-Scale Application on Downtown Toronto. in *IEEE Transactions on Intelligent Transportation Systems*, 14 (3), pp. 1140-1150.

Elvesæter, B., Berre, A. J., De Man, H., & Li, M.S. (2010) Networked Enterprise transformation and resource management in future internet enabled Innovation Clouds. *Enterprise Interoperability IV*, pp. 313-322.

Falamarzi, A., Moridpour, S., Nazem, M. & Cheraghi, S. (2019) Prediction of tram track gauge deviation using artificial neural network and support vector regression. *Aust. J. Civ. Eng.*, 17, pp. 63–71.

Falkenreck, C., & Wagner, R. (2017) The Internet of Things – Chance and challenge in industrial business relationships. Industrial Marketing Management, 66, pp. 181-195.

Foss, N. J., & Saebi, T. (2017) Fifteen Years of Research on Business Model Innovation: How Far Have We Come, and Where Should We Go? *Journal of Management*, 43 (1), pp. 200-227.

Fu, S., Zhang, H. & Chen, J. (2018) Time Sensitive Networking Technology Overview and Performance Analysis. *ZTE communications*, 16 (4). https://doi.org/10.19729/j.cnki.1673-5188.2018.04.009. Available at: https://res-www.zte.com.cn/mediares/magazine/publication/com_en/article/201804/FUShousai.pdf (Accessed: 10 July 2021)

Ganta, S. (2020) *A Framework to Incorporate Industry 4.0 into SME to Enhance Resource Planning*. Master thesis, Marladalen University, Sweden. Available at: https://www.diva-portal.org/smash/get/diva2:1416560/FULLTEXT01.pdf [Accessed on: 12 June 2021]

Ganzarain, J. & Errasti, N. (2016) Three Stage Maturity Model in SME's Toward Industry 4.0. *Journal of Industrial Engineering and Management*, 9 (5), pp. 1119–1128.

Gao, Z., Cecati, C. & Ding, S.X. (2015) A Survey of Fault Diagnosis and Fault-Tolerant Techniques—Part II: Fault Diagnosis With Knowledge-Based and Hybrid/Active Approaches. *IEEE Transactions on Industrial Electronics*, 62 (6), pp. 3768-3774.

Garcia, M.A.R, Rojas, R., Gualtieri, L., Rauch, E. & Matt, D. (2019) A human-in-the-loop cyber-physical system for collaborative assembly in smart manufacturing. *Procedia CIRP*, 81, pp. 600-605.

Giordani, P. & Kiers, H. (2004) Principal Component Analysis of symmetric fuzzy data. *Computational Statistics & Data Analysis*, 45, pp. 519-548.

Gislason, P.O., Benediktsson,J.A., & Sveinsson, J.R. (2006) Random forests for land cover classification. *Pattern Recognit. Lett.*, 27 (4), pp. 294–300.

Glorot, X., Bordes, A., & Bengio, Y. (2011) Deep Sparse Rectifier Neural Networks. *Proc. Conf. Artificial Intelligence and Statistics*, pp. 315-323.

Glova, J., Sabol, T., & Vajda, V. (2014) Business models for the internet of things environment. *Procedia Economics and Finance*, 15, pp. 1122-1129.

Gokalp, M.O., Kayabay, K., Akyol, M.A., Eren, P.E., & Koçyiğit, A. (2016) Big Data for Industry 4.0: A Conceptual Framework. *Int. Conf. Comput. Sci. and Comput. Intel. (CSCI)*, pp. 431-434.

Golightly, D., Kefalidou, G., & Sharples, S. (2017) A cross-sector analysis of human and organisational factors in the deployment of data-driven predictive maintenance. *Information Systems and e-Business Management*, pp. 1-22. https://doi.org/10.1007/s10257-017-0343-1

Goodfellow, I.J., Warde-Farley, D., Mirza, M., Courville, A.C. & Bengio, Y. (2013) Maxout networks. *in Proc. Int. Conf. Mach. Learn.*, 28 (3), pp.1319–1327.

Gorecky, D., Schmitt, M., Loskyll, M. & Zühlke, D. (2014) Human-machine-interaction in the industry 4.0 era. *12th IEEE INDIN*, pp. 289-294.

Górecki, T. & Łuczak, M. (2015) Multivariate time series classification with parametric derivative dynamic time warping. *Expert Syst. Appl.*, 42, pp. 2305–2312.

Grall, A., Dieulle, L., Berenguer, C. & Roussignol, M. (2002) Continuoustime predictive-maintenance scheduling for a deteriorating system. *IEEE Transactions on Reliability*, 51 (2),pp. 141–150.

Gu, S., Holly, E., Lillicrap, T. & Levine, S. (2017) Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. *IEEE Int. Conf. on Robot. Autom. (ICRA)*, pp. 3389-3396.

Gudiksen, S., Poulsen, S. B. & Buur, J. (2014) Making business models. *International Journal of Co Creation in Design and the Arts*, 10 (1), pp.15-30.

Gupta, J.N.D & Sharma, S.K. (2003) *Globalization and Information Management Strategy. Encyclopedia of Information Systems: Changes in Competitive Environments.* Available at:

https://www.sciencedirect.com/topics/computer-science/traditional-business-model (Accessed: 20 July 2021).

Gutiérrez, C.S.V., Juan, L.U.S., Ugarte, I.Z. & Vilches, V.M. (2018) Time-sensitive networking for robotics. *arXiv:1804.07643*. Available at: https://arxiv.org/pdf/1804.07643.pdf (Accessed: 08 July 2021)

Haenlein, M., & Kaplan, A. (2019) A brief history of artificial intelligence: On the past, present, and future of artificial intelligence. *California Management Review*, 61(4), pp. 5-14.

Hansen, E. & Bøgh, S. (2020) Artificial intelligence and internet of things in small and medium-sized enterprises: A survey. *Journal of Manufacturing Systems*, 58, pp. 362 – 372.

Haque, M.H.(2008) Determination of Nema design induction motor parameters from manufacturing data. IEEE Transactions on Energy conversion, 23(4), pp. 997-1004.

Hashimoto, D.A., Rosman, G., Rus, D. & Meireles, O.R. (2018) Artificial intelligence in surgery: promises and perils. *Annals of surgery*, 268 (1), Article ID: 70, pp. 1-17.

Haykin, S. (2001) *Neural Networks A Comprehensive Foundation*. Second Edition, Tsinghua Publishing House, Beijing.

He, K., Zhang, X., Ren, S., & Sun, J. (2015a) Deep residual learning for image recognition. *arXiv:1512.03385*. Available at: https://arxiv.org/pdf/1512.03385.pdf (Accessed: 12 March 2020)

He, K., Zhang, X., Ren, S. & Sun, J. (2015b) Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. on Patt. Analyz. and Mach. Intel*, 37(9), pp. 1904-1916.

He, K., Zhang, X., Ren, S., & Sun, J. (2015c) Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *in Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1026–1034.

Hecklau, F., Galeitzke M., Flachs, S. & Kohl, H. (2016). Holistic Approach for Human Resource Management in Industry 4.0. *Procedia CIRP*, 54, pp. 1-6.

Hefner, J.T., Spradley, M.K., & Anderson, B., (2014) Ancestry assessment using random forest modeling. *J. Forensic Sci*. 59 (3), pp. 583–589.

Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M. & Silver, D. (2018) Rainbow: Combining improvements in deep reinforcement learning. *Thirty-Second AAAI Conference on Artificial Intelligence, arXiv:1710.02298*. Available at: https://arxiv.org/pdf/1710.02298.pdf (Accessed: 10 July 2021)

Hesser, D.F. & Markert, B. (2019) Tool wear monitoring of a retrofitted CNC milling machine using artificial neural networks. *Manuf. Lett.*, 19, pp. 1–4.

Hevner, A.R., Salvatore T.M., Park, J. & Ram, S. (2004) Design Science Research in Information Systems. *MIS quarterly*, 28(1), pp. 75–105.

Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R.R. (2012) Improving neural networks by preventing co-adaptation of feature detectors. arXiv: 1207.0580. Available at: https://arxiv.org/pdf/1207.0580.pdf (Accessed: 15 February 2020)

Hirschmann (2014) White paper: "Media Redundancy Concepts - High Availability in Industrial Ethernet. Available at: https://www.belden.com/hubfs/emea/resources/Picture%20Park%20Assets/Files%20for%20Redirection/WP_%20Media%20Redundancy%20Concepts_Original_64020.pdf (Accessed: 02 February 2021)

Hopgood, A. (2005) The State of Artificial Intelligence. *Advances in Computers*, 65, pp. 1-75. https://doi.org/10.1016/S0065-2458(05)65001-2.

Hotelling, H. (1933) Analysis of a complex of statistical variables into principal components. J. Educ. Psychol. 24 (417–441) pp. 498–520.

Howard, A.G. (2013) Some improvements on deep convolutional neural network based image classification. *arXiv:1312.5402*. Available at: https://arxiv.org/ftp/arxiv/papers/1312/1312.5402.pdf (Accessed: 12 March 2020).

Huynh, M., Goose, S., Mohapatra, P. & Liao, R. (2011) RRR: Rapid ring recovery submillisecond decentralized recovery for ethernet ring. *IEEE Transactions on Computers*, 60, pp. 1561–1570.

