الجمهورية الشعبية الديمقراطية الجزائرية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

المدرسة العليا للإعلام الآلي - 08 ماي 1945 – بسيدي بلعباس

Ecole Supérieure en Informatique

-08 Mai 1945- Sidi Bel Abbes

MEMOIRE

En Vue de l'obtention du diplôme de **Master**

Filière : **Informatique**

Spécialité : **Ingénierie des Systèmes Informatiques (ISI)**

Thème

Cognitive Digital Twins

for IoT Resilience and Prevention

Présenté par **:**

- Mme SMATI Meriem

Soutenu le : **04/07/2023** Devant le jury composé de :

- M. BENSENANE Hamdane	Président
- M. RAHMOUN Abdellatif	Encadrant
- M. LAVAL Jannik	Encadrant
- M. NIANG Boubou-Thiam	Co-Encadrant
- M. KHALDI Miloud	Examinateur

Année Universitaire : 2022 / 2023

Acknowledgments

First and foremost, I begin by expressing my deepest gratitude and thanks to ALLAH for granting me the opportunities, guidance, good health and strength to accomplish this work and achieve this valuable success.

I would also like to extend my heartfelt appreciation to my most important people, the ones that gave me life: my parents, Abdelmadjid Smati and Samira Seddiki, for their endless love, encouragement, and support. Their belief in my abilities and their sacrifices have been the driving force behind my accomplishments. I am profoundly grateful for their guidance and presence in my life.

To my sisters, Wissem and Sara, and my brother Mohamed, I am indebted for their constant support and understanding. Their presence and encouragement have been a source of strength and motivation throughout my internship journey.

To my grandmothers, that have been cheering me up all along.

To my aunt Amel Hammouche, I am grateful for your presence and constant support throughout both favorable and challenging moments. Your continuous encouragement and guidance have been truly appreciated.

I would also like to express my heartfelt gratitude to my childhood friend, Ines, my friends Sarah, Abir and Ahmed, thank you for your contribution in this project, Your assistance in both technical and emotional aspects, thank you for your friendship, laughter, and support, it has been a constant source of strength and joy.

Special gratitude is also extended to my supervisors, Abdellatif Rahmoun, Jannik Laval and Boubou Thiam Niang, for their valuable guidance, expertise, and mentorship. Their support and dedication have been instrumental in shaping my work and expanding my knowledge.

To all the ESI SBA family, to all these individuals, I extend my sincerest thanks.

Your belief in me, your unwavering support, and the love and encouragement you have shown have been crucial in my personal and professional growth. I am deeply grateful for the impact you have had on my life and for being part of this important chapter.

Dedication

I dedicate this humble work to my dear parents, you are the individuals who instilled in me the values of life and insured I had everything I needed and showered me with love, affection and joy.

In the memory of my grand fathers, papi and jeddou Lkhloufi who taught us the importance of never giving up and persevering in our endeavors.

To my lovely grandmothers, mami and mamara whose unwavering efforts are aimed at our happiness.

To my brother and sisters, Wissem, Sara and Mohamed who have consistently provided their support.

To my aunt Tatina who accompanied me during my whole journey and contributed in this accomplishment.

May they find here all my affection and love.

Abstract

The concept of Digital Twins (DTs) has been evolving to include cognitive capabilities, leading to the emergence of Cognitive Digital Twins (CDTs).

CDTs are digital representations of physical systems that are augmented with cognitive capabilities to execute autonomous activities. They comprise a set of semantically interlinked digital models related to linking and retrieving heterogeneous data, as well as descriptive and simulation models. The CDT concept enhances the cognition capabilities of DTs with semantic technologies, enabling them to be more intelligent, comprehensive, and provide full lifecycle representation of complex systems.

This paper explores the potential of CDTs in enhancing perturbation resilience and maintenance. a super-Digital twin is realized that not only replicates the system's actions, but generates perturbations and anomalies to maintain and empower the system's security.

Keywords: Digital Twins; Cognitive Digital Twins ; Internet of Things; Resilience; Artificial Intelligence; Machine Learning; Deep Learning

Résumé

Le concept de Digital Twins (DTs) a évolué pour inclure des capacités cognitives, ce qui a conduit à l'émergence des Cognitive Digital Twins (CDTs). Les CDTs sont des représentations numériques de systèmes physiques qui sont enrichies de capacités cognitives pour exécuter des activités autonomes. Ils comprennent un ensemble de modèles numériques sémantiquement interconnectés liés à la liaison et à la récupération de données hétérogènes, ainsi que des modèles descriptifs et de simulation. Le concept de CDT améliore les capacités cognitives des DTs grâce aux technologies sémantiques, leur permettant d'être plus intelligents, exhaustifs et de fournir une représentation du cycle de vie complet des systèmes complexes.

Ce document explore le potentiel des CDTs dans l'amélioration de la résilience aux perturbations et de la maintenance. Un super-digital twin est réalisé, qui non seulement reproduit les actions du système, mais génère des perturbations et des anomalies pour maintenir et renforcer la sécurité du système. Le mémoire de master examine les possibilités offertes par les CDTs pour améliorer la résilience aux perturbations et la maintenance des systèmes complexes. Il propose également un cadre de recherche pour déterminer quand et comment un digital twin doit être enrichi de capacités cognitives. L'utilisation des CDTs dans les systèmes de fabrication est également explorée.

Mots Clés: Jumeaux numériques ; Jumeaux numériques cognitifs ; Internet des objets ; Résilience ; Intelligence artificielle ; Apprentissage automatique ; Apprentissage profond

ملخص

لقد تطور مفهوم التوائم الرقمية ليشمل القدرات المعرفية ، مما أدى إلى ظهور التوائم الرقمية المعرفية.

التوائم الرقمية المعرفية هي تمثيلات رقمية للأنظمة المادية التي يتم تعزيزها بالقدرات المعرفية لتنفيذ الأنشطة المستقلة. وهي تتألف من مجموعة من النماذج الرقمية المترابطة بشكل كبير والمتعلقة بربط واسترجاع البيانات غير المتجانسة ، فضلاً عن النماذج الوصفية والمحاكاة.
يعزز مفهوم التوائم الرقمية المعرفية القدرات الإدراكية للتوائم الرقمية باستخدام التقنيات الدلالية ، مما يمكّنهم من أن يكونوا أكثر وضوحًا وشمولية ويوفر تمثيلاً كاملاً لدورة الحياة للأنظمة المعقدة.

تستكشف هذه الورقة إمكانات التوائم الرقمية المعرفية في تعزيز مقاومة الاضطرابات وصيانتها. يتم تحقيق التوأم الرقمي الفائق الذي لا يكرر فقط إجراءات النظام ، ولكنه يولد الاضطرابات والشذوذ للحفاظ على أمان النظام وتمكينه.

الكلمات الرئيسية: التوائم الرقمية. التوائم الرقمية المعرفية. انترنت الأشياء؛ صمود؛ الذكاء الاصطناعي؛ التعلم الالي؛ تعلم عميق

Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

DT Digital Twin

IDMU Integral Digital Mock-Up

NASA National Aeronautics and Space Administration

IoT Internet of Things

GE General Electric

QoS Quality of Service

AI Artificial Intelligence

ML Machine Learning

ITU International Telecommunication Union

NGN Next-Generation Networks

RFID Radio-Frequency IDentification

BLE Bluetooth Low Energy

PLM Product Life-cycle Management

DTP Digital Twin Prototype

DTI Digital Twin Instance

PT Physical Twin

CMMs Coordinate Measuring Machine

VVA Verification, Validation and Accreditation

BD Big Data

DL Deep Learning

CPSs Cyber-Physical Systems

IT Information Technology

MAPE-K Monitor-Analyze-Plan-Execute over a shared Knowledge

KPIs Key Performance Indicators

CDT Cognitive Digital Twin

UI User Interface

MES Manufacturing Execution System

ERP Entreprise Resource Planning

WMS Warehouse Management System

API Application Programming Interface

NN Neural Network

KNN K-Nearest Neighbour

SVM Support vector machine

PCA Principal Component Analysis

SVD Singular Value Decomposition

HMM Hidden Markov model

MLPs Multilayer Perceptrons

CNNs Convolutional Neural Networks

CNN Convolutional Neural Network

RNNs Recurrent Neural Networks

LSTM Long Short-Term Memory

CSDT Cognitive Super-Digital Twin

Part I

General Introduction

Chapter 1

General Introduction

Contents

1 Context

Since its advent, and with the emergence of new technologies such as 5G, Artificial Intelligence (AI) and edge computing, IoT has continued to evolve and has become more powerful and capable of transforming the way businesses operate. And with the increasing use of IoT devices in businesses, the need of enhancing the resilience and maintenance of these devices has been indispensable especially since they need to continue to function optimally and securely even if there are disruptions, including cyber-attacks, power outages, and physical damage. Considering the growing number of connected devices and the data they generate, it's important for businesses to have systems and processes in place to manage and maintain these devices.

One approach to improving the resilience of IoT systems and avert calamities is to design the infrastructure and industrial systems with redundancy and failover mechanisms. Redundancy involves designing systems with backup devices, networks, or data centers to ensure that if one component fails, the system can continue to function. However, implementing this methodology would require a significant investment of both finances and resources. As a result, instead of depending on redundancy, "Digital Twin (DT)" have been employed.

2 Problem Statement

Maintenance and resilience are critical aspects of any system or asset, from a simple household appliance to a complex manufacturing plant or a transportation network.

Effective maintenance practices help to ensure that equipment and systems operate as intended, with minimal downtime, reduced risk of failures, and increased efficiency. Resilience, on the other hand, refers to the ability of a system to withstand disruptions or shocks and recover quickly from them, while maintaining its essential functions and capabilities.

However, traditional methods of maintenance are often reactive, based on fixing problems after they occur, rather than preventing them from happening in the first place. This approach can be time-consuming, costly, and not always effective in detecting and preventing failures, particularly in complex and interdependent systems. Moreover, the increasing complexity of systems and equipment, coupled with the growing demand for reliability and efficiency, makes maintenance and resilience even more challenging.

3 Objectives

In this paper, we aim to:

- Define the DT concepts and principles since it has a highly adaptive nature to to meet the specific needs of their intended use case.
- Analyze and compare the various methodologies employed in constructing a DT and using it in the field of resilience.
- Introduce the framework CSDT.

4 Thesis Organization

The given work is organized into four chapters, which are outlined as follows:

- **Chapter 1 « General Introduction » :** In this first chapter, we provide a contextual analysis and explicate the problem statement and objectives of our study.

- **Chapter 2: « Background and Definition » :** This second chapter aims to provide a comprehensive overview of the various components which are pertinent to the topic at hand.

- **Chapter 3: « State of the Art » :** This third chapter procures a comprehensive analysis of the various works examined, followed by a detailed discussion of each approach, culminating in a comparative study and synthesis.

- **Chapter 4: « Conclusion » :** This final part marks the culmination of the research conducted in this paper and serves as the introduction to the proposed Digital Twin and key findings presented in the final project report.

Part II

Background and Definitions

Chapter 2

Background

Contents

1 IoT

1.1 Introduction

The IoT is a network of interconnected physical devices that communicate with each other and exchange data over the internet. These devices can range from smartphones and laptops to sensors, cameras, and appliances. The IoT is transforming many industries, from healthcare and transportation to manufacturing and agriculture. Its devices are often embedded with sensors that collect data, which can be analyzed to gain insights and inform decision-making.

In this section, different definitions, architectures and components concerning this topic will be seen.

1.2 Definition of IoT

IoT or Internet of things, the backbone of the Digital Age, the technology that has the power and potential to revolutionize numerous industries, has no standard definition yet. But here are a few definitions that has been collected:

- The term IoT is defined by the International Telecommunication Union (ITU) as a global infrastructure for the information society that enables the interconnection of different assets based on communication technologies. [1]. And in terms of a network that it is “Available anywhere, anytime, by anything and anyone" [2].

- ITU-T Study Group 13 leads the work of the ITU on standards for Next-Generation Networks (NGN) and future networks (ITU, SERIES Y, 2005). It has defined IoT as: “A global infrastructure for the information society, enabling advanced services by connecting physical and/or virtual things based on existing and evolving interoperable information.” [2]

- In [3], it has been defined as “a network of physical objects. The internet has transformed from being solely a network of computers into a vast network encompassing devices of all types and sizes. This includes vehicles, smartphones, home appliances, toys, cameras, medical instruments, industrial systems, and even animals. All connected ,all communicating and sharing information based on stipulated protocols in order to achieve smart reorganizations, positioning, tracing, safe and control and even personal real time online monitoring , online upgrade, process control and administration. ”

1.3 Historical Evolution of IoT

The Internet of Things has its root in the early days of the Internet, but it was not defined until 1999 when the term "Internet of Things" was coined by Kevin Ashton[1], ideas, research or studies on subjects related to IoT have been around for some time.

Since its creation, IoT has been constantly evolving through several stages of technological development and expanding into new areas of application, It is much more than just a simple technology. It has become an integral part of daily life, increasing efficiency and comfort. [4]

In [5], it is mentioned that the first internet appliance was a Coke machine at Carnegie Melon University in the early 1980s. The programmers had the ability to connect to the machine to the internet and remotely monitor the status of the machine, see if there were drinks available and determine the temperature of the drinks.

From there, the idea of interconnected devices proliferated. In the 1990s, the auto industry pioneered the use of RFID (radio-frequency identification) technology to track inventory in factories and warehouses. By the early 2000s, early examples of

[1]Kevin Ashton is a British technology pioneer ans his work in Radio-Frequency IDentification (RFID) technology and supply chain management paved the way for the development of the Internet of Things.

the internet of things began to emerge in the commercial market. One such example was the introduction of telematics – the combination of telecommunications and informatics – into vehicles, enabling navigation, vehicle diagnostics, and on-demand entertainment.

Presumably, before 2025, IoT will have a significant impact on daily life. IoT can be used in Electronic Voting, Electronic Identifications and in Medical Field to Support Patients. Robots are working in several sections using IoT. Remote sensing Robots are also Using IoT. IoT based systems are widely used in Farming. Remote sensing robots are collecting data with IoT protocols. [4]

1.4 Characteristics of IoT

The fundamental characteristics of the IoT are defined in [3] as follows:

Interconnectivity: anything has the potential to be interconnected with the global information and communication infrastructure.