Ifm electronics (2013). *From process monitoring to vibration analysis, condition monitoring systems*. pp. 5-12. Available at: http://www.ifm.com/download/files/ifm-efector-octavis-brochure-GB-2013/$file/ifm-efector-octavis-brochure-GB-2013.pdf (Accessed: 04 February 2020)

Ingaldi, M. & Ulewicz, R. (2019) Problems with the Implementation of Industry 4.0 in Enterprises from the SME Sector. *Sustainability*, 12, 217. https://doi.org/10.3390/su12010217.

Iravani, S. & Duenyas, I. (2002) Integrated maintenance and production control of a deteriorating production system. *IIE Transactions*, 34(5), pp. 423- 435.

Islam, M.T., Taha, A.E.M & Akl, S. (2014) A survey of access management techniques in machine type communications. *IEEE Commun. Mag.*, 52 (4), pp. 74–81.

Ismail, H., Reid, I., Mooney, J., Poolton, J. & Arokiam, I. (2007) How Small and Medium Enterprises Effectively Participate in the Mass Customization Game. *IEEE Transactions on Engineering Management*, 54(1), pp. 86-97.

Jiang, H., He, Z., Ye, G. & Zhang, H. (2020) Network Intrusion Detection Based on PSO-XGBoost Model. *IEEE Access*, 8, pp. 58392-58401.

Jiang, J., Kamel, M. & Chen, L. (2004). Reinforcement learning and aggregation. *in Proc. IEEE Int. Conf. Syst., Man Cybern.*, pp. 1303–1308

Jin, X., Weiss, B.A., Siegel, D. & Lee, J. (2016) Present status and future growth of advanced maintenance technology and strategy in us manufacturing. *International journal of prognostics and health management*, 7(Spec Iss on Smart Manufacturing PHM). Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5207222/pdf/nihms835319.pdf (Accessed: 22 July 2021)

Jirdehi, M.A. & Rezaei, A. (2016) Parameters estimation of squirrel-cage induction motors using ann and anfis. *Alexandria Engineering Journal*, 55 (1), pp. 357-388.

Johnson, M. (2010) *Seizing the White Space: Business Model Innovation for Growth and Renewal*. 1. s.l.: Harvard Business Press.

Jolliffe, I. & Cadima, J. (2016) Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374, 20150202. https://doi.org/10.1098/rsta.2015.0202.

Jovanovski, B., Maierhofer, C., Boshnyaku, A., Fischer, C., Seykova, D., Chapotot, E., Sousa, F., Tschandl, M. & Ferreira, V. (2019) *Industry 4.0: A Comprehensive Approach -Main Features and Impacts on SMEs.* Chain project - Changing SME business by industry 4.0. Available at: https://chainproject.eu/files/2019/10/Industry-4.0-A-Comprehensive-Approach-Main-Features-and-Impacts-on-SMEs.pdf [Accessed on: 13.07.2021]

Kaelbling, L.P., Littman, M.L. & Moore, A.W. (1996) Reinforcement learning: A survey. *J. Artif. Intell. Res.*, 4 (1), pp. 237–285.

Kagermann, H., Wahlster,W. & Helbig, J. (2013) *Recommendations for Implementing the Strategic Initiative Industrie 4.0: Securing the Future of German Manufacturing Industry.* Final report of the Industrie 4.0 Working Group. Frankfurt: Acatech. Available at: https://en.acatech.de/publication/recommendations-for-implementing-the-strategic-initiative-industrie-4-0-final-report-of-the-industrie-4-0-working-group/ (Accessed: 31 July 2021)

Kamunge, M.S., Njeru, A. & Tirimba, O.I. (2014) Factors affecting the performance of small and macro enterprises in Limuru town market of Kiambu County. *International Journal of Scientific and Research Publications*, 4(12), pp. 1-20.

Kaplan, S. & Winby, S. (2007) Organizational Models for Innovation: Organizational Designs that Support Strategic Innovation & Growth. Available at: http://www.vps.ns.ac.rs/Materijal/mat936.pdf (Accessed: 25 May 2021)

Kharb, S. & Singhrova, A. (2019) Fuzzy based priority aware scheduling technique for dense industrial iot networks. *Journal of Network and Computer Applications*, 125, pp. 17–27.

Khoshnevisan, M., Joseph, V., Gupta, P., Meshkati, F., Prakash, R. & Tinnakornsrisuphap, P. (2019) 5G Industrial Networks With CoMP for URLLC and Time Sensitive Network Architecture. *IEEE Journal on Selected Areas in Communications*, 37 (4), pp. 947-959.

Kiangala, K.S., & Wang, Z. (2018) Initiating predictive maintenance for a conveyor motor in a bottling plant using industry 4.0 concepts. Int J Adv Manuf Technol, 97, pp. 3251–3271.

Kiangala, K.S. & Wang, Z. (2020 a) An Effective Predictive Maintenance Framework for Conveyor Motors Using Dual Time-Series Imaging and Convolutional Neural Network in an Industry 4.0 Environment. *IEEE Access*, 8, pp. 121033-121049.

Kiangala, K.S. & Wang, Z. (2020 b) An adaptive framework for configuration of parameters in an Industry 4.0 manufacturing SCADA system by merging machine learning techniques. *Int. Conf. Artif. Intel., Big Data, Comput. Data Comm. Syst. (icABCD)*, https://doi.org/10.1109/icABCD49160.2020.9183818.

Kiangala, K.S. & Wang, Z. (2021) An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment. *Machine Learning with Applications*, 4. https://doi.org/10.1016/j.mlwa.2021.100024.

Kobzan, T., Schriegel, S., Althoff, S., Boschmann, A., Otto, J. & Jasperneite, J. (2018) Secure and time-sensitive communication for remote process control and monitoring. *IEEE 23rd Int. Conf. on Emerg. Technol. and Fact. Autom. (ETFA)*, pp. 1105–1108. doi:10.1109/ETFA.2018.8502539.

Kolasiński, T.W. (2012) Devious competitiveness paths of SMEs in Sub-Sahara Africa: Selected issues. *Management &Business Administration*, Central Europe, 5(118), pp. 83-99.

Kothari, C.R. (2004) *Research methodology, methods and techniques*. 2nd ed. New Delhi: International Publishers.

Kotsiantis, S.B. (2007) Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, 31, pp. 249-268.

Krizhevsky, A., Sutskever, I. & Hinton, G.E. (2012) Imagenet classification with deep convolutional neural networks. *Adv. in Neur. Inf. Proces. Syst. (Curran Associates, Inc)*, 25, pp. 1097–1105.

Krupitzer, C., Wagenhals, T., Züfle, M., Lesch, V., Schäfer, D., Mozaffarin, A., Edinger, J., Becker, C. & Kounev, S. (2020) A Survey on Predictive Maintenance for Industry 4.0. arXiv:2002.08224. Available at: https://arxiv.org/pdf/2002.08224.pdf (Accessed: 21 July 2021)

Kulkarni, K., Devi, U., Singhee, A., Hazra, J. and Rao, P. (2018) Predictive Maintenance for Supermarket Refrigeration Systems Using Only Case Temperature Data. *Ann. Amer. Contr. Conf. (ACC)*, pp. 4640–4645.

Leahy, K., Hu R.L., Konstantakopoulos, I.C., Spanos, C.J., & Agogino, A.M. (2016) Diagnosing wind turbine faults using machine learning techniques applied to operational data. *in Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM*), doi: 10.1109/ICPHM.2016.7542860.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W. & Jackel, L.D. (1989) Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1 (4), pp. 541-551, 1989.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., & Jackel, L.D. (1990) Handwritten Digit Recognition with a Back-Propagation Network. *Proc. Advances in Neural Information Processing Systems Conf.*, pp. 396-404.

Lee, J., Kao, H.A. & Yang, S. (2014) Service innovation and smart analytics for industry 4.0 and big data environment. *Procedia CIRP*, 16, pp. 3 - 8.

Leong, C.Y. & Koshijima, I. (2018). Internet of Things (IoT) for Dynamic Change Management in Mass Customization. Available at: http://ceur-ws.org/Vol-2063/lidari-paper1.pdf (Accessed: 03 July 2021)

Leurent, E. & Mercat, J. (2019) Social attention for autonomous decision-making in dense traffic. *In Machine Learning for Autonomous Driving Workshop at NeurIPS*. Available at: https://arxiv.org/pdf/1911.12250.pdf (Accessed: 11 March 2021)

Li, J., Deng, L., Haeb-Umbach, R. & Gong, Y. (2016) *Chapter 2 - Fundamentals of speech recognition*. in Robust Automatic Speech Recognition, J. Li, L. Deng, R. Haeb-Umbach and Y. Gong, Eds: Academic Press, pp. 9-40.

Li, S., Xu, L.D. & Zhao, S. (2018) 5G internet of things: A survey. *Journal of Industrial Information Integration*, 10, pp. 1-9. https://doi.org/10.1016/j.jii.2018.01.005.

Li, X., Li, D., Wan, J., Vasilakos, A., Lai, C. & Wang, S. (2015) A review of industrial wireless networks in the context of industry 4.0. *Wireless Network*, pp. 1–19, https://doi.org/10.1007/s11276-015-1133-7.2015.

Liao, H., Zhou, Z., Zhao, X., Zhang, L., Mumtaz, S., Jolfaei, A., Ahmed, S. & Bashir, A. (2020) Learning-Based Context-Aware Resource Allocation for Edge Computing-Empowered Industrial IoT. *IEEE Internet of Things Journal*, 7 (5), pp. 4260-4277.

Liaw, A. & Wiener, M., (2002) Classification and regression by Random forest. *R News*, 2 (3), pp. 18–22.

Liberto, D. (2020) *Small and Mid-size Enterprise (SME)*, Investopedia. Available at: https://www.investopedia.com/terms/s/smallandmidsizeenterprises.asp (Accessed: 22 July 2021)

Lichtblau, K., Stich, V., Bertenrath, R., Blum, M., Bleider, M., Millack, A., Schmitt, K., Schmitz, E. & Schröter, M. (2015) IMPULS-industrie 4.0-readiness. *Foundation for mechanical engineering, plant engineering, and information technology*. Available at: https://industrie40.vdma.org/documents/4214230/26342484/Industrie_40_Readiness_Study_1529498007918.pdf/0b5fd521-9ee2-2de0-f377-93bdd01ed1c8 (Accessed: 01 August 2021).

Lin, C.C., Deng, D.J., Chih, Y.L. & Chiu, H.T. (2019) Smart manufacturing scheduling with edge computing using multiclass deep Q network. *IEEE Transactions on Industrial Informatics*, 15, pp. 4276–4284.

Lin, J., Yu,W., Zhang, N., Yang, X., Zhang, H. & Zhao,W. (2017) A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet of Things Journal*, 4, pp. 1125–1142. https://doi.org/10.1109/JIOT.2017.2683200.

Lin, M., Chen, Q. & Yan, S. (2013) Network in network. *arXiv:1312.4400*. Available at: https://arxiv.org/pdf/1312.4400.pdf (Accessed: 20 February 2020)

Lo Bello, L. & Steiner, W. (2019) A perspective on IEEE time-sensitive networking for industrial communication and automation systems. *Proceedings of the IEEE*, 107, pp. 1094–1120.

Lohan, E.S., Koivisto, M., Galinina, O., Andreev, S., Tölli, A., Destino, G. et al. (2018) Benefits of Positioning-Aided Communication Technology in High-Frequency Industrial IoT. *IEEE Communication Magazine*, 56 (12), pp. 142–148.

Maas, A.L., Hannun, A.Y., & Ng A.Y. (2013) Rectifier nonlinearities improve neural network acoustic models. *in Proc. Int. Conf. Mach. Learn.(ICML)*, pp.1-6.

Madakam, S., Ramaswamy, R & Tripathi, S. (2015) Internet of Things (IoT): A Literature Review. *Journal of Computer and Communications*, 3, pp. 164-173.

Malica T. (2012) Collective Intelligence vs. Artificial Intelligence- Review. *International Journal of Emerging Technology and Advanced Engineering*, 2(7), pp. 198–199.

Mao, H., Alizadeh, M., Menache, I., & Kandula, S. (2016) Resource management with deep reinforcement learning. *In Proc. of ACM Workshop on Hot Topics in Networks*, pp. 50-56.

Marki, B., Bijakši, S. & Šanti, M. (2016) Artificial Intelligence in Determination of Marketing Customer. *Media, culture and public relations*, 7, pp. 5–13.

Matt, D.T. (2007) Reducing the Structural Complexity of Growing Organizational Systems by Means of Axiomatic Designed Networks of Core Competence Cells. *Journal of Manufacturing Systems*, 26, pp. 178–187.

Matt, D.T., Modrák, V. & Zsifkovits, H. (2020) *Industry 4.0 for SMEs Challenges, Opportunities and Requirements*. Open Access eBook, Palgrave Macmillan. https://doi.org/10.1007/978-3-030-25425-4

Matt, D.T., & Rauch, E. (2020) *SME 4.0: The role of small-and medium-sized enterprises in the digital transformation.* In Industry 4.0 for SMEs challenges, opportunities and requirements, ed. D.T. Matt, V. Modrak, and H. Zsifkovits, pp. 3–36. Basingstoke: Palgrave Macmillan. https://doi.org/10.1007/978-3-030-25425-4_1.

Matt, D.T., Rauch, E. & Fraccaroli, D. (2016) Smart Factory fur den Mittelstand. *ZWF Zeitschrift Für Wirtschaftlichen Fabrikbetrieb*, 111 (1–2), pp. 52–55.

Matt, D.T., Rauch, E., & Riedl, M. (2018) *Knowledge Transfer and Introduction of Industry 4.0 in SMEs: A Five-Step Methodology to Introduce Industry 4.0*. In Analyzing the Impacts of Industry 4.0 in Modern Business Environments, ed. R. Brunet-Thornton and F. Martinez, pp. 256–282. Hershey, PA: IGI Global.

Meyer Zu Wickern, V. (2019) *Challenges and Reliability of Predictive Maintenance*. Research Work, Rhein-waal university of applied sciences, Germany. https://doi.org/10.13140/RG.2.2.35379.89129

Min, H. (2010) Artificial intelligence in supply chain management: theory and applications. *International Journal of Logistics: Research and Applications*, 13(1), pp. 13-39

Mittal, S., Khan, M.A., Romero, D. & Wuest, T. (2018) A critical review of smart manufacturing & industry 4.0 maturity models: Implications for small and mediumsized enterprises (SMES). *Journal of Manufacturing Systems*, 49, pp. 194 – 214.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, AA., Veness, J., Bellemare, M.G., Graves, A., Rie dmiller, M., Fidjeland, A.K., Ostrovski, G., et al. (2015) Human-level control through deep reinforcement learning. *Nature*, 518, pp. 529–533.

Mo, H., Sun, H., Liu, J., & Wei, S. (2019) Developing window behavior models for residential buildings using XGBoost algorithm. *Energy and Buildings*, 205, Article ID: 109564. https://doi.org/10.1016/j.enbuild.2019.109564.

Modrak, V., Bednar, S. & Marton, D. (2015) Generating product variations in terms of mass customization. *IEEE 13th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pp. 187-192.

Mullainathan, S., & Spiess, J. (2017). Machine Learning: An Applied Econometric Approach. *Journal of Economic Perspectives*, 31(2), pp. 87-106. doi:10.1257/jep.31.2.87

Munir, M. (2015) *Eigenvalues-Theory and Applications*. Depart. of Math., Govern. Postgrad. College, Karl-Franzens-Universität, Graz, Austria. Available at: https://www.researchgate.net/publication/309012418_Eigenvalues-Theory_and_Applications (Accessed: 06 February 2020)

Muriithi, S. (2017) African small and medium enterprises (SMES) contributions, challenges and solutions. *International Journal of Research & Reflection in Management Science*, 5 (1), pp.36 – 48.

Musaddiq, A., Zikria, Y.B., Hahm, O., Yu, H., Bashir, A.K., & Kim, S.W. (2018) A survey on resource management in IoT operating systems. *IEEE Access*, 6, pp. 8459–8482.

Mwarari, M.M. & Ngugi, P.K. (2013) Factors influencing listing of Kenyan SMEs in the securities market for capital raising opportunities. *European Journal of Management Sciences and Economics*, 1(2), pp. 99-115.

Nair, V. & Hinton, G.E (2010) Rectified linear units improve restricted Boltzmann machines. *in Proc. 27th Int. Conf. Mach. Learn. (ICML)*, pp. 807–814.

Najafabadi, M., Villanustre, F., Khoshgoftaar, T., Seliya, N., Wald, R., & Muharemagic, E. (2016) Deep Learning Techniques in Big Data Analytics. *Journal of Big Data*, 2, pp. 1-21. https://doi.org/10.1007/978-3-319-44550-2_5.

Nayak, A. & Dutta, K. (2017) Impacts of machine learning and artificial intelligence on mankind. *Int. Conf. Intel. Comput. Cont. (I2C2)*. https://doi.org/10.1109/I2C2.2017.8321908.

Nguyen, K.A., Do, P. & Grall, A. (2015) Multi-level predictive maintenance for multi-component systems. *Reliability Engineering and System Safety*, 144, pp. 83–94.

Nguyen, H. & La, H. (2019) Review of deep reinforcement learning for robot manipulation. *in Proc. 3rd IEEE Int. Conf. Robotic Comput.*, pp. 590–595.

Opitz, D. & Maclin, R. (1999) Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, 11, pp. 169- 198.

Orzes G., Poklemba R. & Towner W.T. (2020) *Implementing Industry 4.0 in SMEs: A Focus Group Study on Organizational Requirements.* In: Matt D., Modrák V., Zsifkovits H. (eds) Industry 4.0 for SMEs. Palgrave Macmillan, Cham. https://doi.org/10.1007/978-3-030-25425-4_9

Osterwalder, A. & Pigneur, Y., (2010). *Business Model Generation*. s.l.: John Wiley & Sons, Inc.

Ou, X., Chang, Q, Arinez, J. & Zou, J. (2018) Gantry Work Cell Scheduling through Reinforcement Learning with Knowledge-guided Reward Setting. *IEEE Access*, 6, pp. 14699-14709.

Oviatt, S. & Cohen, P. (2000) Perceptual user interfaces: multimodal interfaces that process what comes naturally. *Commun. ACM*, 43(3), pp. 45-53.