Things-related services: The IoT can provide services related to physical objects while taking into account factors such as protecting privacy and maintaining consistency between the virtual and physical aspects of the objects. To achieve this, changes are required to both the technologies used in the physical world and the information world. These changes are necessary to ensure that thing-related services can be provided within the limitations and requirements of physical objects.

Heterogeneity: The IoT devices exhibit heterogeneity because they operate on varying hardware platforms and networks, and utilize different networks to communicate with other devices or service platforms.

Dynamic changes: The condition of devices in the IoT is subject to dynamic changes, such as transitioning between sleep and wake modes, connection and disconnection, as well as variations in contextual factors such as location and speed. Additionally, the quantity of devices in the network can also fluctuate dynamically.

Enormous scale: At a minimum, the amount of devices necessitating management and intercommunication within the IoT will be ten times larger than the devices currently linked to the Internet.

Safety: While reaping the advantages of the IoT, it's crucial not to overlook safety concerns. As both the originators and beneficiaries of the IoT, we need to prioritize safety considerations. This encompasses safeguarding our personal information and our physical welfare. To ensure comprehensive security, we must develop a security framework that can scale to secure the endpoints, networks, and the data transmitted across them.

Connectivity: The ability to connect facilitates access and compatibility within a network. Accessibility pertains to being able to join a network, whereas compatibility refers to the capacity to exchange and utilize data in a standardized manner.

1.5 IoT Architectures

As mentioned, IoT has brought a significant change in the manner and the way we interact with physical objects and devices located in our environment. It facilitates communication and information exchange between them over the internet. Nevertheless, due to the vast number of devices involved and the complexity of the network infrastructure required to support them, designing and implementing IoT architecture is an essential starting point.

There are various and plenty types of IoT architectures, each with its unique advantages and challenges and depend fully on the corresponding use case and subject. The choice of architecture relies as well on several other factors beside specific use case, such as the network topology, the scalability, the reliability, and cost-effectiveness.

In this subsection, the most common IoT architectures are mentioned:

1.5.1 The Three and Five Layered Architectures

The article [6] talked about two architectures that are widely used: The three and five-layered architectures that are represented in Figure 2.1

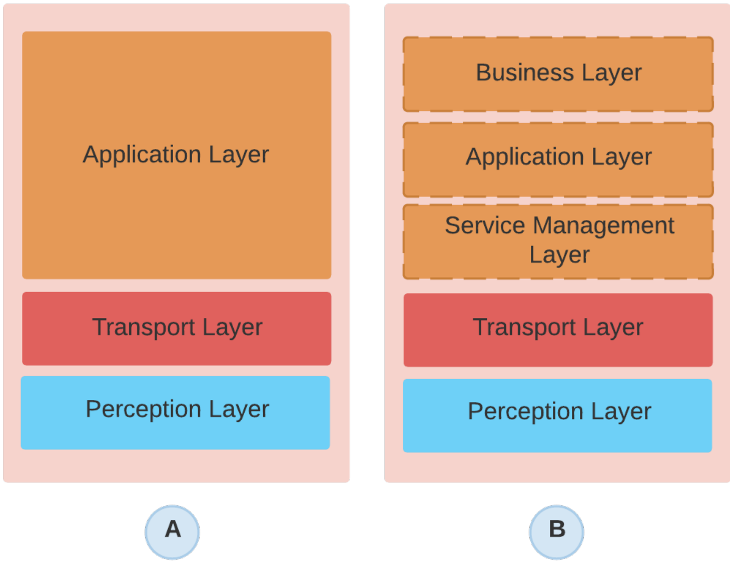

Figure 2.1: A. IoT layered architecture three layered and B. five layered architecture.

- **Perception Layer:** The perception layer is composed of a set of group of objects. These objects act as a bridge between the Physical world and the digital realm utilizing sensors to capture data. Its main objective is to gather information from the environment through a range of sensors for example temperature, humidity, light, CO2 sensors, cameras, etc., according to the specific use case and needs of the application. Researchers are primarily concerned with ensuring the proper identification, management, and security of these objects within this layer.

- **Transport Layer:** The purpose of this layer is to establish secure connections between objects and facilitate the sharing of information among them. Different communication protocols such as Ethernet, WiFi, Wi-MAX, ZigBee, and Bluetooth Low Energy (BLE) can be used to enable this information exchange. However, there are still certain challenges that need to be addressed at this layer, such as reducing energy consumption in the network, ensuring Quality of Service (QoS), and adapting to dynamic topologies.

- **Service Management Layer:** It can also be referred to as "the Middleware Layer", this level facilitates and enables the integration of diverse and heterogeneous devices into IoT applications. Additionally, Moreover, it plays a pivotal role in processing raw data collected by objects in the perception layer. This data is typically characterized by its large volume and diverse nature.

- **Application Layer:** The layer in question is primarily tasked with providing application-specific and use case-specific services to end-users, that is why it plays a significant role in enhancing the convenience, safety and overall quality of life of end-users. However, the ability to tailor services to meet specific needs and preferences makes this layer critical in the success and adoption of IoT applications. As such, developers and researchers must continually work to identify and address the unique challenges associated with providing application-specific services in this layer.

- **Business Layer:** The business layer serves as the supervisor of an IoT system's operations and services, utilizing raw data acquired from other layers to create flow charts, graphs, and business models. Additionally, this layer is responsible for monitoring, analyzing, and evaluating the IoT system and its related components. Decision-making is a central activity of the business layer, as it plays a critical role in determining the direction and success of the IoT system.

1.5.2 The Edge Fog Cloud

Edge, fog, and cloud computing are different types of data storage and management in IoT.

Edge computing refers to computation at the edge of a device's network, while fog computing is an extension of cloud computing that acts as a layer between the edge and the cloud.

Fog computing is designed to overcome the challenges of edge computing, such as delays in detection, by processing data in real-time.

The cloud, on the other hand, refers to the on-demand delivery of IT services/resources over the internet.

Down below each layer is explained taken from [7].

- **Edge:**

  Edge computing involves processing data locally within the network, specifically on edge devices and gateways, instead of relying on centralized storage. By avoiding data transfer to the cloud, it enables quick response times and unmatched speed.

  When it comes to decentralized storage, edge computing stands as the most secure option. Unlike cloud storage, which distributes data across numerous servers, edge computing employs a vast number of local nodes, potentially reaching into the thousands. Each device within the edge network can function as an independent server, making it extremely difficult for hackers to breach. Gaining synchronized access to thousands of dispersed devices is practically unattainable.

  This distinction also sets fog computing apart from edge computing. Fog computing serves as a network that connects to the cloud, while edge devices operate with loose connections and have the ability to act autonomously.

- **Fog:**

  Fog computing serves as an intermediary layer positioned between the conventional centralized data storage system (cloud) and edge devices. Its purpose is to extend the capabilities of the cloud by bringing computation and data storage closer to the edge. Fog encompasses multiple nodes, known as fog nodes, forming a decentralized ecosystem—this stands as the primary contrast between fog and cloud computing.

  When data reaches the fog layer, the individual node determines whether to process it locally or transmit it to the cloud. Consequently, the data remains accessible even offline since certain portions of it are stored locally. This presents another significant divergence between fog computing and cloud computing, as the latter relies on remote servers to execute and store all the intelligence and computations.

- **Cloud:**

  It is a centralized storage situated further from the endpoints than any other type of storage. This explains the highest latency, bandwidth cost, and network requirements. On the other hand, cloud is a powerful global solution that can handle huge amounts of data and scale effectively by engaging more computing resources and server space. It works great for big data analytics, long-term data storage and historical data analysis.

2 Internet of Things Maintenance

2.1 Introduction

When it comes to equipment maintenance, there are two prevailing perspectives [8].

- The first approach, commonly adopted, involves waiting until a piece of equipment or machinery breaks down before fixing it. While this approach may suffice for certain aspects of life, it is impractical for large-scale industrial organizations that depend on numerous assets for their daily operations.

- A more prudent strategy for such enterprises entails taking a proactive maintenance approach. This approach involves conducting regular repairs on equipment to prevent failures and disruptions. However, identifying equipment issues is not always apparent to the naked eye. Therefore, organizations may require technological assistance, such as software, to gain a better understanding of their assets' performance and anticipate potential problems. One valuable tool in this regard is IoT predictive maintenance.

2.2 Definition of Maintenance

Maintenance refers to the process of preserving or restoring something to a satisfactory condition by performing regular checks, repairs, or updates. It can be applied to various aspects of life, such as mechanical systems, buildings, software, and even personal health. The primary goal of maintenance is to prevent the breakdown or failure of a system or object, increase its lifespan, and ensure it remains safe and functional. In essence, maintenance involves proactive measures to avoid problems and ensure continued smooth operation.

Maintenance in the context of IoT infrastructures involves a series of tasks designed to identify and correct issues before they cause downtime or other problems. These tasks may include software updates, firmware updates, hardware repairs or replacements, and regular inspections to monitor the performance of the system.

2.3 IoT Predictive Maintenance

IoT predictive maintenance is a maintenance strategy that utilizes the IoT to collect and analyze data concerning assets, equipment, or machinery. Through the use of sensors and other instruments, data regarding the status of the equipment is gathered, enabling the detection of any potential issues that should be addressed to prevent future outages and avoid unnecessary downtime.

In the context of IoT predictive maintenance, the Internet of Things typically involves sensors and monitors that are either placed on or integrated into equipment. These sensors monitor various variables that may indicate potential equipment issues. The collected data is then transmitted to other components within the network, such as predictive maintenance software or other smart manufacturing systems. By continuously collecting and transmitting real-time equipment performance data, other IoT technologies can conduct predictive maintenance analytics. These analytics help identify potential issues that could lead to equipment failure. This process enables organizations to better anticipate the likelihood of outages or disruptions, allowing them to adopt a proactive maintenance approach [8].

To perform predictive maintenance, equipment health and performance are assessed through periodic or continuous monitoring of asset conditions. The data obtained from IoT devices, which connect various systems and assets, enables businesses to anticipate, plan for, and take proactive measures in response to potential events such as equipment failure or parts repair before they occur. Predictive maintenance is typically carried out during normal working conditions to avoid disrupting business operations [9].

3 Internet of Things Resilience

3.1 Introduction

Resilience is the ability of a system to recover from disruptions and continue to function effectively. In the context of technology, resilience is becoming increasingly important as our reliance on interconnected systems grows. Disruptions such as cyber-attacks, natural disasters, and equipment failures can have significant consequences for businesses and individuals. Resilient systems are designed to minimize the impact of such disruptions and ensure continuity of service.

3.2 Definition of Resilience

Resilience can be defined as the capacity to adapt, adjust and the ability of a system to continue operating and delivering services in the face of various and different types of failures or disruptions, such as hardware or software failures, network outages, cyber attacks, or natural disasters. Resilient computer systems are designed to anticipate and withstand these challenges, and to recover quickly and efficiently in the event of a failure or disruption by implementing redundancy, fault tolerance, and disaster recovery mechanisms, as well as conducting regular testing and maintenance to ensure that the system remains robust and reliable. Resilience is a critical attribute of modern computer systems, particularly those that are used to provide essential services or support critical business operations.

3.3 IoT resilience

In the context of IoT, resilience refers to the capability of IoT systems to maintain reliable and secure connectivity, data transmission, and functionality in the face of various challenges and disruptions, such as network congestion, hardware failures, cyber attacks, or power outages.

It also refers to the ability of IoT systems to resist perturbances, recover from emergencies, and continue functioning in the face of disruptions. There are several scientific efforts to make IoT systems resilient, and AWS IoT Core features data redundancy and specific features for data resiliency, such as device shadow and AWS IoT Device Advisor. However, AWS IoT Core resources are region-specific and not replicated across regions unless specifically done so. Resilience is increasingly important as IoT becomes a critical part of the global internet [10] [11].

4 Machine Learning

4.1 Definition of Machine Learning

Machine learning is a branch of AI and computer science that focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy [12]. It allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so.

It holds significance as it provides enterprises with insights into customer behavior trends and operational patterns, facilitating the creation of innovative products. Prominent companies like Facebook, Google, and Uber have embraced ML as a fundamental aspect of their operations, establishing it as a crucial factor for gaining a competitive edge [13].

4.2 Types of Machine Learning

ML algorithms can be broadly classified into three types:

- Supervised Learning.
- Unsupervised Learning.
- Reinforcement Learning.

Figure 2.2 represents a diagram that illustrates the different ML algorithm, along with the categories.

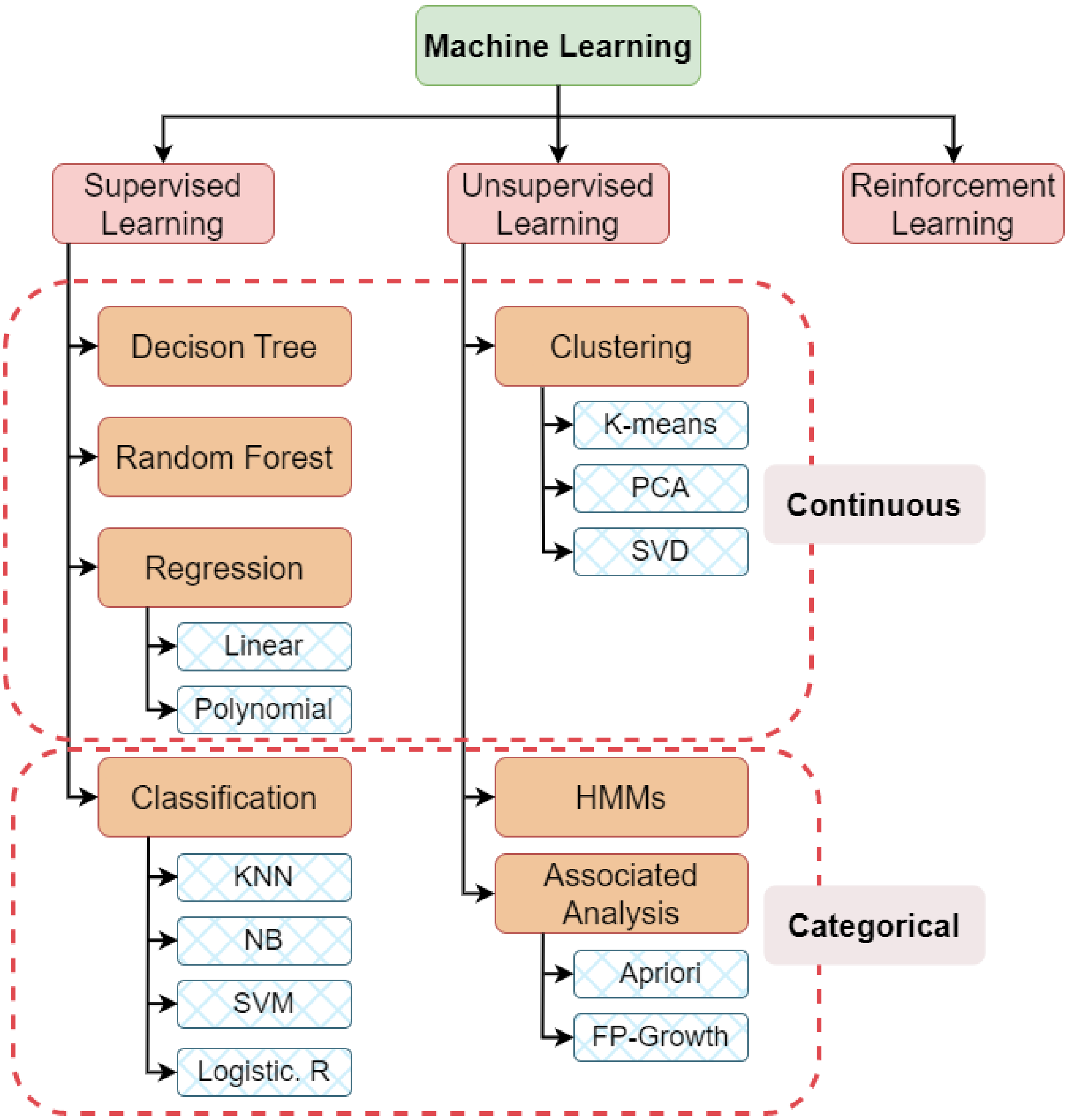

Figure 2.2: Machine learning Types and Algorithms.