Paolanti, M., Romeo, L., Felicetti, A., Mancini, A., Frontoni, E. & Loncarski, J. (2018) Machine learning approach for predictive maintenance in Industry 4.0. *14th IEEE/ASME Int. Conf. on Mechatron. and Emb. Syst. and App. (MESA)*, https://doi.org/10.1109/MESA.2018.8449150.

Parida, V., Westerberg, M., & Frishammar, J. (2012) Inbound open innovation activities in high-tech SMEs: the impact on innovation performance. *Journal of small business management*, 50(2), pp. 283-309.

Pasluosta, C.F., Gassner, H., Winkler, J., Klucken, J. & Eskofier, B.M. (2015) An emerging era in the management of parkinson's disease: Wearable technologies and the internet of things. *IEEE Journal of Biomedical and Health Informatics*, 19, pp. 1873–1881.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A. (2017) *Automatic differentiation in Pytorch.* Available at: https://openreview.net/pdf?id=BJJsrmfCZ (Accessed: 03 June 2021)

Patterson, J. & Gibson, A. (2017) *Deep Learning: A Practitioner's Approach*. Sebastopol, CA, USA: O'Reilly Media.

Pearson, K. (1901) On lines and planes of closest fit to systems of points in space. *Phil. Mag.* 2, pp. 559–572.

Peffers, K., Tuunanen T., Rothenberger M.A., Chatterjee S., (2007) A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), pp. 45-78

Perera, C., Liu, C.H., Jayawardena, S. & Chen, M. (2014) A survey on internet of things from industrial market perspective. *IEEE Access*, 2, pp. 1660–1679.

Petrillo, A., De Felice,F., Cioffi, R. & Zomparelli, F. (2018) Fourth Industrial Revolution: Current Practices, Challenges, and Opportunities. In Digital Transformation in Smart Manufacturing (Intechopen). Available at: https://www.intechopen.com/chapters/58010 (Accessed: 18 June 2021)

Pham, K., Kim, D., Park, S., & Choi, H. (2021) Ensemble learning-based classification models for slope stability analysis. *Catena*, 196. https://doi.org/10.1016/j.catena.2020.104886

Pine, B.J. (1993) *Mass Customization: The New Frontier in Business Competition*. Boston, MA: Harvard Bus. Sch. Press

Prause, M. (2019) Challenges of Industry 4.0 Technology Adoption for SMEs: The Case of Japan. *Sustainability*, 11, Article ID 5807, https://doi.org/10.3390/su11205807

Prinz, F., Schoeffler, M., Lechler, A. & Verl, A., (2018) End-to-end redundancy between real-time I4.0 components based on time-sensitive networking. *IEEE 23rd Int. Conf. on Emerg. Technol. and Fact. Autom. (ETFA)*, pp. 1083–1086.

Qi, Q., & Tao, F. (2018). Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. *IEEE Access*, 6, pp. 3585-3593.

Qi, Q. & Tao, F. (2019) A Smart Manufacturing Service System Based on Edge Computing, Fog Computing, and Cloud Computing. *IEEE Access*, 7, pp. 86769-86777.

Rácz A, Bajusz D & Héberger K. (2021) Effect of Dataset Size and Train/Test Split Ratios in QSAR/QSPR Multiclass Classification. *Molecules*, 26(4):1111. https://doi.org/10.3390/molecules26041111

Ramzai, J. (2019) *Simple guide for ensemble learning methods*. Available at: https://towardsdatascience.com/simple-guide-for-ensemble-learning-methods-d87cc68705a2 (Accessed: 02 July 2020)

Rauch, E., Dallasega, P., & Unterhofer, M. (2019). Requirements and Barriers for Introducing Smart Manufacturing in Small and Medium-Sized Enterprises. *IEEE Engineering Management Review*, 47(3), pp. 87-94.

Rauch, E., Seidenstricker, S., Dallasega, P. & Hammerl, R. (2016) Collaborative Cloud Manufacturing: Design of Business Model Innovations Enabled by Cyberphysical Systems in Distributed Manufacturing Systems. *Journal of Engineering*, Article ID 1308639. http://dx.doi.org/10.1155/2016/1308639.

Rauch, E., Unterhofer, M., Rojas, R., Gualtieri, L., Woschank, M. & Matt, D. (2020) A Maturity Level-Based Assessment Tool to Enhance the Implementation of Industry 4.0 in Small and Medium-Sized Enterprises. *Sustainability*, 12 (9), pp. 1-18. https://doi.org/10.3390/su12093559

Rauch, E., Vickery, A.R., Brown, C.A. & Matt, D.T. (2020) SME Requirements and Guidelines for the Design of Smart and Highly Adaptable Manufacturing Systems. In *Industry 4.0 for SMEs Challenges, Opportunities and Requirements: Challenges, Opportunities and*

Requirements, ed. D.T. Matt, V. Modrak, and H. Zsifkovits, pp. 39 -71.

Redmon, J. & Farhadi, A. (2018) Yolov3: An incremental improvement. arXiv:1804.02767. Available at: https://arxiv.org/pdf/1804.02767.pdf (Accessed: 08 July 2021)

Reinstein, I., (2017) *XGBoost a Top Machine Learning Method on Kaggle, Explained*. Available at: http://www.kdnuggets.com/2017/10/xgboost-top-machine-learning-method-kaggle-explained.html (Accessed: 23 January 2021).

Rezaeibagha, F., Mu, Y., Huang, X., Yang, W. & Huang, K. (2019). Fully secure lightweight certificateless signature scheme for IIoT. *IEEE Access*, 7, pp. 144433–144443.

Ribeiro, T., Gonçalves, F., Garcia, I., Lopes, G. & Ribeiro, A.F. (2019) Q-Learning for Autonomous Mobile Robot Obstacle Avoidance. *IEEE Intern. Conf. Autonom. Robot Syst. Compet. (ICARSC)*, https://doi.org/10.1109/ICARSC.2019.8733621

Romeo, L., Paolanti, M., Bocchini, G., Loncarski, J., & Frontoni, E. (2018) An innovative Design Support System for Industry 4.0 Based on Machine Learning Approaches. *5th International Symposium on Environment-Friendly Energies and Applications (EFEA)*. https://doi.org/10.1109/EFEA.2018.8617089.

Rönnberg, H. & Areback, J. (2020) *Initiating transformation towards AI in SMEs*, Master thesis, Luleå University of Technology, Sweden. Available at: https://www.diva-portal.org/smash/get/diva2:1438217/FULLTEXT01.pdf (Accessed: 5 June 2021)

Sadouk, L. (2018) CNN Approaches for Time Series Classification. *Intechopen*. https://doi.org/10.5772/intechopen.81170. Available at: https://www.intechopen.com/books/time-series-analysis-data-methods-and-applications/cnn-approaches-for-time-series-classification (Accessed: 10 April 2020)

Safar, L., Sopko, J., Bednar, S. & Poklemba, R. (2018) Concept of SME Business Model for Industry 4.0 Environment. *TEM Journal*, 7, pp. 626-637.

Safar, L., Sopko, J., Dancakova, D. & Woschank, M. (2020) Industry 4.0—Awareness in South India. *Sustainability*, 12(8), pp. 1-18. https://doi.org/10.3390/su12083207

Sanders, C. (2011) A Guide to Vibration Analysis and Associated Techniques in Condition Monitoring. *DAK Consulting - Chiltern House*, pp. 3-8. Available at: https://www.coursehero.com/file/95153471/Vibration-Analysis-Guide-v2-0pdf/ (Accessed: 22 August 2021)

Schmidhuber, J. (2015) Deep learning in neural networks: An overview. Neural Networks, 61, pp. 85 – 117.

Schriegel, S., Kobzan, T. & Jasperneite, J. (2018) Investigation on a distributed sdn control plane architecture for heterogeneous time sensitive networks. *14th IEEE International Workshop on Factory Communication Systems (WFCS)*, pp. 1–10. doi:10.1109/WFCS.2018.8402356.

Spath, D., Ganschar,O., Gerlach, S., Hammerle, T.K. & Schlund, S. (2013) *Production work of the future - Industry 4.0*, Stuttgart: Fraunhofer publishing house. Available at:

https://www2.iao.fraunhofer.de/images/iao-news/produktionsarbeit-der-zukunft.pdf (Accessed: 11 July 2021)

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2014) Overfeat: Integrated recognition, localization and detection using convolutional networks. *in Proc. Int. Conf. Learning Representations, Scottsdale*, pp. 1–16.

Simonyan, K. & Zisserman, A. (2015) Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556.* Available at: https://arxiv.org/pdf/1409.1556.pdf (Accessed: 10 March 2020)

Soltani, M. (2018) Joint optimization of opportunistic predictive maintenance and multi-location
spare part inventories for a deteriorating system considering imperfect actions. *arXiv preprint arXiv:1810.06315*. Available at: https://arxiv.org/ftp/arxiv/papers/1810/1810.06315.pdf (Accessed: 12 July 2021)

Spiegel, S., Mueller, F., Weismann, D., & Bird, J. (2018) Cost-sensitive learning for predictive maintenance. *arXiv preprint arXiv:1809.10979*. Available at: https://arxiv.org/pdf/1809.10979.pdf (Accessed: 10 June 2021)

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014) Dropout: A simple way to prevent neural networks from overfitting. *The Journ. of Mach. Learn. Resear.*, 15 (1), pp. 1929–1958.

Srivastava, R.K., Masci, J., Kazerounian, S., Gomez, F. & Schmidhuber, J. (2013) Compete to compute. *in Adv. in Neur. Inform. Proces. Syst. (NIPS)*, pp. 2310–2318.

Stähler, P. (2002) Business Models as a Unit of Analysis for Strategizing. *Proceedings of the International Workshop on Business Models*. Available at: http://www.hec.unil.ch/aosterwa/Documents/workshop/Draft_Staehler.pdf (Accessed: 20 June 2021)

Stalidis G., Karapistolis D. & Vafeiadis T. (2015) Marketing Decision Support Using Artificial Intelligence and Knowledge Modeling: Application to Tourist Destination Management. *Procedia – Social and Behavioral Sciences*, 175, pp. 106–113.

Stojanova, T., Suzic, N. & Tekic, A. (2012) Implementation of Mass Customization Tools in Small and Medium Enterprises. *International Journal of Industrial Engineering and Management*, 3(4), pp. 253-260.

Su, Y., Lu, X., Zhao, Y., Huang, L. & Du, X. (2019) Cooperative Communications With Relay Selection Based on Deep Reinforcement Learning in Wireless Sensor Networks. *IEEE Sensors Journal*, 19 (20), pp. 9561-9569. https://doi.org/10.1109/JSEN.2019.2925719.

Sun, Z., Ozay, M. & Okatani, T. (2016). Design of Kernels in Convolutional Neural Networks for Image Classification. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science(), vol 9911. Springer, Cham. https://doi.org/10.1007/978-3-319-46478-7_4

Suo, G., Song, L., Dou, Y. & Cui, Z. (2019) Multi-dimensional Short-Term Load Forecasting Based on XGBoost and Fireworks Algorithm. *18th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pp. 245-248.

Susto, G.A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015) Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Trans. Ind. Informat*., 11(3), pp. 812–820.

Sutton, R.S. (1992) Introduction: The Challenge of Reinforcement Learning. *Machine Learning*, 8, pp. 225-227.

Sutton, R.S. & Barto, A.G. (1998) *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press.

Szedlak, C., Poetters, P. & Leyendecker, B. (2020) Application of Artificial Intelligence In Small and Medium-Sized Enterprises. *Proceedings of the 5th NA Intern. Conf. Indust. Engin. Operat. Manag. Detroit*, pp. 1574 – 1582.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. & Rabinovich, A. (2014) Going deeper with convolutions. arXiv:1409.4842. Available at: https://arxiv.org/pdf/1409.4842.pdf (Accessed: 02 March 2020)

Tang, J., Song, J., Ou, J., Luo, J., Zhang, X. & Wong, K. (2020) Minimum Throughput Maximization for Multi-UAV Enabled WPCN: A Deep Reinforcement Learning Method. *IEEE Access*, vol. 8, pp. 9124-9132.

Tao, F., Qi, Q., Liu, A., & Kusiak, A. (2018) Data-driven smart manufacturing. *Journal of Manufacturing Systems*, 48, pp. 157–169.

Teece, D.J (2010) Business Models, Business Strategy and Innovation. *Long Range Plan*, 43, pp. 172–194

Tessarolo, A., De Martin, M., Diffen, D., Branz, M. & Bailoni, M. (2014) Practical assessment of homothetic dimensioning criteria for induction motors. *7th IET Intern. Conf. on Pow. Electr., Mach. and Drives*, doi: 10.1049/cp.2014.0512.

Tessarolo, A., De Martin, M., Giulivo, D., Diffen, D., Lipardi, G., Mazzuca, T., & Piva, L.(2014) A heuristic homothetic approach to the dimensioning of induction motors from specification data. *IEEE Annual Conference from Research to Industry: The need for a More Effective Transfer (AEIT)*, doi: 10.1109/AEIT.2014.7002046.

Teti, R. & Kumara, S.R.T (1997) Intelligent Computing Methods for Manufacturing Systems, *CIRP Ann-Manuf. Technol*., 46 (2), 1997, pp. 629- 652.

The World Bank (2021) *Small and medium enterprises (SMEs) finance: Improving SMEs' access to finance and finding innovative solutions to unlock sources of capital*. Available at: https://www.worldbank.org/en/topic/smefinance (Accessed: 24 July 2021)

Thoben, K.D., Wiesner, S. & Wuest, T. (2017) "Industrie 4.0" and smart manufacturing-a review of research issues and application examples. *International Journal of Automation Technology*, 11 (1), pp. 4–16. doi:https://doi.org/10.20965/ijat.2017.p0004.

Tian, S. & Hu, Y. (2019) The Role of OPC UA TSN in IT and OT Convergence. *Chinese Automation Congress (CAC)*, pp. 2272-2276.

Tieng, H., Chen, C.F., Cheng, F.T. & Yang, H.C. (2017) Automatic virtual metrology and target value adjustment for mass customization. *IEEE Robotics and Automation Letters*, 2 (2), pp. 546−553.

Timmers, P. (1998) Business models for electronic markets. *Electronic markets*, 8(2), pp. 3-8.

Truve, T., Wallin, W., & Ryfors, D. (2019) *Swedish manufacturing SMEs readiness for Industry 4.0: what factors influence an implementation of artificial intelligence and how ready are manufacturing smes in sweden?* Bachelor's Thesis, Jönköping University, Jönköping International Business School, Sweden. Available at: http://www.diva-portal.org/smash/get/diva2:1321698/FULLTEXT01.pdf (Accessed: 15 June 2021)

TTTech. (2015) *IEEE standard Ethernet for guaranteed real-time communication*. Available at: https://www.tttech.com/technologies/time-sensitive-networking-tsn/ (Accessed: 13 July 2021)

Tyralis, H., Papacharalampous, G., & Langousis, A., (2019) A brief review of random forests for water scientists and practitioners and their recent history in water resources. *Water (Basel)*, 11 (5), Article ID: 910. https:// doi.org/10.3390/w11050910.

Ussif, R. & Salifu, K. (2020) Contributions of small & medium enterprises to economic developments in sub-saharan Africa. *International Journal of Academic Accounting, Finance & Management Research (IJAAFMR)*, 4, pp. 63-78.

Vaishnavi, V., Kuechler, B. & Petter, S. (2017) Design science research in Information systems. *Association for Information Systems*. Available at: http://www.desrist.org/design-research-in-information-systems (Accessed: 8 July 2021).

Veit, F., Geyer-Klingeberg, J., Madrzak, J., Haug, M. & Thomson, J (2017). The Proactive Insights Engine: Process Mining meets Machine Learning and Artificial Intelligence. *15th Intern. Conf. Business Proc. Manag*. Available at: http://ceur-ws.org/Vol-1920/BPM_2017_paper_192.pdf (Accessed: 12 July 2021)

Velliyur, M. (2019) *The Fourth Industrial Revolution is here— what makes it different?* Available at: https://medium.com/@vmahesh/the-fourth-industrial-revolution-is-here-what-makes-it-different-8264720e88a2 (Accessed: 23 July 2021)

Venkatesh, J. (2007) An introduction to total productive maintenance (TPM). *The plant maintenance resource center*, pp. 3-20.

Vitturi, S., Zunino, C. & Sauter, T. (2019) Industrial Communication Systems and Their Future Challenges: Next-Generation Ethernet, IIoT, and 5G. *in Proceedings of the IEEE*, 107 (6), pp. 944-961.

Visa, S., Ramsay, B., Ralescu, A.L., & Van Der Knaap, E. (2011) Confusion Matrix-based Feature Selection. *in Proc. Midwest Artif. Intel. Cognit. Scienc. Conf.*, 710, pp. 120–127.

Wahyuni, D. (2012) The Research Design Maze: Understanding Paradigms, Cases, Methods and Methodologies', *Journal of applied management accounting research*, 10(1), pp. 69–80.

Wan, J., Li, D., Yan, H. & Zhang, P. (2010) Fuzzy feedback scheduling algorithm based on central processing unit utilization for a software-based computer numerical control system. *Proceedings of the institution of mechanical engineers Eng. Manuf.*, 224 (7), pp. 1133–1143.

Wan, J., Yan, H., Li, D., Zhou, K. & Zeng, L. (2013) Cyber-physical systems for optimal energy management scheme of autonomous electric vehicle. *Computer Journal*, 56(8), pp. 947–956.

Wan, J., Yi, M., Li, D., Zhang, C., Wang, S. & Zhou, K. (2016) Mobile services for customization manufacturing systems: An example of industry 4.0. *IEEE Access*, 4, pp. 8977–8986.

Wan, L., Zeiler, M., Zhang, S., Cun, Y.L. & Fergus, R. (2013) Regularization of neural networks using dropconnect. *in Proc. Int. Conf. Machine Learning*, pp. 1058–1066.

Wang, C., Deng, C. & Wang, S (2019) Imbalance-XGBoost: Leveraging Weighted and Focal Losses for Binary Label-Imbalanced Classification with XGBoost. *arXiv:1908.01672*. Available at: https://arxiv.org/pdf/1908.01672.pdf (Accessed: 24 January 2021).

Wang, J., Ma, Y., Zhang, L., Gao, R.X. & Wu, D. (2018) Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems*, 48, pp. 144 – 156, special Issue on Smart Manufacturing. https://doi.org/10.1016/j.jmsy.2018.01.003.