4.2.1 Supervised Learning

Supervised learning is a category within ML that relies on external guidance for the machine to learn [14]. In supervised learning, models are trained using labeled datasets [12]. Following training and processing, the model is evaluated by providing it with sample test data to determine if it accurately predicts the desired output.

The objective of supervised learning is to establish a mapping between input data and output data. It mirrors the concept of a student learning under the supervision of a teacher. An example of supervised learning is spam filtering.

Supervised learning can be further categorized into two types of problems:

- Classification.
- Regression.

4.2.2 Unsupervised Learning

Unsupervised learning is an algorithmic approach in ML that examines and clusters datasets lacking pre-existing labels to uncover patterns and insights [12].

Unlike supervised learning, unsupervised learning does not rely on a training dataset to guide the models. Instead, the models autonomously discover concealed patterns and group the data based on similarities and dissimilarities [15].

Unsupervised learning is employed to unveil the underlying structure of datasets and finds applications across diverse domains, aiding in data feature summarizing and explanation.

Additionally, it serves as a means of testing AI and is capable of performing more intricate processing tasks compared to supervised learning systems [12].

Hence further, it can be classified into two types:

- Clustering.
- Association.

Examples of some Unsupervised learning algorithms are K-means Clustering, Apriori Algorithm, Eclat, etc.

4.2.3 Reinforcement Learning

Reinforcement Learning is a form of ML that allows an agent to learn within an interactive environment through trial and error, utilizing feedback obtained from its own actions and experiences [16].

This approach revolves around rewarding desired behaviors and penalizing undesired ones.

The primary focus of reinforcement learning is determining how intelligent agents should take actions in an environment to maximize cumulative rewards [17].

Reinforcement learning algorithms acquire knowledge from outcomes and make decisions about the subsequent actions to be taken. It has demonstrated successful applications in various domains, such as robot control, elevator scheduling, telecommunications, backgammon, checkers, and Go.

Reinforcement learning serves as a valuable technique for automated systems seeking to identify the optimal behavior or path in specific situations. Q-Learning algorithm is used in reinforcement learning.

Figure 2.3[2], realized by Thomas Malone, represent the way on what Machine Learning models can perform.

4.3 Machine Learning Models

4.3.1 Decision Trees

The decision tree is a supervised learning algorithm primarily employed for solving classification problems, although it can also tackle regression problems. It ac-

[2]See: https://bit.ly/3gvRho2

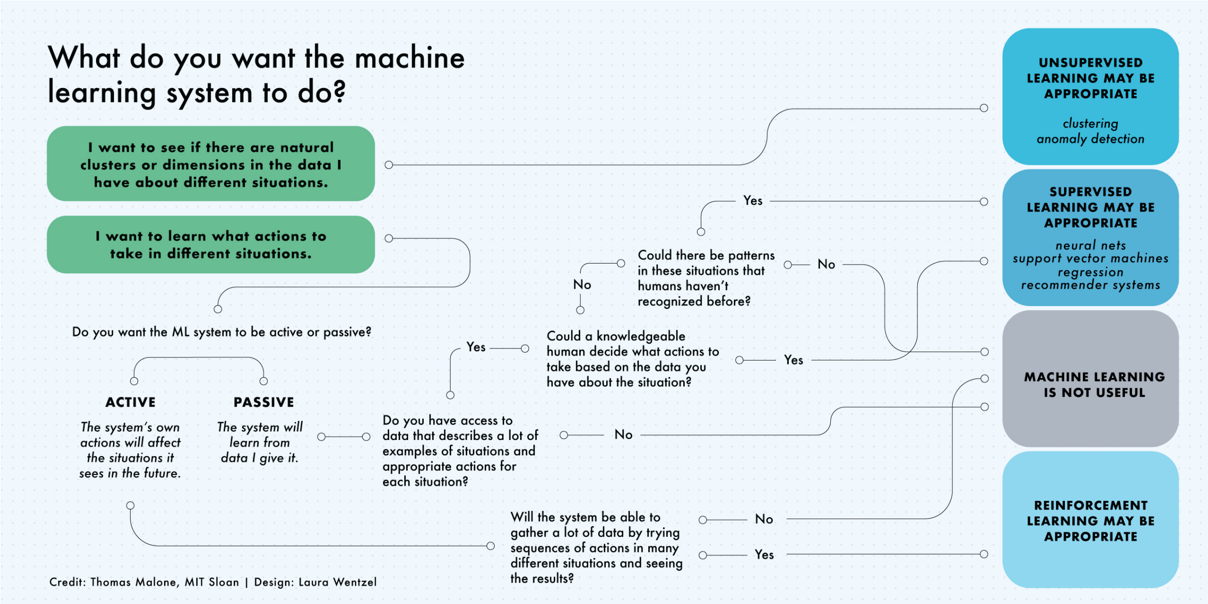

Figure 2.3: Source: Thomas Malone | MIT Sloan.

commodates both categorical and continuous variables [14].

The decision tree presents a tree-like structure comprising nodes and branches to represent decisions and their possible consequences (Figure 2.4). It initiates with a root node and further branches out to leaf nodes. Internal nodes represent dataset features, branches denote decision rules, and leaf nodes signify the problem's outcomes [18].

Decision tree algorithms find practical application in various real-world scenarios. For instance, they are utilized in distinguishing between cancerous and non-cancerous cells and providing car purchase recommendations to customers.

4.3.2 Random Forests

Random forest is a supervised learning algorithm employed in ML for both classification and regression tasks. It operates as an ensemble learning technique, leveraging multiple classifiers to generate predictions and enhance the model's performance [14].

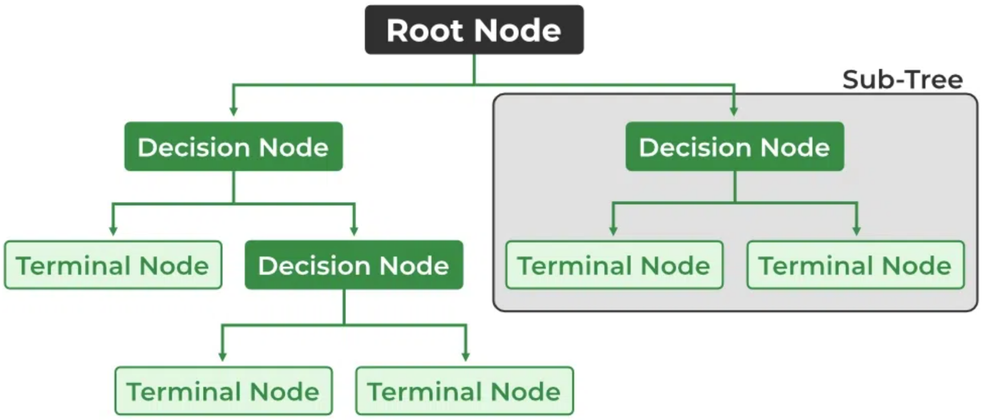

Figure 2.4: Decision Tree

This approach encompasses numerous decision trees that operate on subsets of the provided dataset, amalgamating their outcomes to improve predictive accuracy. It is recommended to have a random forest consisting of 64 to 128 trees, as a higher number of trees typically leads to increased algorithmic precision.

In other words, The fundamental concept behind random forest is the wisdom of crowds. A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models. It forest uses bagging and feature randomness when building each tree to ensure that the trees are uncorrelated [19].

When classifying a new dataset or object, each tree produces a classification result, and the algorithm predicts the final output based on majority voting.

Random forest demonstrates efficient processing capabilities, making it suitable for handling missing and inaccurate data. Additionally, it offers a swift execution speed.

4.3.3 Naive Bayes

The Naive Bayes classifier is a supervised learning algorithm utilized for making predictions by considering the probability of an object. It derives its name from Bayes theorem, as it follows the assumption that variables are independent of each other, hence "naïve." [14]

Naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable.

Despite their naive design and oversimplified assumptions, Naive Bayes classifiers have worked well in many complex real-world situations [20].

Bayes theorem, on which this algorithm is based, deals with conditional probability. It calculates the likelihood of event A occurring given that event B has already taken place. The equation for Bayes theorem is expressed in Equation (2.1).

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{2.1}$$

Naïve Bayes classifier is one of the best classifiers that provide a good result for a given problem. It is easy to build a naïve bayesian model, and well suited for the huge amount of dataset. It is mostly used for text classification.

4.3.4 K-Nearest Neighbors

The K-Nearest Neighbour (KNN) algorithm is a supervised learning technique applicable to both classification and regression problems. It operates by establishing the similarities between a new data point and existing data points. Utilizing these similarities, the algorithm categorizes the new data point into the most similar class. It is also referred to as a "lazy learner" algorithm because it retains all available datasets and classifies each new instance with the assistance of its K-nearest neighbors.

To assign the new instance to the most similar class, KNN calculates the distance between data points using a distance function. Common distance functions include Euclidean, Minkowski, Manhattan, or Hamming distance, chosen based on the specific requirements of the problem [14].

5 Deep Learning

5.1 Definition of Deep Learning

DL, a subset of ML, employs ANN comprising multiple layers to extract high-level features from raw input data. It mimics the human learning process and is considered a form of AI.

The algorithms used in DL are organized hierarchically, with each layer growing in terms of complexity. They find application in various tasks, including supervised and unsupervised learning, such as speech recognition, image classification, and natural language processing. Deep learning plays a crucial role in data science, which encompasses statistics and predictive modeling, offering significant advantages to data scientists responsible for gathering, analyzing, and interpreting large volumes of data.

5.2 Deep Learning Models

Several DL algorithms are widely used, including Multilayer Perceptrons (MLPs), CNNs, Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Radial Function Networks, and Self-Organizing Maps.

MLPs are considered the fundamental and oldest deep learning algorithm. CNNs are particularly effective for image and video recognition tasks, while RNNs and LSTMs are commonly employed for natural language processing and speech recognition. Radial Function Networks and Self-Organizing Maps are utilized for clustering and classification purposes.

As mentioned before, DL algorithms are designed to run dynamically through multiple layers of NN, with pre-training specifically tailored to the given task.

A few of the cited models are about to be presented down below:

5.2.1 CNNs

A Convolutional Neural Network (CNN) is a widely utilized neural network architecture in the realm of AI's Computer Vision field [21]. Popular type of neural network architecture used in the field of Computer Vision within Artificial Intelligence. Computer vision enables computers to interpret and understand visual data, such as images. In the realm of Machine Learning, Artificial Neural Networks exhibit strong performance. They are employed in various datasets encompassing images, audio, and text. Different types of Neural Networks serve different purposes. For instance, Recurrent Neural Networks, particularly Long Short-Term Memory (LSTM) networks, are suitable for predicting word sequences, while Convolutional Neural Networks are commonly used for image classification. A typical Neural Network consists of three types of layers:

- Input Layers: This initial layer receives the input data for the model. The number of neurons in this layer is equivalent to the total number of features in the data (e.g., the number of pixels in an image).

- Hidden Layers: The input from the Input layer is transmitted to the hidden layer(s). The number of hidden layers can vary depending on the model and the size of the data. Each hidden layer may contain a different number of neurons, typically exceeding the number of features. The output of each layer is computed by performing matrix multiplication between the output of the previous layer, which has learnable weights, and subsequently adding learnable biases. This is followed by an activation function, which introduces nonlinearity to the network.

- Output Layer: The output from the hidden layer is fed into a logistic function, such as sigmoid or softmax, which converts the output of each class into a probability score for that class. The data is fed into the model, and the output from each layer is obtained through a process called feedforward. Subsequently, the error is calculated using an error function, such as cross-entropy or square loss error. The error function measures the performance of the network. The next step involves backpropagation, where derivatives are calculated to minimize the loss. Backpropagation is essential for adjusting the model's parameters and improving its performance.

Convolutional Neural Network consists of multiple layers like the input layer, Convolutional layer, Pooling layer, and fully connected layers. The Convolutional layer applies filters to the input image to extract features, the Pooling layer down-samples the image to reduce computation, and the fully connected layer makes the final prediction. The network learns the optimal filters through backpropagation and gradient descent.

5.2.2 Multilayer Perceptrons

MLP serves as an extension of the feed-forward neural network. It encompasses three distinct layers as depicted in

the input layer, the output layer, and hidden layers, as depicted in Figure 2.5:

1. Input Layer.

2. Output Layer.

3. Hidden Layers.

The input layer receives the input signal for processing, while the output layer is responsible for performing tasks such as prediction and classification.