Wang, N. & Hsu, W. (2020) Energy Efficient Two-Tier Data Dissemination Based on Q-Learning for Wireless Sensor Networks. *IEEE Access*, 8, pp. 74129-74136.

Wang, R.Y. & Xu, L. (2012) Multi-Agent Dam Management Model Based on Improved Reinforcement Learning Technology. *Applied Mechanics and Materials*, 198–199, pp. 922–926.

Wang, S., Wan, J., Imran, M., Li, D. & Zhang, C. (2016) Cloud-based smart manufacturing for personalized candy packing application. *Journal of Supercomputing*, pp. 1–19. https://doi.org/10.1007/s11227-016-1879-4.

Wang, S., Wan, J., Zhang, D., Li, D. & Zhang, C. (2016) Towards smart factory for industry 4.0: A self-organized multi-agent system with big data based feedback and coordination. *Computer Network*, 101, pp. 158–168.

Wang, Y., Xia, S., Tang, Q., Wu, J. & Zhu, X. (2018). A Novel Consistent Random Forest Framework: Bernoulli Random Forests. *IEEE Transactions on Neural Networks and Learning Systems*, 29 (8), pp. 3510-3523.

Wang, Y.C. & Usher, J.M. (2005) Application of reinforcement learning for agent-based production scheduling. *in Eng. Appl. Artif. Intell.*, 18 (1), pp. 73–82.

Wang, Z. & Oates, T. (2015) Imaging time-series to improve classification and imputation. *arXiv:1506.00327*. Available at: https://arxiv.org/pdf/1506.00327.pdf (Accessed: 11 February 2020)

Wangler, L. & Botthof, A., (2019) E-Goverance: Digitalisierung und KI inder öffentlichen Verwaltung. *In: V. Wittpahl, Hrsg. Künstliche Intelligenz. technologie, Anwendung, Gesellschaft*. Heidelberg: Springerverlag, pp. 122- 141

Watkins, C.J. & Dayan, P. (1992) Technical note: Q-learning. *Mach. Learn.*, 8, (3–4), pp. 279–292.

Weiss, G. (1999) *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. 1st ed. Cambridge, MA, USA: MIT Press.

Weking, J., Stöcker, M., Kowalkiewicz, M., Böhm, M. & Krcmar, H. (2018) Archetypes for Industry 4.0 Business Model Innovations. *Twenty-fourth Americas Conference on Information Systems (AMCIS)*. Available at: https://aisel.aisnet.org/amcis2018/StrategicIT/Presentations/3/ (Accessed: 22 July 2021).

Wenking, M., Benninghaus, C., & Friedli, T. (2016) Umsetzungsbarrieren und-losungen von Industrie 4.0. *ZWF Zeitschrift Für Wirtschaftlichen Fabrikbetrieb*, 111 (12), pp. 847–850.

Wicaksono, H. (2011) Q learning behavior on autonomous navigation of physical robot. *8th Intern. Conf. Ubiq. Robots Amb. Intel. (URAI)*, pp. 50-54.

Willemse, J. (2010) *The forum SA: SME failure statistics*. Available at: http://www.theforumsa.co.za/forums/showthread.php?t=7808 (Accessed: 31 July 2021)

Williams, C. (2019) *How the fourth industrial revolution will reshape risk management.* ERM Insights. Available at: https://www.erminsightsbycarol.com/fourth-industrial-revolution-risk-management/ (Accessed: 23 July 2021)

World Trade Report, WTO (2016) *Levelling the Trading Field for SMEs.* Available at: https://www.wto.org/english/res_e/booksp_e/world_trade_report16_e.pdf. (Accessed: 31 July 2021)

Wright, P. & Bourne, D.A. (1988), *Manufacturing intelligence*, Boston, MA 317 Addison-Wesley.

Xu, R. & Luo, F. (2021) Risk prediction and early warning for air traffic controllers' unsafe acts using association rule mining and random forest. *Safety Science*, 135. https://doi.org/10.1016/j.ssci.2020.105125.

Yam, R.C., Tse, P.W., Li, L. & Tu, P. (2001) Intelligent predictive decision support system for condition-based maintenance. *International Journal of Advanced Manufacturing Technology*, 17 (5), pp. 383–391.

Yang, C.L., Chen, Z.X. & Yang, C.Y. (2019) Sensor Classification Using Convolutional Neural Network by Encoding Multivariate Time Series as Two-Dimensional Colored Image. *Sensors*, 20 (1), Article ID 168, pp. 1-15. https://doi.org/10.3390/s20010168.

Yang, F. & Gu, S. (2021) Industry 4.0, a revolution that requires technology and national strategies. *Complex Intell. Syst*., 7, pp. 1311–1325.
Yang, G., Xie, L., MÃntysalo, M., Zhou, X., Pang, Z., Xu, L.D., Kao-Walter, S., Chen, Q. & Zheng, L.R. (2014) A health-iot platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box. *IEEE Transactions on Industrial Informatics*, 10, pp. 2180–2191.

Yang, X.D., Wang, J.M. & Zhang, L.N. (2017) Application of XGBOOST in Ultra-Short Term Load Forecasting. *Electrical Drive Automation*, 39, pp. 21-25

Yao, X., Zhou, J., Zhang, J. & Boër, C.R. (2017) From Intelligent Manufacturing to Smart Manufacturing Industry 4.0 Driven by Next Generation Artificial Intelligence and Further On. *IEEE 5th Intern. Conf. on Enterp. Syst. (ES)*, pp. 311-318.

Yue, L., Yi,Z., Pan,J., Li,X., & Li, J., (2021) Identify M Subdwarfs from M-type Spectra using XGBoost. *Optik*, 225. https://doi.org/10.1016/j.ijleo.2020.165535.

Zeiler, M.D. & Fergus, R. (2014) Visualizing and understanding convolutional networks. *in Proc. Eur. Conf. Comput. Vis.*, pp. 818–833.

Zeiler, M.D., Ranzato, M., Monga, R., Mao, M.,Yang, K.,Le, Q.V., Nguyen, P., Senior, A., Vanhoucke, V., Dean, J. & Hinton, G.E. (2013) On rectified linear units for speech processing. *in Proc. IEEE Int. Conf. Acoust. Speech Signal Process*, pp. 3517–3521.

Zenisek, J., Holzinger, F. & Affenzeller, M (2019) Machine learning based concept drift detection for predictive maintenance. *Comput. Ind. Eng*. 2019, 137, Article ID 106031. https://doi.org/10.1016/j.cie.2019.106031.

Zhang, D. (2012) A taxonomy of agent technologies for ubiquitous computing environments. *Transactions on Internet Information Systems*, 6(2), pp. 547–565.

Zhang, Q., Lin, J., Sha, Q., He, B. & Li, G. (2020) Deep Interactive Reinforcement Learning for Path Following of Autonomous Underwater Vehicle. *IEEE Access*, 8, pp. 24258-24268.

Zhang, Y. (2017) iDoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization. *Future Generation Computer System*, 66, pp. 30–35.

Zhang, Y., Qiu, M., Tsai, C.W., Hassan, M.M. & Alamri, A. (2015) HealthCPS: Healthcare cyber-physical system assisted by cloud and big data. *IEEE Systems Journal*, 11(1), pp. 88-95.

Zhang, Y., Zhang, B. & Wu, Z. (2019) Multi-Model Modeling of CFB Boiler Bed Temperature System Based on Principal Component Analysis. *IEEE Access*, pp. 389-399.

Zhang, Z. & Xu, J.M. (2005) A dynamic route guidance arithmetic based on reinforcement learning. *Intern. Conf. Mach. Learn. Cybern.*, 6, pp. 3607-3611.

Zhao, M., Lu, H., Yang, S. & Guo, F. (2020) The Experience-Memory Q-Learning Algorithm for Robot Path Planning in Unknown Environment. *IEEE Access*, 8, pp. 47824-47844, 2020.

Zhao, T., Zheng, S. & Li, W. (2018) Research on Credit Risk Analysis Based on XGBOOST. *Software engineering*, 21, pp. 33-35

Zhao, Y., Li, Y., Zhang, X., Geng, G., Zhang, W. & Sun, Y. (2019) A survey of networking applications applying the software defined networking concept based on machine learning. *IEEE Access*, 7, pp. 95397–95417.

Zhou, Y.T. & Chellappa, R. (1988) Computation of optical flow using a neural network. *Proc. IEEE Int. Conf. Neural Netw.*, 2 pp. 71–78.

Zhou, Z., Li, X. & Zare, R.N, (2017) Optimizing chemical reactions with deep reinforcement learning. *ACS central science*, 3 (12), pp. 337–1344.

Zott, C., & Amit, R. (2010) Business model design: an activity system perspective. *Long range planning*, 43(2), pp. 216-226.

Zott, C., Amit, R., & Massa, L. (2011) The Business Model: Recent Developments and Future Research. *Journal of management*, 37(4), pp. 1019-1042.