The true computational engine of the MLP resides within an arbitrary number of hidden layers positioned between the input and output layers. Similar to a feed-forward network, the data flows in a forward direction from the input layer to the output layer [22].

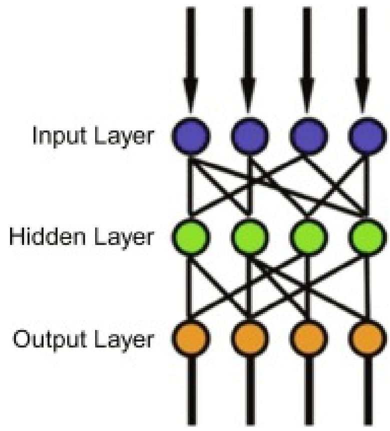

Figure 2.5: MLP with a Single Hidden Layer

In an MLP, the neurons are trained using the backpropagation learning algorithm. MLPs are specifically designed to approximate any continuous function and can effectively address problems that are not linearly separable. Prominent use cases of MLPs include pattern classification, recognition, prediction, and approximation.

5.2.3 Reccurent Neural Networks

An ANN known as a RNN is specifically designed for handling sequential or time-series data.

Unlike CNN, RNNs incorporate hidden states and allow the utilization of previous outputs as inputs. This enables RNNs to effectively process sequential data by utilizing the output from one time step as the input for the next step. RNNs find extensive applications in various fields such as natural language processing and speech recognition. Nonetheless, RNNs do have certain drawbacks, including challenges with training due to issues like the vanishing and exploding gradients [23].

Another constraint of traditional RNNs is the lack the ability to incorporate future inputs into the current state. Furthermore, RNNs encounter difficulties when dealing with long-term dependencies, which can result in problems such as gradient vanishing and exploding.

However, a solution to these limitations emerged in the form of Long Short-Term Memory Networks (LSTMs). LSTMs were introduced to address these shortcomings by enabling the learning of long-term dependencies through the retention of information over extended periods [24].

1. **Long Short Term Memory (LSTM):**

   LSTM is a specific type of ANN that finds application in DL and ML tasks. It serves as a variation of RNNs and exhibits the capability to effectively handle lengthy time-series data, enabling the learning of order dependencies in sequence prediction tasks.

   In contrast to conventional feedforward neural networks, LSTM incorporates feedback connections and possesses the ability to process not only individual data points but also complete data sequences.

   One of the primary objectives of LSTM is to address the challenge of long-term dependencies encountered by RNNs. While RNNs struggle to predict information stored in long-term memory, LSTM provides more accurate predictions by leveraging recent information. The structure of LSTM consists of a chain comprising four neural networks and incorporates memory blocks known as cells. These cells retain information, and the manipulation of memory is facilitated by specialized components called gates [25].

   Each recurrent neural network consists of a series of repeating neural network modules, forming a chain. These networks incorporate loops, allowing information to be retained within the network. Figure 1 illustrates a simple recurrent neural network with loops. In this figure, the neural network denoted as Figure 1, A takes the input xt and generates the output ht. The presence of a loop facilitates the transfer of data from one phase of the network to the next. LSTM is explicitly designed to tackle the problem of long-term dependencies. Each recurrent neural network is composed of a sequence of repeating neural network modules. To aid in comprehension of the subsequent sections, Table 2 presents a list of symbols that are utilized to explain the various concepts [25] [26].

Figure 2.6 illustrates a simple recurrent neural network with loops. LSTM takes the input xt and generates the output ht. The presence of a loop facilitates the transfer of data from one phase of the network to the next. LSTM is explicitly designed to tackle the problem of long-term dependencies. Each recurrent neural network is composed of a sequence of repeating neural network modules [26].

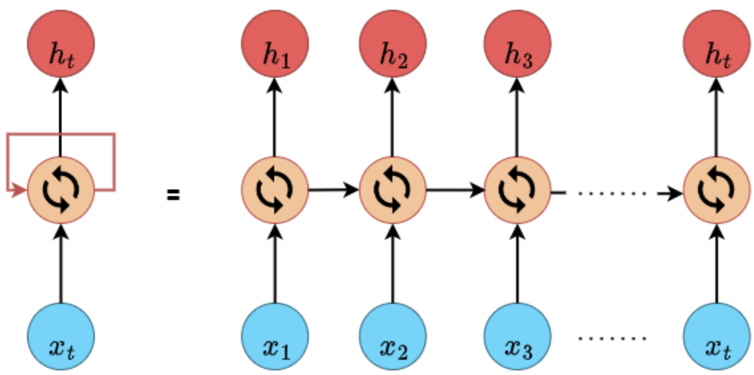

Figure 2.6: Simple Recurrent Neural Network

6 Machine Learning VS Deep Learning

Taking into account Figure 2.7:

- The functioning of ML models can be illustrated through the example of image recognition for distinguishing between cats and other animals.

In this scenario, the ML model takes images of cats as input. It then extracts distinct features from these images, such as shape, height, nose, eyes, and other relevant characteristics. By employing a classification algorithm, the model analyzes these features and generates a prediction as output.

- The functioning of DL can be comprehended using the same example of distinguishing mentioned previously.

  In DL models, the images serve as input and are directly fed into the algorithms, eliminating the need for manual feature extraction. The images traverse through various layers of an artificial neural network, allowing the model to predict the final output.

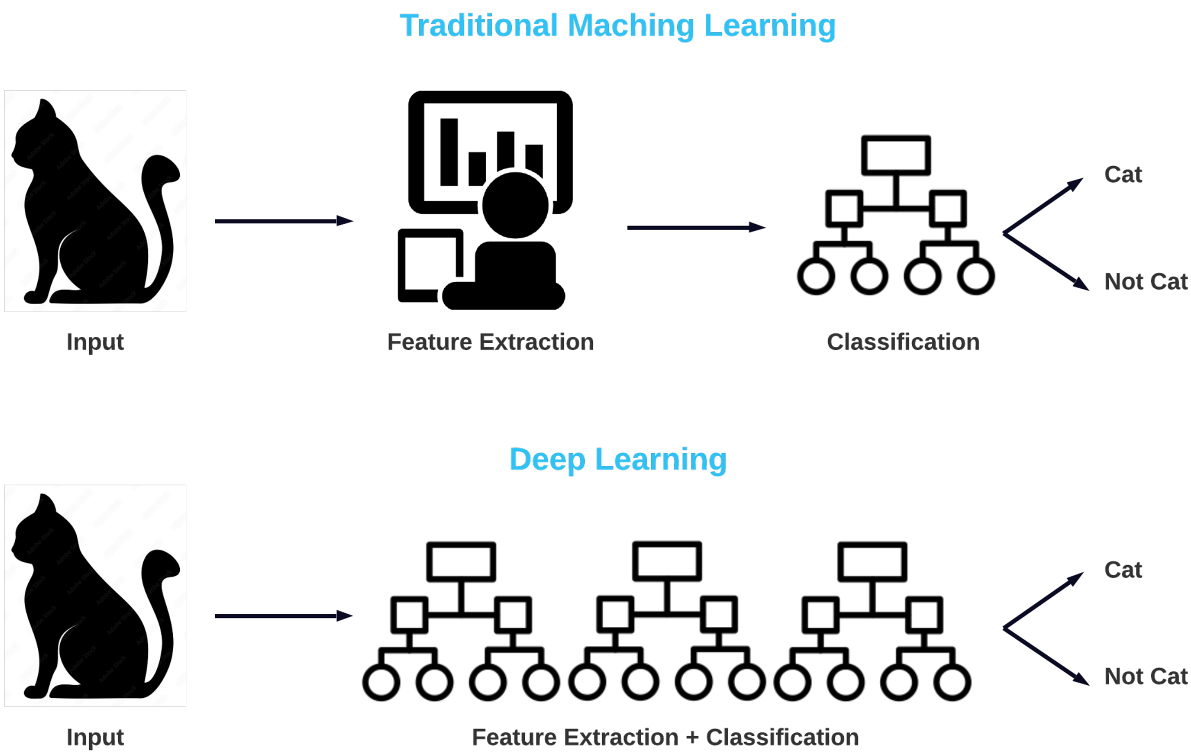

Figure 2.7: Machine Learning VS Deep Learning

Table 2.1 taken from [27] shows the Key comparisons between ML and DL.

Parameter	ML	DL
Data Dependency	Although ML depends on the huge amount of data, it can work with a smaller amount of data.	DL algorithms highly depend on a large amount of data, so we need to feed a large amount of data for good performance.
Execution Time	ML algorithm takes less time to train the model than DL,However, testing the model can be time-consuming and requires a significant duration.	DL takes a long execution time to train the model, but less time to test the model.
Hardware Dependencies	Since ML models do not need much amount of data, so they can work on low-end machines.	The DL model needs a huge amount of data to work efficiently, so they need GPU's and hence the high-end machine.
Feature Engineering	ML models need a step of the interaction with the expert performing feature extraction, after which it continues to progress.	DL is the enhanced version of ML, so it does not need to develop the feature extractor for each problem; the problem-solving approach focuses on allowing the model to learn high-level features directly from the data.
Problem-solving approach	To solve a given problem, the traditional ML model breaks the problem in sub-parts, and after solving each part, produces the final result.	The problem-solving approach of a DL model is unlike tradictional ML models.
Interpretation of result	The ease of interpreting the result for a specific problem is evident. As when we work with ML, we can interpret the result easily, it means why this result occur, what was the process.	The interpretation of the result for a given problem can get very difficult. We may get a better result for a given problem than the ML model, but we cannot find why this particular outcome occurred, and the reasoning.
Type of data	ML models mostly require data in a structured form.	DL models can work with structured and unstructured data both as they rely on the layers of the ANN.
Suitable For	ML models are suitable for solving both simple and moderately complex problems.	DL models are suitable for solving complex problems.

Table 2.1: Key Differences Between Machine Learning and Deep Learning.

7 Digital Twins

7.1 Digital Twins History

The article [28] recorded in 2019 that over 850 academic papers on the topic of Digital Twins have been published since 2016.

The concept of a "twin" has its roots in the National Aeronautics and Space Administration (NASA) Apollo program of the 1970s. During this time, NASA built a replica of space vehicles on Earth that mimicked the equipment's condition during the mission. This was done to ensure that NASA could test and prepare for every possible scenario that might occur during the mission. This was the first application of the "twin" concept [29].

In 2003, Michael Grieves, a professor of engineering at the University of Michigan, proposed the idea of a DT in his Product Life-cycle Management (PLM) course. DT is a virtual digital representation of physical products that can be used to simulate and analyze real-world scenarios in a virtual environment. DT technology enables manufacturers to create a digital copy of a physical product, which can then be used to monitor and predict its performance, optimize its design, and reduce the time and cost of maintenance and repairs.

In 2012, NASA applied DT to integrate high-fidelity simulation with a vehicle's on-board health management system, maintenance history, and fleet data to mirror the life of its flying twin. This allowed NASA to monitor the health and performance of their equipment in real-time, identify potential problems before they occurred, and increase safety and reliability.

The development of the IoT has boosted the manufacturing industry's adoption of DT technology. With the IoT, manufacturers can connect their physical products to the internet and collect data on their performance in real-time. This data can then be used to create a DT of the product, which can be used to monitor and optimize its performance, predict maintenance needs, and improve its design.

Enterprises like Siemens and General Electric (GE) [3], have developed DT platforms for real-time monitoring, inspection, and maintenance. These platforms enable manufacturers to monitor their products in real-time, identify potential problems before they occur, and reduce the time and cost of maintenance and repairs.

In 2017, Tao and Zhang proposed a five-dimensional DT framework to guide the digitalization and intellectualization of the manufacturing industry. The framework provides theoretical guidance for the digitalization and intellectualization of the manufacturing industry and includes five dimensions: physical, cyber, human, virtual, and knowledge.

From 2017 to 2019, Gartner continuously ranked DT among the top 10 technological trends with strategic values. DT is becoming increasingly important in the manufacturing industry, as it enables manufacturers to monitor and optimize their products in real-time, predict maintenance needs, reduce the time and cost of maintenance and repairs, and improve their designs.

Figure 2.8 provides a brief History summary of the DT.

Similar to what was previously stated, since their inception, DT have rapidly evolved and become increasingly popular in various industries such as manufacturing, healthcare, and urban planning, among others, thanks to their

[3] GE is a multinational conglomerate that operates in various industries including aviation, healthcare, renewable energy, and power generation. It was founded in 1892 and is based in Boston.

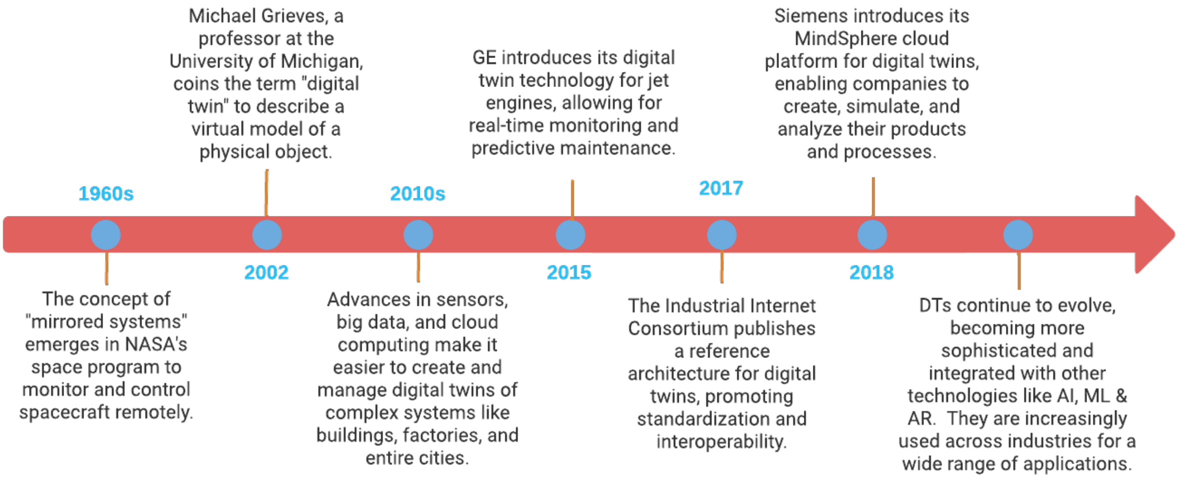

Figure 2.8: Brief History of Digital Twins.

ability to replicate real-world objects, processes, or systems with a high level of accuracy, simulate and test different scenarios, monitor their performance in real-time, and optimize their design and operation, thereby enabling organizations to make more informed decisions, improve their efficiency, reduce costs, and enhance their customer experience, and as technology advances and more data is collected, analyzed, and shared, it is likely that digital twins will continue to play a vital role in shaping the future of many sectors and transforming the way we live, work, and interact with the world around us.