APPENDICES

5.3.4.2 1. Programming codes

A. An intelligent PM framework using CNN and time-series imaging

#PCA time-series dimensionality reduction in RStudio IDE

```
# Importing the dataset
dataset = read.csv('parameters1.csv')

# Splitting the dataset into the Training set and Test set
install.packages ('caTools')
library (caTools)
set.seed(123)
split = sample.split(dataset$Fault, SplitRatio = 0.8)
training_set = subset (dataset, split == TRUE)
test_set = subset (dataset, split == FALSE)

# Feature Scaling
training_set [-12] = scale (training_set [-12])
test_set [-12] = scale (test_set [-12])

# Applying PCA
install.packages ('caret')
library (caret)
install.packages ('e1071')
library (e1071)
pca = preProcess(x = training_set[-12], method = 'pca', pcaComp = 2)
training_set = predict (pca, training_set)
training_set = training_set[c (2, 3, 1)]
test_set = predict (pca, test_set)
test_set = test_set[c (2, 3, 1)]
```

#SVM classification modelling using the above PCA data to compare results with CNN classification

```
# Fitting SVM to the Training set
install.packages ('e1071')
library (e1071)
classifier = svm(formula = Fault ~ .,
                 data = training_set,
                 type = 'C-classification',
                 kernel = 'linear')

# Predicting the Test set results
y_pred = predict (classifier, newdata = test_set[-3])

# Making the Confusion Matrix
cm = table (test_set[, 3], y_pred)
```

 # PCA time-series data conversion into images using GAF in Python IDE (sample code for case1, same applies for case2 and case 3)

```
import matplotlib.pyplot as plt
from mpl_toolkits.axes_grid1 import ImageGrid
from pyts.image import GramianAngularField
import pandas as pd


#Load Dataset
dataset = pd.read_csv("case1.csv")


# Transform the time series into Gramian Angular Fields
gasf = GramianAngularField(image_size=3, method='summation')
X_gasf = gasf.fit_transform (dataset)
gadf = GramianAngularField(image_size=3, method='difference')
X_gadf = gadf.fit_transform (dataset)

# Settings of the converted images
fig = plt.figure(figsize=(8, 4))
grid = ImageGrid(fig, 111,
                 nrows_ncols=(1, 2),
                 axes_pad=0.15,
                 share_all=True,
                 cbar_location="right",
                 cbar_mode="single",
                 cbar_size="7%",
                 cbar_pad=0.3,
                 )
#Generate Gramian images for each time series one by one starting by [0]
images = [X_gasf [0], X_gadf [0]]
titles = ['GASF', 'GADF']
for image, title, ax in zip(images, titles, grid):
    im = ax.imshow(image, cmap='rainbow', origin='lower')
```

```
    ax.set_title (title, fontdict={'fontsize': 12})
ax.cax.colorbar (im)
ax.cax.toggle_label (True)
#plt.suptitle('Gramian Angular Fields', y=0.98, fontsize=16)
plt.show ()
```

 # Convolutional Neural Network using ReLU in Python IDE

```
# Part 1 - Building the CNN

# Importing the Keras libraries and packages
import numpy as np
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
from sklearn.metrics import confusion_matrix
import sklearn.metrics as metrics
from keras.preprocessing.image import ImageDataGenerator


# Initializing the CNN
classifier = Sequential()

# Step 1 - Convolution
classifier.add(Convolution2D(32, 3, 3, input_shape = (64, 64, 3), activation = 'relu'))

# Step 2 - Pooling
classifier.add(MaxPooling2D(pool_size = (2, 2)))

# Step 3 - Flattening
classifier.add(Flatten())

# Step 4 - Full connection
classifier.add(Dense(output_dim = 128, activation = 'relu'))
classifier.add(Dense(output_dim = 3, activation = 'softmax'))

# Compiling the CNN
classifier.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics =
['accuracy'])

# Part 2 - Fitting the CNN to the images

train_datagen = ImageDataGenerator (rescale = 1./255,
                                    shear_range = 0.2,
                                    zoom_range = 0.2,
```

```
                                horizontal_flip = True)

test_datagen = ImageDataGenerator (rescale = 1./255)

#Change the directory path accordingly
training_set =
train_datagen.flow_from_directory('C:/Convolutional_Neural_Networks/newdataset/Training
',
                                        target_size = (64, 64),
                                        batch_size = 32,
                                        class_mode = 'categorical')

#Change the directory path accordingly
test_set =
test_datagen.flow_from_directory('C:/Convolutional_Neural_Networks/newdataset/Test',
                                    target_size = (64, 64),
                                    batch_size = 32,
                                    class_mode = 'categorical',
                                    shuffle = False)

classifier.fit_generator(training_set,
                    samples_per_epoch = 12000,
                    nb_epoch = 3,
                    validation_data = test_set,
                    nb_val_samples = 3000)


test_steps_per_epoch = np.math.ceil (test_set.samples / test_set.batch_size)

predictions = classifier.predict_generator (test_set, steps = test_steps_per_epoch)
# Get most likely class
predicted_classes = np.argmax (predictions, axis=1)

true_classes = test_set.classes
class_labels = list (test_set.class_indices.keys ())

report = metrics.classification_report(true_classes, predicted_classes,
target_names=class_labels)
print (report)
print (confusion_matrix(test_set.classes, predicted_classes))
```

 # Convolutional Neural Network using PReLU in Python IDE

```
# Part 1 - Building the CNN
```

```
# Importing the Keras libraries and packages
import numpy as np
from keras.models import Sequential
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
from keras.layers.advanced_activations import PReLU
from keras.initializers import Constant
from sklearn.metrics import confusion_matrix
import sklearn.metrics as metrics
from keras.preprocessing.image import ImageDataGenerator

# Initialising the CNN
classifier = Sequential()

# Step 1 - Convolution
classifier.add (Convolution2D (32, 3, 3, input_shape = (64, 64, 3), activation = 'linear'))
classifier.add (PReLU (alpha_initializer=Constant (value=0)))

# Step 2 - Pooling
classifier.add (MaxPooling2D (pool_size = (2, 2)))


# Step 3 - Flattening
classifier.add (Flatten ())

# Step 4 - Full connection
classifier.add (Dense(output_dim = 128, activation = 'linear'))
classifier.add (PReLU (alpha_initializer=Constant (value=0)))
classifier.add (Dense (output_dim = 3, activation = 'softmax'))

# Compiling the CNN
classifier.compile (optimizer = 'adam', loss = 'categorical_crossentropy', metrics =
['accuracy'])

# Part 2 - Fitting the CNN to the images


train_datagen = ImageDataGenerator (rescale = 1./255,
                                    shear_range = 0.2,
                                    zoom_range = 0.2,
                                    horizontal_flip = True)

test_datagen = ImageDataGenerator (rescale = 1./255)

#Change directory path accordingly
```

```
training_set =
train_datagen.flow_from_directory('C:/Convolutional_Neural_Networks/newdataset/Training
',
                                  target_size = (64, 64),
                                  batch_size = 32,
                                  class_mode = 'categorical')

#Change directory path accordingly
test_set =
test_datagen.flow_from_directory('C:/Convolutional_Neural_Networks/newdataset/Test',
                                 target_size = (64, 64),
                                 batch_size = 32,
                                 class_mode = 'categorical',
                                 shuffle = False)
classifier.fit_generator (training_set,
                 samples_per_epoch = 12000,
                 nb_epoch = 3,
                 validation_data = test_set,
                 nb_val_samples = 3000)


test_steps_per_epoch = np.math.ceil (test_set.samples / test_set.batch_size)

predictions = classifier.predict_generator(test_set, steps=test_steps_per_epoch)
# Get most likely class
predicted_classes = np.argmax (predictions, axis=1)

true_classes = test_set.classes
class_labels = list (test_set.class_indices.keys ())

report = metrics.classification_report(true_classes, predicted_classes,
target_names=class_labels)
print (report)
print (confusion_matrix(test_set.classes, predicted_classes))
```

B. Automatic parameter configuration using MLR and DT ML algorithms

#Multiple Linear Regression

```
# Importing the dataset
dataset = read.csv('Dataset1.csv')

# Splitting the dataset into the Training set and Test set

#install.packages('caTools')
library(caTools)
```

```
set.seed(123)
split = sample.split(dataset$Heating.pressure, SplitRatio = 0.8)
training_set = subset (dataset, split == TRUE)
test_set = subset (dataset, split == FALSE)

# Feature Scaling
# training_set = scale(training_set)
# test_set = scale(test_set)
# Fitting Multiple linear regression to the training set
regressor = lm(formula = Heating.pressure ~ .,
               data = training_set)

#Predicting the Test Set Result
y_pred = predict (regressor, newdata = test_set)
```

#Decision Tree Regression

```
#Importing dataset

dataset = read.csv('DataDT.csv')
dataset = dataset[1:2]


#Splitting the dataset into training set and test set
#install.packages('caTools')
#library(caTools)
#set.seed(123)
#split = sample.split(dataset$Salary, SplitRatio = 2/3)
#training_set = subset(dataset, split == TRUE)
#test_set = subset(dataset, split == FALSE)


#Fitting the Decision Tree Regression Model to the dataset
#install.packages('party')
library(party)
library(rpart)
regressor = rpart(formula = Number.of.Bumps ~ .,
                  data = dataset,
                  control = rpart.control(minsplit = 10))


#Predicting a new result with the Decision Tree Regression model

y_pred = predict (regressor, data.frame (Product.Weight = 550))
y_pred = predict (regressor, data.frame (Product.Weight))
```

C. Product customization framework using XGBoost and RF ML algorithms

The programming code and data utilized are available online in Code ocean repository. "An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment": https://codeocean.com/capsule/2685317/tree/v1

D. Enhanced safety response mechanism using Q-learning and Speech recognition

#Q-learning sample code for obstacle-free path selection (In Python IDE from Google colab)

```
# Import the library
import numpy as np


# Set Hyperparameters values for the Q-learning algorithm
alpha=1
gamma=0.7

# Mapping each location letter to a digit (Python does not understand the letters as locations).
location_values = {'A':0,'B':1,'C':2,'D':3,'E':4,'F':5,'G':6,'H':7,'I':8,'J':9,'K':10,'L':11,'M':12,
                   'N':13,'O':14,'P':15,'Q':16,'R':17,'S':18,'T':19,'U':20,'V':21,'W':22,'X':23}


# Define a variable to receive the dynamic location for the simulation - the variable is a character.

loc = ''
loc =(input('Please enter your location '))
print(loc)

# Selection of the closest 'Safe Target' based on the entered location

if loc =='A' or loc =='B' or loc =='C' or loc =='G' or loc =='H' or loc =='I':
  reward= np.array([
                  [0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [1,0,0,0,0,0,10,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0],
                  [0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                  [0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0],
```

```
                [0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,1,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
                ])
elif loc =='M' or loc =='N' or loc =='O' or loc =='S' or loc =='T' or loc =='U':
  reward= np.array([
                [0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0],
                [0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,1,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,10,1,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
                ])

elif loc =='P' or loc =='Q' or loc =='R' or loc =='V' or loc =='W' or loc =='X':
  reward= np.array([
                [0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
```

```
                [0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0],
                [0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,1,0,0],
                [0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,1,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,10,1],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
                ])
else:
  reward= np.array([
                [0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,1,10,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0],
                [0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,0,0],
                [0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0],
                [0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,0,0,1,0,0],
                [0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1],
                [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0],
```

```
                ])

# Reset the matrix B to zero assuming the autonomous robot does not know anything about the
environment

B= np.array(np.zeros([24,24]))


# Computing the Q-learning algorithm to the reward matrix

for x in range(1000):
    location_current=np.random.randint(0,24)
    next_actions= []
    for y in range(24):
        if reward[location_current, y]>0:
            next_actions.append(y)
    next_location=np.random.choice(next_actions)
    temp_difference= reward[location_current,next_location]+gamma*B[next_location, np.argmax(B[next_
location,])]-B[location_current,next_location]
    B[location_current,next_location]=B[location_current,next_location]+alpha*temp_difference

print(B.astype(int))

# State to Location require to display Route
map_location = {state: location for location, state in location_values.items()}

map_location


# Code for Displaying Route

def route(starting_location, ending_location):
    route = [starting_location]
    next_location = starting_location
    while (next_location != ending_location):
        starting_state = location_values[starting_location]
        next_state = np.argmax(B[starting_state,])
        next_location = map_location[next_state]
        route.append(next_location)
        starting_location = next_location
    return route

#Print the route based on the selected location

if loc =='A' or loc =='B' or loc =='C' or loc =='G' or loc =='H' or loc =='I':
  print(route(loc,'G'))

elif loc =='M' or loc =='N' or loc =='O' or loc =='S' or loc =='T' or loc =='U':
  print(route(loc,'S'))
```

```
elif loc =='P' or loc =='Q' or loc =='R' or loc =='V' or loc =='W' or loc =='X':
  print(route(loc,'W'))

else:
  print(route(loc,'E'))
```

#Speech recognition to send voice command to S7-1200 Siemens PLC in Python IDE

```
import snap7.client as c
from snap7.util import *
from snap7.types import *


import pyttsx3
import speech_recognition as sr


def ReadMemory(plc,byte,bit,datatype):
        result = plc.read_area(areas['MK'],0,byte,datatype)
        if datatype==S7WLBit:
                return get_bool(result,0,bit)
        elif datatype==S7WLByte or datatype==S7WLWord:
                return get_int(result,0)
        elif datatype==S7WLReal:
                return get_real(result,0)
        elif datatype==S7WLDWord:
                return get_dword(result,0)
        else:
                return None

def WriteMemory(plc,byte,bit,datatype,value):
        result = plc.read_area (areas['MK'],0,byte,datatype)
        if datatype==S7WLBit:
                set_bool (result,0,bit,value)
        elif datatype==S7WLByte or datatype==S7WLWord:
                set_int (result,0,value)
        elif datatype==S7WLReal:
                set_real (result,0,value)
        elif datatype==S7WLDWord:
                set_dword (result,0,value)
        plc.write_area (areas["MK"],0,byte,result)

if __name__=="__main__":
        plc = c.Client()
        plc.connect ('172.16.2.254',0,1)
        #WriteMemory (plc,0,1,S7WLBit,0)
        #a=ReadMemory (plc,50,0,S7WLWord)
        #print (a)
```

```
class Shutdown:

    def Speak(self, audio):     #Method to Give Command
        engine = pyttsx3.init('sapi5')
        voices = engine.getProperty ('voices')
        engine.setProperty ('voice', voices[1].id)
        engine.say (audio)
        engine.runAndWait ()


    def takeCommand (self):    #Method to give Input as Voice
        r = sr.Recognizer ()
        with sr.Microphone() as source:
            print('Listening')
            r.pause_threshold = 0.7
            audio = r.listen (source)

            try:
                print("Recognizing")
                Query = r.recognize_google (audio, language='en-us')    #en= English
                print("the query is printed='", Query, "'")

            except Exception as e:      #handle Exception
                print("Please Speak Again")
                return "None"
            return Query

    def sysstop (self):        #Method to stop the system
        take = self.takeCommand()
        choice = take

        #Stopping the system
        if choice == 'emergency':
            print ("Shutting down Your System")
            WriteMemory (plc,0,1,S7WLBit,1)
            self.Speak ("System stopped")


if __name__ == '__main__':    #Run the Code
    Maam = Shutdown ()
    Maam.sysstop ()
```

5.3.4.3 2. Research Ethics Approval

UNISA

UNISA SOE ETHICS REVIEW COMMITTEE

Date: 10/03/2020

Dear Mrs. Kahiomba Son a Klangala

Decision: Ethics Approval from 10/03/2020 to 10/03/2023

FRC Reference # : 2019/CSET/SOE/ 005
Name : Mrs. Kahiomba Sonia Kiangala

Student #: 60988568

Staff #: N/A

Researcher(s): Name: Mrs. Kahiomba Sonia Kiangala
E-mail address: sokiangala@gmail.com
Telephone #: 0765703473/0126573607

Supervisor (s): Name: Prof. Zenghui Wang
Email: wangz@unisa.ac.za
Telephone: 0114713513

External Researchers: N/A

RECEIVED

Working title of research:

Application and optimization of artificial intelligence techniques for small to medium sized manufacturing enterprises in an industry 4.0 environment

Qualification: PhD

Thank you for the application for research ethics clearance by the Unisa SOE Ethics Review Committee for the above mentioned research. Ethics approval is granted for 5 years.

RECEIVED OFFICE OF THE DIRECTOR 10 -03- 2020 School of Engineering College of Science, Engineering and Technology

*The **negligible risk application** was **reviewed** by the SOE Ethics Review Committee on 10/03/2020 in compliance with the Unisa Policy on Research Ethics and the Standard Operating Procedure on Research Ethics Risk Assessment. The decision was approved on 10/03/2020.*

The proposed research may now commence with the provisions that:

1. The researcher(s) will ensure that the research project adheres to the values and principles expressed in the UNISA Policy on Research Ethics.
2. Any adverse circumstance arising in the undertaking of the research project that is relevant to the ethicality of the study should be communicated in writing to the SOE Committee.
3. The researcher(s) will conduct the study according to the methods and procedures set out in the approved application.
4. Any changes that can affect the study-related risks for the research participants, particularly in terms of assurances made with regards to the protection of participants' privacy and the confidentiality of the data, should be reported to the Committee in writing, accompanied by a progress report.
5. The researcher will ensure that the research project adheres to any applicable national legislation, professional codes of conduct, institutional guidelines and scientific standards relevant to the specific field of study. Adherence to the following South African legislation is important, if applicable: Protection of Personal Information Act, no 4 of 2013; Children's act no 38 of 2005 and the National Health Act, no 61 of 2003.
6. Only de-identified research data may be used for secondary research purposes in future on condition that the research objectives are similar to those of the original research. Secondary use of identifiable human research data require additional ethics clearance.
7. No field work activities may continue after the expiry date 10/03/2025. Submission of a completed research ethics progress report will constitute an application for renewal of Ethics Research Committee approval.
8. Field work activities may only commence from the date on this ethics certificate.
9. [Permission to conduct research involving UNISA employees, students and data should be obtained from the Research Permissions Subcommittee (RPSC) prior to commencing field work.] AND/OR
10. [Permission to conduct this research should be obtained from the [company, CE organisation, DoE, etc name] prior to commencing field work.]

Add any other conditions if relevant.

Note:

*The reference number **2020/CSET/SOE/ 005** should be clearly indicated on all forms of communication with the intended research participants, as well as with the Committee.*

Yours sincerely,

Signature.....

Prof. E Onyari-Benecha

Chair of SOE ERC

E-mail: onyarek@unisa.ac.za

Tel: (011) 471-3379

Signature.....

Prof BB Mamba

Executive Dean : CSET

E-Mail: mambabb@unisa.ac.za

Tel: **(011) 670-9230**

12/03/2020