Figure 2.9, taken from the article [28], shows advancement, evolution and development of DT over time.

This new technology is going to be discussed further more in Chapter 3.

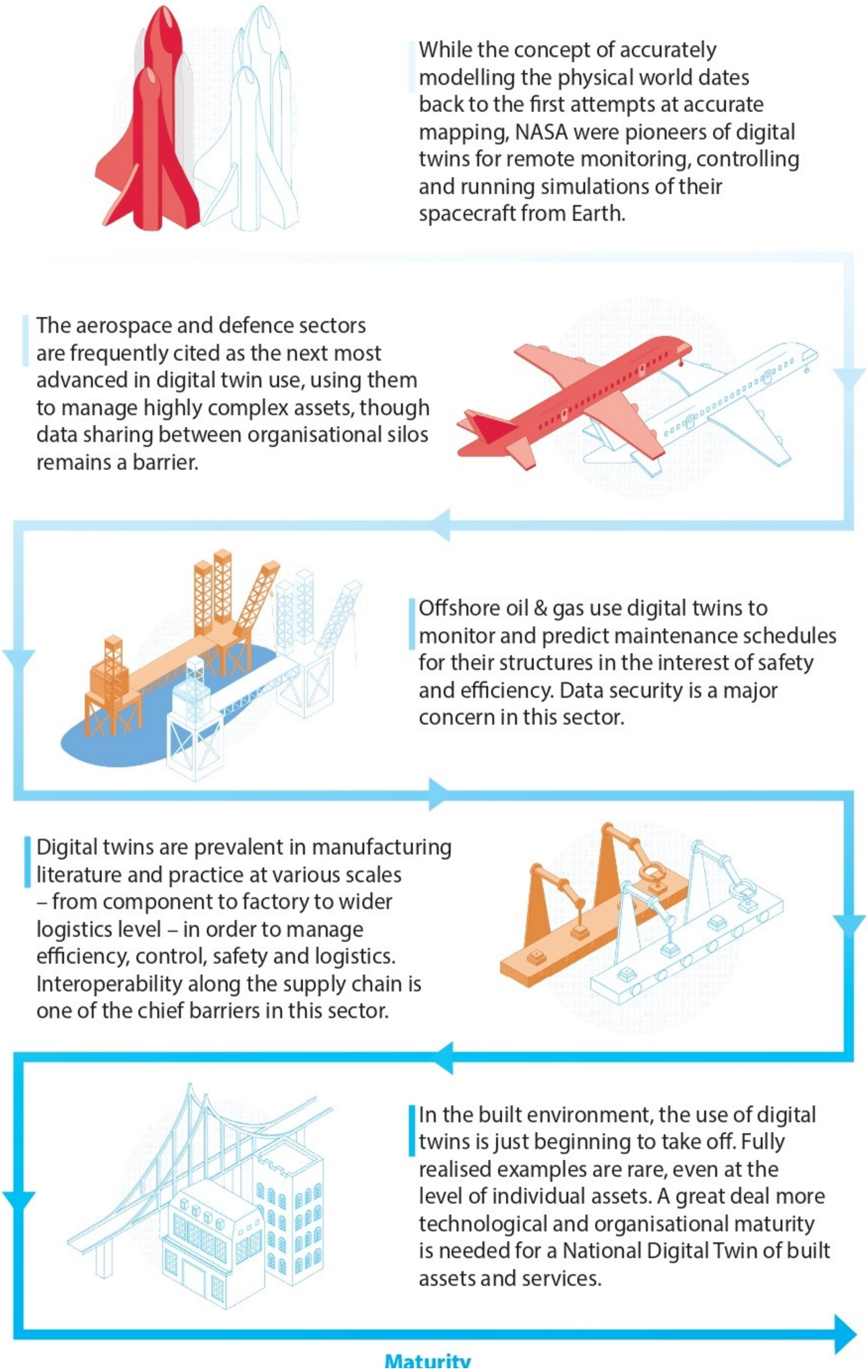

Figure 2.9: Development and spread of Digital Twins over time.

Part III

State of the Art

Chapter 3

State of the Art

Contents

1 Digital Twins Concepts

1.1 Introduction

The concept of digital twins has become increasingly important in the field of IoT in recent years. Essentially, a digital twin is a virtual representation of a physical object or system, created using data gathered from sensors and other sources.

In the context of IoT, digital twins can be used to monitor and manage physical assets and systems, such as buildings, vehicles, and manufacturing equipment, in real-time. By simulating the behavior of the physical object or system, digital twins can help to identify potential issues and optimize performance. In this chapter, a comprehensive exploration of the digital twin concept is provided in more detail in the form of a state of the art that encompasses its general definition, characteristics and architecture.

1.2 Definition of Digital Twins

DT lacks a universally accepted definition and established standards for its implementation, which makes it difficult to design, implement, and adopt this technology widely [30]. Moreover, as DT is used across various domains and relies on evolving technologies, it needs to be tailored for each domain and is subject to the current state of these technologies.

Table 3.1 displays a range of DT definitions along with their corresponding reference and applied fields.

Numerous articles have focused on the absence of a fixed and pre-established

Domain	Definition	
Aerospace	- A DT is an integrated multiphysics, multiscale, probabilistic simulation of an as-built vehicle or a system that uses the state-of-the-art physical models and other relevant information to accurately replicate the life and behavior of its corresponding flying counterpart. The DT is ultra-realistic and may consider one or more important and interdependent vehicle systems. - DT is a life management and certification paradigm whereby models and simulations consist of as-built vehicle state, along with recorded loads, environmental conditions, and specific historical data related to the vehicle, in order to facilitate detailed and precise modeling of individual aerospace vehicles throughout their operational lifespan.	[31] [32]
Industry	DT is an evolving digital profile of the historical and current behavior of a physical object or process that helps optimize business performance. It is based on massive, cumulative, real-time, real-world data measurements across an array of dimensions.	[33]
Engineering	A DT is a digital replica of physical assets, processes, and systems that can be used for various purposes, such as simulation, optimization, and monitoring.	[34]
Healthcare	A DT is a personalized, dynamic, and data-driven computational model that can be used to simulate an individual's physiology and health status, and to predict their response to treatment or changes in lifestyle.	[35]
Agriculture	DT is a dynamic approximation of an entity in virtual space, continuously updated through the collection of data, models, and what-if simulation. In the majority of applications found in current research, agricultural DT form a simplified or functionally reduced view of the observed entity or system, as cost, complexity, and goals are balanced with functionality and replication correctness requirements, as guided by the functional requirements of the intended application.	[36]
Manufacturing	DT are software models that represent the attributes and operating behavior of physical assets and processes. They support better decision making by simulating how assets behave given certain inputs.	[37]

Table 3.1: Diverse Definitions of Digital Twins in Literature

concept for DTs. This gap has been addressed in several articles, which have proposed the following definitions:

- Grieves and Vickers define the DT as a connection of virtual and digital representations that comprehensively depict and describes the existing physical asset, encompassing its molecular composition and overall geometry. When functioning optimally, a Digital Twin provides all the information that would typically be gleaned from examining the physical counterpart. There are two types of Digital Twins: Digital Twin Prototype (DTP) and Digital Twin Instance (DTI) [30].
- "Various terms have been given in multiple literature works, such as- 'ultra-high fidelity', 'cradle-to-grave', 'integrated' model , Integral Digital Mock-Up (IDMU). These terms are important and relevant to the DT concept, however, having multiple definitions and terms has delayed reaching a consensus on a single representative, unifying definition. In the simplest words, a digital twin is a 'digital' 'twin' of an existing physical entity" [30].
- "A DT is the virtual digital representation equivalent to physical products" [38].

1.2.1 Deducing a General Definition of Digital Twins

In the course of exploring the literature on DT, it becomes apparent that many articles have examined the concept of DT within a particular domain, such as manufacturing or healthcare, as a result, a comprehensive and universally applicable definition of DT has been elusive.

However, by synthesizing the information collected from these various sources, we can arrive at a global definition of DT that encompasses the most salient features and characteristics of the concept:

DT refer to a combination of virtual machines and computer-based models that enable the simulation, emulation, or mirroring of the behavior and characteristics of a physical entity, such as an object, a process, a human, or a human-related feature. The relationship between a DT and its Physical Twin (PT) is established through a bijective connection that enables continuous interaction, communication, and synchronization between the two.

Unlike static models or simulations, DT are living, intelligent, and evolving models that follow the life-cycle of their PT to monitor, control, and optimize their processes and functions. DT can predict future statuses, such as defects, damages, or failures, and simulate and test novel configurations to proactively apply maintenance operations.

The twinning process is facilitated by a closed-loop optimization approach that considers the DT, its PT, and the external surrounding environment. This approach ensures that DT are more than just simple models or simulations. They are a dynamic and responsive tool that allows designers, engineers, and operators to enhance the efficiency, safety, and performance of physical systems across various industries.

To aid in visualization,Figure 3.1 is provided.

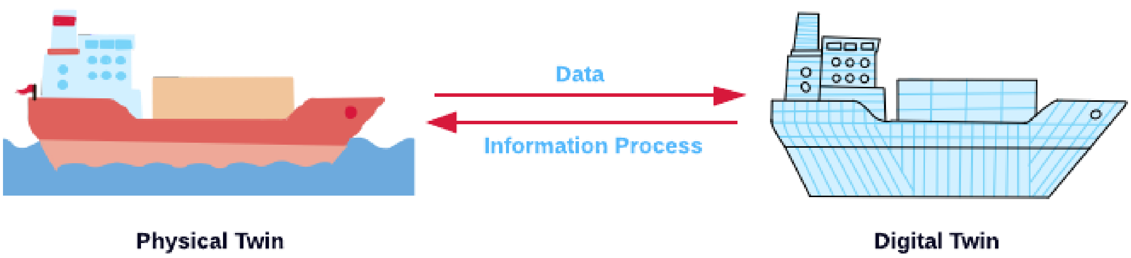

Figure 3.1: Digital Twin's Example Representation

1.3 Components of a Digital Twin

The concept of DT was first introduced by Grieves [39] and defined as having three components: the digital or virtual part, the physical product or asset, and the connection between them.

- On the virtual side, the amount of the information available has improved. Numerous behavioral characteristics have been added so that not only the product can be visualized, but its performance capabilities can be tested as well.
- On the physical side, more information about the characteristics of the physical product can be collected: All types of physical measurements from automated quality control station such as Coordinate Measuring Machine (CMMs) can be collected.

As the concept evolved, other authors such as Tao et al. have extended the definition of a DT to include additional components, such as data and service. Tao et al. also identified Verification, Validation and Accreditation (VVA) as necessary components of a DT. With the introduction of data models, Miller et al. further expanded the definition of a DT to include an integration of multiple models [30].

However, despite these efforts to refine the definition of a DT, reaching a consensus on its fundamental requirements remains challenging. This is because the necessary components and properties of a DT vary across different works, and the domain-dependence of DT calls for a defining the components that can be generalized across domains.

To address this challenge, researchers have collated and integrated the necessary components and properties from previous works to provide a comprehensive definition of a DT. The properties and components considered necessary for a DT are based on their existence in the literature and the researchers' understanding of the DT concept. By integrating the contributions of previous works, which have only been concerned with some components of DT, researchers aim to provide a more holistic definition of a DT.

Based on this analysis and understanding, researchers have defined the elementary and imperative components of a DT. These components provide a comprehensive definition of a DT that can be used across domains. By refining the definition of a DT and its fundamental requirements, researchers aim to provide a framework that can support the development and implementation of DT in a range of applications and industries.

The Table 3.2 taken from [30], summarises how each component contributes uniquely to the functions of DT. Removing any component voids the DT of the functionality and its uniqueness. The three first rows are required, and the rest contribute to the uniqueness of the Digital Twin.

Characteristics	**Definitions**
Physical Asset	What the digital twin is a twin of.
Digital Asset	The Digital Twin
Continuous Bijective Relation	For real-time synchronisation and twinning.
IoT	For data collection and information sharing.
Time Continuous Data	For synchronisation and input to ML.
ML	For analytics of the asset.
Security	To prevent data leaks and information compromises.
Evaluation metrics / Testing	To evaluate the performance of DT.

Table 3.2: The Required and Optional Components of a Digital Twin.

☞ **Note:** In this paper, the considered components are: Physical Asset, Digital Asset, Continuous Bijective Relation, IoT, ML.

1.4 Characteristics and Requirements of a Digital Twin

Although the definition of a digital twin may appear straightforward, it is the properties of the technology that distinguish it as more than just a mere digital replica. Some properties are required to create an accurate and authentic digital twin, while others are dynamic and can evolve over time. This section will explore both types of characteristics in detail.

1.4.1 Essential Characteristics (Requirements)

The Necessary properties and features mentioned in the Article [30] are:

- Real-time connection with the physical entity by making a bi-univocal relation between DT and the physical asset which means that the PT is uniquely paired with its DT.
- Self-evolution is a characteristic that has not been explored much. Seld-evolution means that a DT can learn and adapt in real-time, by providing feedback to both physical asset and DT. This can be easily harnessed now due to the up rise of machine learning tools: to remodel and redesign itself (such as reinforcement learning). The frequency of this synchronisation depends on the update scenarios, such as event-based (supply chain), periodic intervals (aircraft), condition based (logistics), etc.
- Continuous ML analysis (dependent on the frequency of the synchronisation), not just one-time output forecasting.
- Availability of time-series (or time continuous) data for monitoring, and as input to ML system.
- Domain dependence (or Domain specific services): According to the domain, a DT may provide or prioritise services specific to the industry.

These are the same 'domain specific' services which exist in the physical asset (for example the optimisation problem).

- Knowledge Database: it provides the Digital Twin with the Knowledge base required to provide Services. In Order to filter out the specific Knowledge from the huge amounts of Data collected on the Internet – that is to say Big Data (BD) – these Amounts of Data must be analyzed accordingly [40].

Table 3.3 shows the characteristics presented in [41].

1.4.2 Dynamic Characteristics

A hierarchy of digital twins can be established by utilizing these dynamic properties [30].

- **Autonomy:** A DT could either make changes to the physical asset itself, or a human in control could make changes to the DT. This applies differently to different hierarchies of components present in the twin, such as to some parts of the ML system, or some part of the decision making system. Hence, the property of a DT to be autonomous, not autonomous, or partly autonomous.This classification also includes the self-evolution mechanism of DT (what changes must it make to itself, and what changes must be approved by a human).
- **Synchronisation:** Synchronisation of data could either continuously or at certain time intervals. These depend on a number of factors such as technology, resources available, need for the data and type of ML algorithm being used. A DT could have sub-components which could be partly continuously synchronised and partly event-based synchronised. This synchronisation can result in different hierarchies based on the following:

Characteristics	Definitions
Physical Entity/ Physical Twin	The physical entity/twin exists in the external real environment.
Virtual Entity/ Virtual Twin	The virtual entity/twin that exists in the virtual environment.
Physical Environment	The environment within which the PT exists.
Virtual Environment	The environment within which the virtual entity/twin exists.
State	The measured values for all parameters corresponding to the PT, DT and its environment.
Metrology	The act of measuring the state of the physical/virtual entity/twin.
Realisation	The act of changing the state of the physical/virtual entity/twin.
Twinning	The act of synchronising the states of the physical and virtual entity/twin.
Twinning Rate	The rate at which twinning occurs.
Physical-to-Virtual Connection/ Twinning	The data connections/process of measuring the state of the physical entity/twin/environment and realising that state in the virtual entity/twin/environment.
Virtual-to-Physical Connection/ Twinning	The data connections/process of measuring the state of the virtual entity/twin/environment and realising that state in the physical entity/twin/environment.
Physical Processes	The processes within which the physical entity/twin is engaged, and/or the processes acting with or upon the physical entity/twin.
Virtual Processes	The processes within which the virtual entity/twin is engaged, and/or the processes acting with or upon the virtual entity/twin.

Table 3.3: The characteristics of the Digital Twin and their descriptions.

(a) How often the data is collected?

(b) How often the data is stored?

(c) How often the DT is updated?

1.4.3 Key Characteristics Highlighted in this Paper

In this paper, the characteristics taken into consideration are the following:

- Real-time connection with the physical entity.
- ML analysis.
- Domain dependence.
- Knowledge database
- Availability of time-series data.
- Synchronisation.

1.5 An Overview on the Predecessors of Digital Twins and Their Key Differences

The process of generating virtual representations of physical objects, facilities, or processes results in the creation of virtual models that belong to a virtual space. These models are essentially computer-generated replicas of their real-world counterparts and exist within a digital environment. In this subsection, a distinction is made between the several types of Digital Models.

These Models differ primarily in how Data Flows between an original in Physical Space and its Model in Virtual Space. As can be seen in Figure 3.2, the Organization of the Data Flow in these Models is either manual and/or automatic. These three Types of Digital Models are presented below.

1.5.1 Digital Model / Digital Simulation Model

In [30], the author talked about the flow of data of a DT by mentioning that it has only manual exchange of data and that it does not showcase the real-time state of the model.

Similarly, the author of [40] defines the purpose of a Digital Simulation Model: it is to replicate a system with its dynamic internal Processes in order to obtain Knowledge that can be transferred to the original Physical System.

The Simulation is mainly realized with the Support of Computers using an experimental Digital Model. This is typically carried out spontaneously and only at certain Times. In doing so, often only those Features of the original System are modeled that are of Importance for specific Problems to be solved.

As already mentioned above, the special Feature of the Use of Digital Simulation Models is that the Data between the Physical Original System and the Simulation Model is not transferred directly (automatically) in both directions – but indirectly – and often manually.

1.5.2 Digital Shadow

Digital Shadow is a saved data copy of the physical state [30], it sums all the data that is left behind every time a digital service is used, such as the Internet or a mobile phone. It is a collection of data traces put together for a specific purpose and can include measured parameters as well as historical data [42].

It has a one way data flow from physical object to the digital object [30].

In the industrial sector, digital shadows represent virtual copies that are created to interact with other people and environments. It is possible to make digital shadows of digital twins because they can capture and simplify the multitude of information that they generate.

1.5.3 Digital Twin

The digital twin on the other hand, has fully integrated data flow where the digital twin properly reflects the actual state of the physical object.

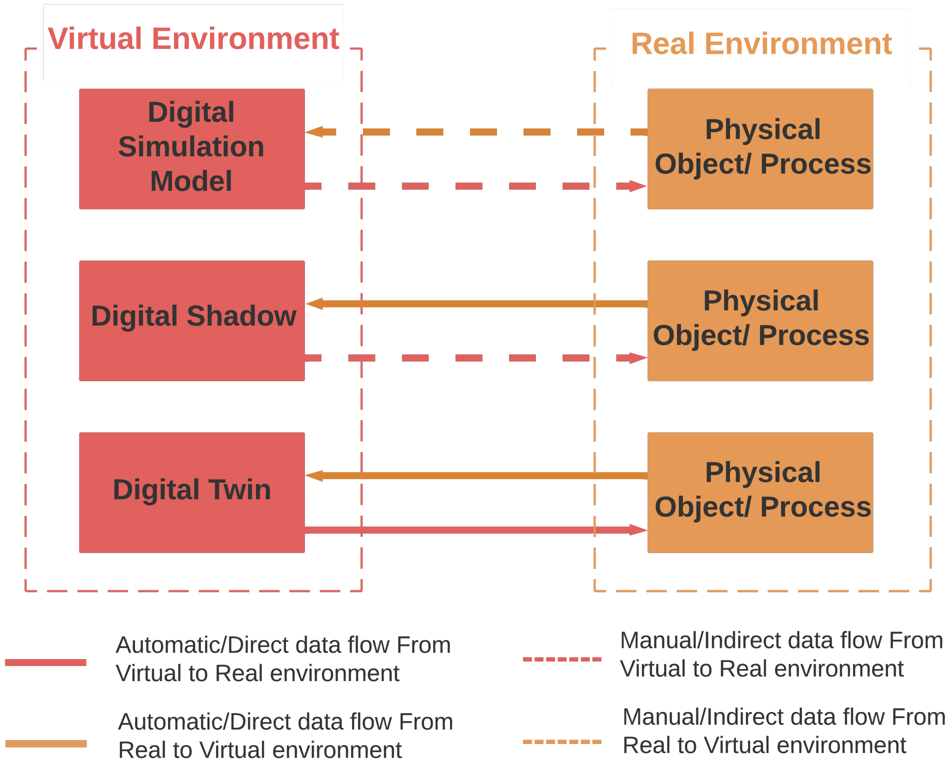

Figure 3.2: Digital Simulation VS Digital Shadow VS Digital Twin

1.5.4 Digital Model VS Digital Shadow VS Digital Twin

After defining each concept individually, a comparative analysis can be made.

Digital model, digital shadow, and digital twin are related concepts but have distinct differences.

A digital model is a computerized, data model of a building, product, or some other object that describes the form of an existing or proposed object.

A digital shadow represents virtual copies that we create to interact with other people and environments. In the industrial sector, it is used to monitor and optimize the performance of physical assets.

A digital twin is a virtual replica of a physical asset that is used to simulate, predict, and optimize the performance of the asset. It emphasizes the bi-directional approach, where the information flow not only from digital assets to the physical world but also loops back from the physical world to the digital world [43] [44] [45] [46] [47].

1.6 Different Types of Digital Twins

There are four distinct types of digital twin technology [48] [49], each with its own characteristics and benefits. These types include component, asset, system, and process twins. In this subsection, each of these types are going to be seen in more detail.

To assist with visualization, Figure 3.3 is provided, which showcases an example of each type of DT.

1.6.1 Components Twins

Digital models of individual components or parts, such as motors, sensors, switches, and valves, are known as component twins. These twins are the basic unit of a DT and the smallest example of a functioning component. They offer detailed information regarding a component's behavior and performance in real-time as well as over time. This enables organizations to monitor the performance and health of these components and make necessary changes whenever required.

1.6.2 Asset Twins

Digital models of physical assets and when two or more components work together, such as buildings, machines, and vehicles, are referred to as asset twins. These twins provide real-time information about the operational status, performance data and environmental conditions of an asset. As a result, organizations can minimize downtime and enhance the efficiency of their operations.

1.6.3 System Twins/Unit Twins

The next level of magnification involves system or unit twins, which enables to detect different assets connected to form a whole functioning system.

These twins facilitate the monitoring and analysis of a system's performance, helping organizations to pinpoint areas that require improvement. System twins enable organizations to optimize their processes and enhance their operational efficiency. They provide visibility regarding the interaction of assets, and may suggest performance enhancements.

1.6.4 Process Twins

Digital models of entire business processes or customer journeys are referred to as process twins. It is the macro level of magnification. They furnish comprehensive information on how customers interact with an organization's products and services in real-time, assisting organizations in identifying areas where customer experience can be enhanced. They reveal how systems work together to create an entire production facility. Process twins can help determine the perfect timing schemes that ultimately influence overall effectiveness.

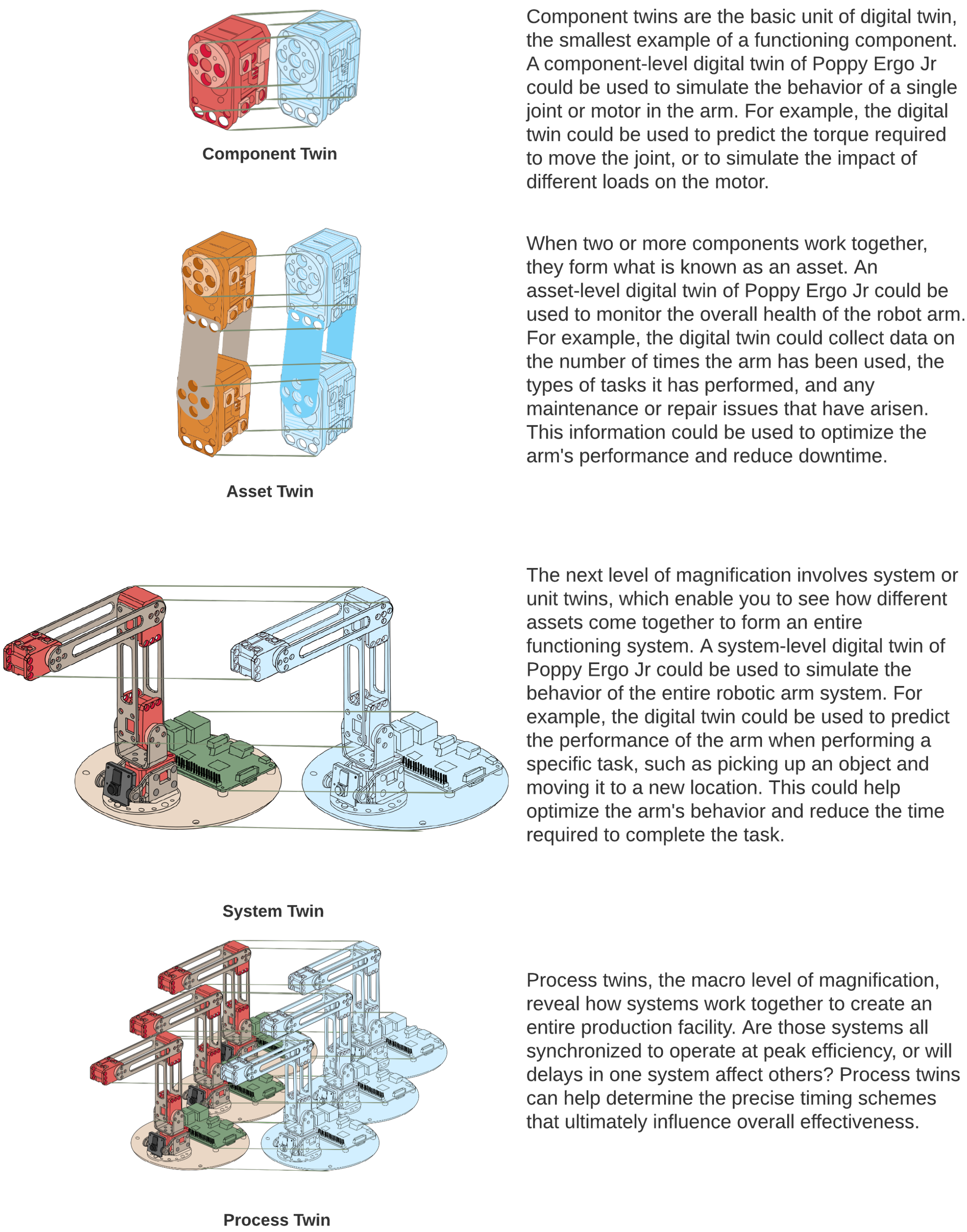

Figure 3.3: Digital Twin's types Example Robot Poppy Ergo Jr

In [28], two different types of a DT have been given:

(a) **A Dynamic DT** fed by live data flows from a physical asset, for example a building, or one of its components, like a lift motor. Insights and programmed instructions from the digital twin can then impact the physical twin using real-time control mechanisms, for example shutting down a faulty lift or adjusting the temperature of a room.

(b) **A Static DT** that changes periodically as long-term data about a physical asset are added in. This type of digital twin is used for strategic planning, and feedback into the physical twin is achieved through the capital investment process.

2 Digital Twins with ML and DL

2.1 Introduction

ML and DL are an important aspect of DT technology, as they can be used to predict and analyze data in order to improve decision-making and optimize performance. There are several studies that explore the integration of ML and DL in DT technology, including the use of DL for decision support [50].

It is used in DT as well to create smart machines and plants whereby the inputs from sensors are analyzed in real-time.DT integrate IoT, AI and ML with software analytics to create digital living. The purpose of integrating DL and DT is to improve the accuracy of the DT model and to reduce the time and cost of the modeling process. ML provides important real-time insights that enhance situational awareness and enable fast, effective responses. It often can predict the future behavior of the system and provide recommendations for optimizing the system's performance [51] [52] [53].

2.2 ML Appliance in Digital Twins

As it is mentioned in [53], there are two widely used Data Science (specifically ML) areas used in DT that are explained down bellow and has been summarized in Table 3.4.

2.2.1 Diagnostic and Predictive Analytics:

The field of IoT has brought about significant advancements in the realm of smart machines and plants. With the ability to connect a vast network of devices, IoT enables the seamless exchange of data and information between interconnected devices, systems, and humans.

As stated in [53], by integrating ML algorithms with IoT, intelligent systems that analyze and understand vast amounts of data in real-time can be created. These systems can then use this data to diagnose potential problems and predict future behaviors of the system.

The Twin is one such intelligent system that uses IoT and ML algorithms to analyze and understand inputs from various sensors in real-time. The Twin is essentially a virtual replica of the physical system, and it continually updates itself based on the data received from the sensors.

Using advanced ML algorithms, the Twin can learn from historical data and use this information to make predictions about the future behavior of the system. This ability to predict future behaviors can help prevent failures and other problems before they occur, saving time, money, and potentially even lives.

The Twin can also diagnose the causes of problems by analyzing sensor data in real-time. By identifying patterns and anomalies in the data, the Twin can quickly determine the root cause of the issue and suggest potential solutions.

In summary, IoT-based ML models, such as the Twin, are revolutionizing the way a complex system is designed and maintained.

By enabling real-time analysis and understanding of sensor data, these models can help prevent problems before they occur, improving efficiency and reducing downtime.

2.2.2 Prescriptive Analytics:

Prescriptive Analytics is a field of data science that involves using advanced mathematical and computational techniques to identify optimal or feasible solutions to complex problems. Specifically, prescriptive analytics involves simulating an entire network of interconnected systems to identify the best possible solution from a very large set of candidate solutions, given a set of variables and constraints that must be adhered to.

The primary objective of prescriptive analytics is to maximize stated business goals, such as throughput, utilization, output, and other key performance indicators. This can involve creating schedules for resources such as vehicles, personnel, and machines, to ensure maximum efficiency and productivity.

In practice, prescriptive analytics is widely used in supply chain planning and scheduling. For example, a logistics provider might use prescriptive analytics to create a schedule for its resources to ensure on-time delivery, while a manufacturer might use the technique to optimize the utilization of machines and operators to achieve maximum on-time, in-full deliveries.

To solve these complex decision-driven problems, prescriptive analytics relies on a technique called Constrained Mathematical Optimization. This involves formulating mathematical models that take into account all of the variables and constraints that must be considered in order to arrive at an optimal or feasible solution.

Powerful solvers are then used to solve these complex mathematical models, often involving millions of variables and constraints, to arrive at the best possible solution. This approach is highly effective at solving complex problems that would be too difficult or time-consuming to solve manually, and can help organizations make better decisions and achieve their stated business goals more efficiently.

To summarize, ML models predict likely outcomes for a given set of input features based on history, and Optimization models helps you decide that should a predicted outcome(s) happen.

☞ **Note:** This study focuses on the first point explained in Section 2.2.1.

Diagnostic and Predictive Analytics	Prescriptive Analytics
Given a range of inputs, the Twin should be able to diagnose the causes or predict the future behavior of the system. IoT based machine learning models is what is used to create smart machines and plants whereby the inputs from sensors are analyzed in real time to diagnose, predict and thereby prevent future problems and failures before they occur.	This is where an entire network is simulated to identify an optimal or feasible solution from a very large set of candidates, given a set of variables and constraints to be adhered to, usually with the objective of maximizing stated business goals.

Table 3.4: Diagnostic and Predictive Analytics VS Prescriptive Analytics

2.3 Selecting an Adapted Model for IoT tabular Data

When selecting an adapted model for IoT tabular data, there are several factors to take into consideration. A few of those key considerations are mentioned down below:

- **Data Type:**

  IoT devices generate various different types of data, among them, structured data, unstructured data, time-series data, etc. The chosen model should be capable of handling the specific type of data generated by the IoT devices so that it can give a good result.

- **Complexity:**

  IoT data can be complex and difficult to analyze so the selected model should be able to handle the complexity of the data and provide accurate results.

- **Scale:**

  IoT devices generate a large volume of data, often in real-time. The model that would be chosen should be capable of processing large amounts of data quickly and efficiently.

- **Security:**

  IoT data can be sensitive and confidential. The model must have robust security features to protect the data from unauthorized access.

- **Integration:**

  The model must be compatible with the existing technology stack and able to integrate with other systems and applications in the targeted organization.

- **Deployment:**

  The deployment options for the model should be taking into consideration, including cloud-based, on-premises, or hybrid solutions, depending on the organization's needs.

Some popular models for IoT data analysis include ML algorithms, DL Neural Network (NN) and statistical models. It's important to evaluate different models and their capabilities as it has been done in Chapter 1, Section 4.3.

Before selecting the model that suits best the IoT data, few points needs to be specified especially concerning the data type since it would be the input of the future model. Down below are the characteristics of the selected use case that would be presented in the engineering report:

- The datatype is time series data.
- the Machine learning problem is a classification problem.
- The selected model needs to handle scalability and efficiency, since as mentioned previously, DTs can handle huge data coming from different data sources and large datasets with high-dimensional features efficiently.
- Real-time prediction.

The models that suits more these descriptions are:

- Recurrent Neural Networks.
- Decision Trees.

An article that used RNN in presented in the next subsection.

2.3.1 Design and development of RNN anomaly detection model for IoT networks

2.3.1.1 Description

The contributions of the mentioned paper [26] is to:

- Design of an anomaly detection model for IoT networks using a RNN.
- Design of an anomaly detection model for IoT networks using CNN and RNN.
- A lightweight anomaly detection model for IoT networks using a RNN.
- Performance improvements of multiclass and binary classification models.

The focus is established on the proposed model. But first, the stages of an LSTM are viewed in details.

List of symbols:

$$x_t \text{ : } \textit{Input.}$$

$$h_t \text{ : } \textit{New hidden state.}$$

$$h_{t-1} \text{ : } \textit{Previous hidden state.}$$

$$C_{t-1} \text{ : } \textit{Previous cell state.}$$

$$\tilde{C}_t \text{ : } \textit{Current cell state (Candidate).}$$

$$C_t \text{ : } \textit{New cell state.}$$

$$f_t \text{ : } \textit{Forgot gate.}$$

$$i_t \text{ : } \textit{Input gate.}$$

$$(x) \text{ : } \textit{Sigmoid function.}$$

$$\tanh(x) \text{ : } \textit{Tanh function.}$$

$$W_x \text{ : } \textit{Gate weight.}$$

$$b_x \text{ : } \textit{Gate biases.}$$

(a) **Phase 1:** The initial stage of the procedure involves the implementation of the forget gate, where the determination is made regarding the relevance of specific segments within the cell state. In other words, the focus of this step is on identifying the information that should be disregarded from the cell state.

This assessment is based on the combination of the preceding hidden state and the fresh input data. And the mentioned determination is carried out by a sigmoid layer referred to as the "forget gate layer".

Through the utilization of the sigmoid activation shown in the left side of Figure 3.4, the network analyzes the values in h_{t-1} (previous hidden state) and x_t (new input data) to produce a vector where each element falls within the range of [0, 1] in the cell state C_{t-1} A value of 1 indicates complete retention, while a value of 0 signifies complete discarding [26] [54].

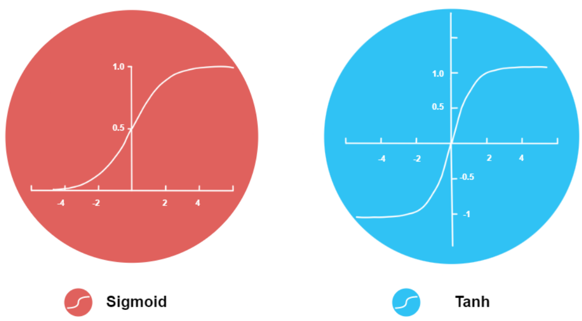

Figure 3.4: Sigmoid and Tanh Functions

The operation of the forget gate layer, which is depicted in Figure 3.5, is captured by Equation (3.1).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3.1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3.2}$$

$$\widetilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3.3}$$

$$C_t = f_t \times C_{t-1} + i_t \times \widetilde{C}_t \tag{3.4}$$

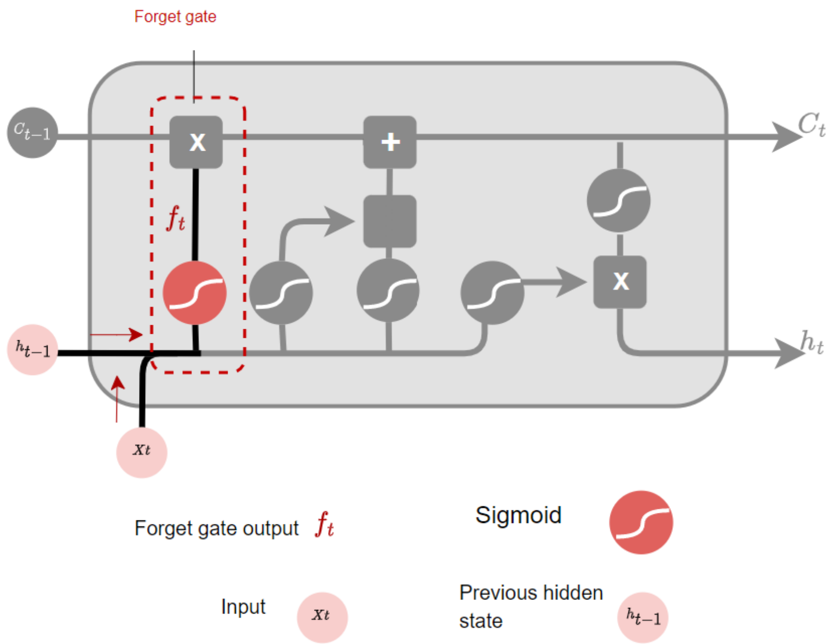

Figure 3.5: LSTM Forget Layer Operation

After generating the output values, they are multiplied element-wise with the previous cell state. This pointwise multiplication serves to diminish the impact of the cell state components that are considered irrelevant by the forget gate network. Those components receive a value close to 0, resulting in reduced influence on subsequent steps [54].

In summary, the forget gate determines which aspects of the long-term memory should be disregarded (given less weight) based on the prior hidden state and the latest data point in the sequence.

(b) **Phase 2:**

In this next step, the memory network and input gate come into play. The objective of this stage is to identify the pertinent information to be incorporated into the long-term memory (cell state) of the network, considering the preceding hidden state (h_{t-1}) and the fresh input data (x_t).

The New Memory Network:

it is a tanh activated neural network which has learned how to combine the previous hidden state and new input data to generate a 'new memory update vector'. This vector essentially contains information from the new input data given the context from the previous hidden state. This vector tells us how much to update each component of the long-term memory (cell state) of the network given the new data [54].

The tanh function has been used in this context because its output values range from -1 to 1, allowing for the inclusion of negative values. The inclusion of negative values is crucial for the intent of diminishing the influence of a component in the cell state.

Input Gate:

In the first part mentioned above, which involves generating the new memory vector, a significant issue arises. It fails to assess whether the new input data holds any significance worth remembering. This is where the input gate comes in.

The input gate operates as a filter, employing a sigmoid-activated network to identify the components of the "new memory vector" that are worth retaining. By producing a vector of values ranging from 0 to 1 (due to the sigmoid activation), the input gate functions as a filter through pointwise multiplication. Similar to our observations with the forget gate, an output value close to zero indicates that the corresponding element of the cell state should not be updated.

Output: The outputs from the first and second parts are multiplied element-wise. This operation ensures that the magnitude of the newly chosen information determined in the second part is regulated and set to 0 if necessary.

The resulting combined vector is then added to the cell state, effectively updating the network's long-term memory [54].

The operation of the Input gate layer, which is depicted in Figure 3.6, is captured by Equation (3.2) and Equation (3.3).

(c) **Phase 3:**

In LSTM networks, the cell state refers to the memory component that carries information throughout the sequence. It serves as a form of long-term memory that allows the network to retain information over longer periods, mitigating the vanishing gradient problem.

The cell state acts as an information highway, enabling the LSTM to preserve relevant information and discard irrelevant information over time. It runs parallel to the hidden state and undergoes a series of operations such as addition, multiplication, and modulation through gates (input

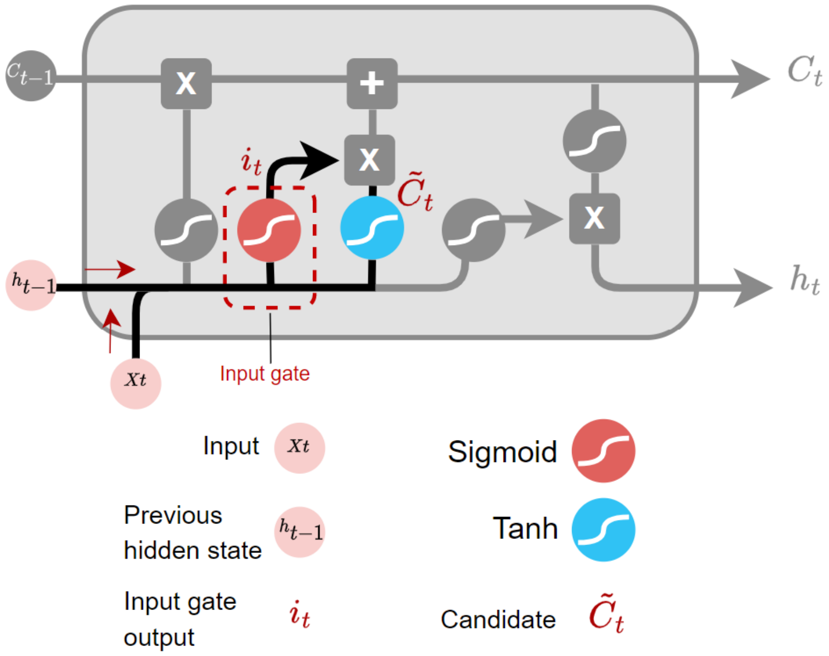

Figure 3.6: LSTM Input Gate Layer Operation

gate, forget gate, and output gate) to regulate the flow of information.

The cell state serves as the primary component that captures the network's memory and plays a crucial role in retaining and updating information throughout the sequence processing in LSTM networks. It is presented in Figure 3.7 and is captured by Equation (3.4).

(d) **Phase 4:**

In order to ensure that only essential information is outputted and saved to the new hidden state, we apply a filter to the updated cell state. However, before applying the filter, we subject the cell state to a tanh function, which confines the values within the range of [-1, 1].

Here is the step-by-step process for this final step [54]:

- The current cell state is pointwise transformed using the tanh function, resulting in the squished cell state that now resides within the

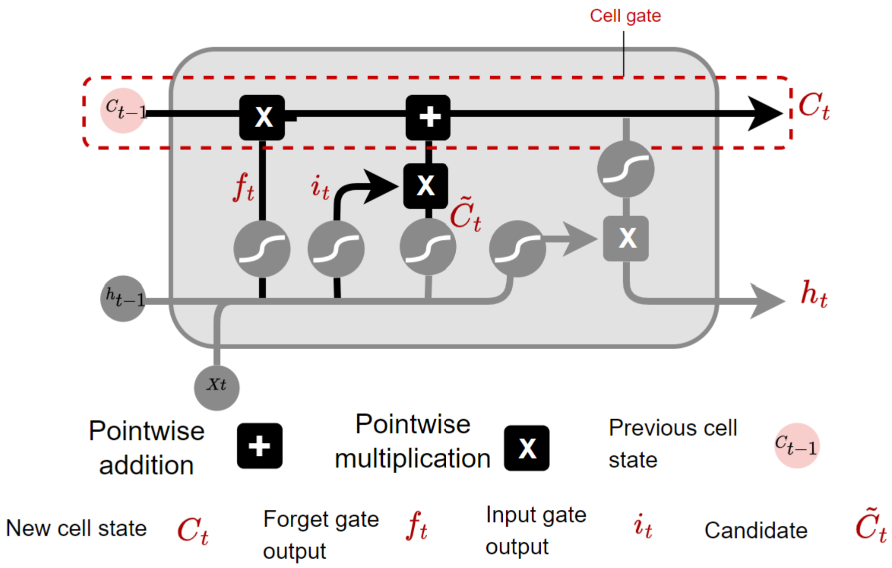

Figure 3.7: LSTM Cell State Operation

interval of [-1, 1].

- Both the previous hidden state and the current input data are passed through a sigmoid-activated neural network, generating the filter vector.
- The squished cell state is then multiplied pointwise with the filter vector obtained from the previous step.
- The resulting output becomes the new hidden state.

This process ensures that the outputted hidden state only contains pertinent information by applying the filter derived from the sigmoid network to the transformed cell state.

This step is presented in Figure 3.8.

Why RNNs instead of another Model?

The concerned article mentioned that DL techniques gained popularity due to their ability to detect computer network threats and abnormalities in various applications and that an RNN model has shown to be effective in multiple

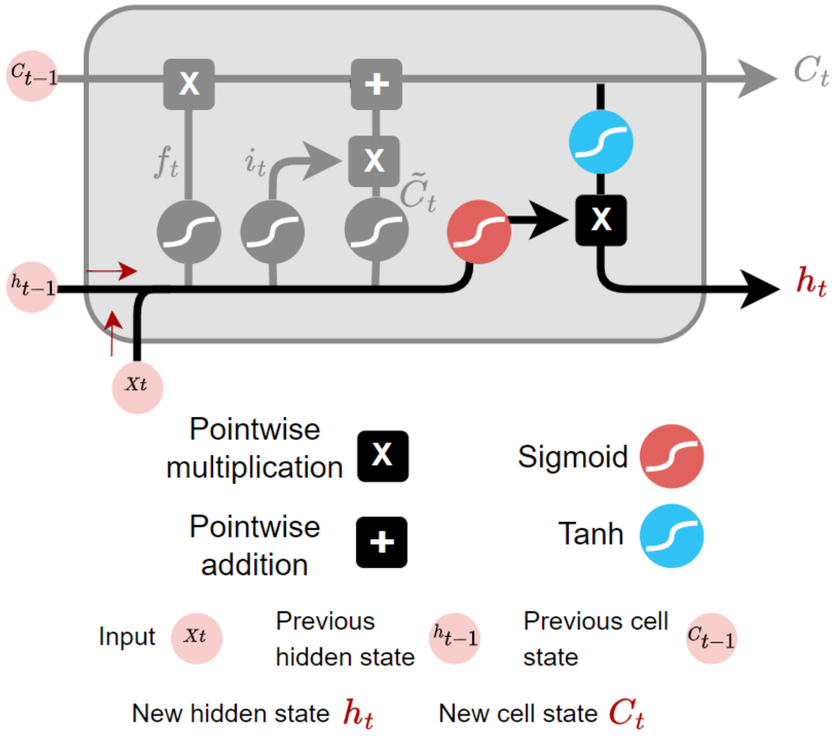

Figure 3.8: LSTM Output Gate Operation

areas due to its better capability, so their realised model consists of an input layer, output layer, and four recurrent, activation, normalization, activity regularization and dropout layers.

However, there are frequent reports and articles stating that Tree-Based Models tend to achieve superior performance compared to Neural Networks.

2.3.2 Why do tree-based models still outperform deep learning on tabular data?

In this article [55], 45 tabular datasets has been used to perform a comparison between various models. Those datasets has been selected depending on different characteristics and differs on:

- Heterogeneous data.
- Real-world data.

- Not deterministic.

The selected models are :

- Scikit Learn's RandomForest.
- GradientBoostingTrees (GBTs) (or HistGradientBoostingTrees when using categorical features).
- XGBoost.
- MLP.
- Resnet.

Figure 3.9 represents the results on medium-sized datasets with only numerical features. Dotted lines correspond to the score of the default hyperparameters. Each value corresponds to the test score of the best model (on the validation set) after a specific number of random search iterations, averaged on 15 shuffles of the random search order. The ribbon corresponds to the minimum and maximum scores on these 15 shuffles.

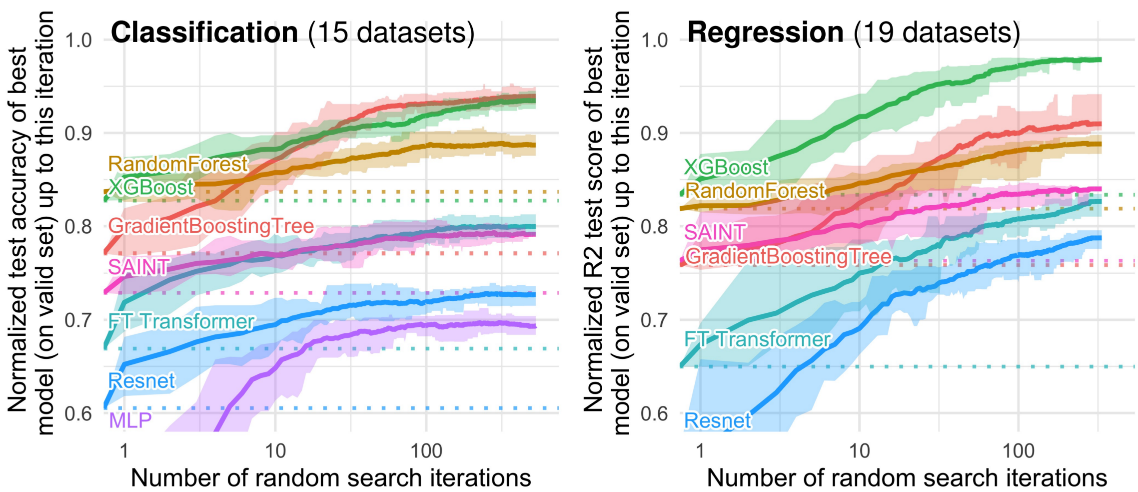

Figure 3.9: Results on medium-sized datasets with only numerical features

And Figure 3.10 represents results on medium-sized datasets, with both numerical and categorical features.

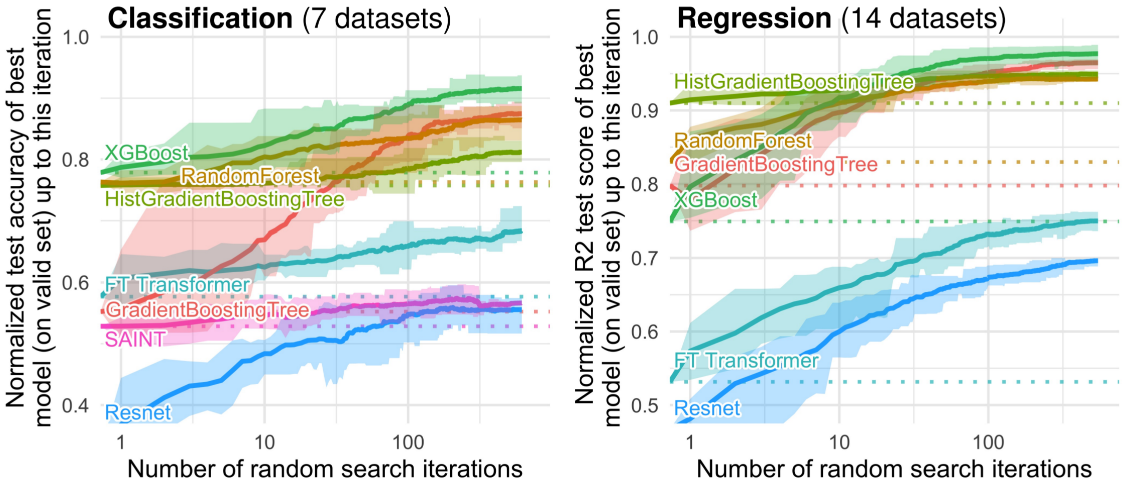

Figure 3.10: Results on medium-sized datasets, with both numerical and categorical features

And it has been proven that tuning the hyperparameters does not make the NNs perform better than tree-based model.

3 The use of Digital Twins for Resilience and Prevention

3.1 Digital twins as run-time predictive models for the resilience of cyber-physical systems: a conceptual framework

The objective of the paper [56] is to propose a new approach for enhancing the resilience of Cyber-Physical Systems (CPSs) by using DT as run-time predictive models.

CPSs are complex systems that integrate physical components with digital technologies and are increasingly being used in safety-critical applications such as transportation, healthcare, and energy systems. However, these systems are also vulnerable to disruptions and failures that can have serious consequences, including safety risks, financial losses, and damage to reputation.

Just like in IoT systems, which is our study-case, disturbance, anomalies and interference is a major problem that can happen in time-series data in every Information Technology (IT) and IoT systems. With the big amounts of time series data produced every day by multiple sensors, the human is no longer able to perform the anomaly detection task manually.

Therefore, developing an approach that can enhance the resilience of CPSs and enable them to adapt and recover from disruptions has been developed is the objective of the referenced article.

Existing approaches to CPSs resilience often focus on reactive measures such as a detection system and recovery. However, these measures may not be sufficient to address the increasing complexity and unpredictability of CPSs. The authors argue that DT can provide a proactive, predictive approach to enhancing CPSs resilience.

They can predict potential failures and recommend actions to prevent them, thus enabling CPSs to anticipate and respond to disruptions more effectively.

Thoroughly, the objective of the paper is to propose a conceptual framework for using DT as run-time predictive models to enhance the resilience of CPSs. The authors aim to demonstrate that this approach can significantly improve the performance, safety, and reliability of CPSs, and can reduce downtime and maintenance costs. The paper also aims to contribute to the field of CPSs by highlighting the potential of DT as a tool for enhancing resilience and providing a framework for further research into their use in CPSs applications.

To adapt this paper to our specific problematic which is enhancing the resilience of IoT systems by using DT , here is what can be extracted :

- The paper highlights the importance of resilience in the context of CPSs and argues that resilience is not just about recovering from disruptions, but also about adapting to changing conditions and mitigating the impact of disruptions.

  This is particularly relevant for IoT systems, which are often subject to a wide range of potential disruptions, such as network outages, cyber-attacks, and environmental factors. By understanding the importance of resilience, An effective approach to enhancing the resilience of an IoT system can be developed.

- The paper proposes a conceptual framework for creating DT of CPSs and using them as run-time predictive models. This approach can be applied to IoT systems as well.

  DT can help to predict and prevent disruptions in IoT systems, and can provide a tool for testing and optimizing these systems in a virtual environment. By considering the use of DT in an IoT system, its resilience and performance improves.

- It has been suggested to explore the potential of DT for enhancing the social and environmental sustainability of CPSs. This is equally relevant for IoT systems, where sustainability is an increasingly important concern. For example, the DT may be used to optimize energy usage, reduce waste, or improve the environmental impact of the IoT system.

3.2 Cognitive Digital Twins for Resilience in Production: A Conceptual Framework

Similar to [56], The objective of this article [57] is to propose a framework to enhance the resilience, but instead of CPS, it is for production systems using cognitive DT.

Why production systems? Because they have become more intricate and interdependent in recent times, making them susceptible to a range of disruptions and uncertainties. To tackle these challenges, experts and researchers are exploring fresh approaches to boost the resilience of production systems. One such approach involves using cognitive DT to improve the system's ability to withstand disruptions and uncertainties.

The article aims to explore the concept of cognitive DT, which are DT that incorporate AI and ML to enhance their capabilities. These DT can provide real-time feedback to operators, predict potential issues before they occur, and optimize production processes.

Here is what this paper discussed and tried to attain as objectives:

- Discuss how cognitive DT can be used to improve production processes and increase resilience. For example, DT can help identify potential issues in the production process and provide recommendations for addressing them, reducing the risk of disruptions.
- Explain how DT can help optimize the production process by simulating different scenarios and identifying the most efficient production methods.
- Elucidate how Cognitive Digital Twin (CDT) presents several challenges, among them data privacy concerns, the need for significant computing power, and the complexity of integrating DT into existing production systems.

To give a solution to these problems and challenges, the goal is to provide a roadmap for the development and deployment of CDT in production systems. The roadmap includes several steps, such as identifying the Key Performance Indicators (KPIs) that the DT will monitor, selecting the appropriate AI and ML algorithms, and developing a data management strategy.

- Provide a framework for using CDT to enhance the resilience of production systems. By leveraging the power of AI and ML, CDT can help production systems adapt to changing conditions, reduce the risk of disruptions, and improve overall efficiency and productivity.

3.3 State of the Art in using Digital Twins for prevention

- In [58], Koen Bruynseels, Filippo Santoni de Sio and Jeroen van den Hoven used Digital Twins in healthcare to reflet the current state of physical objects by redefining 'normality' and 'health' based on individual patterns compared to population patterns, impacting the distinction between therapy and enhancement. The concept of Digital Twins is a valuable tool for analyzing the ethical and conceptual aspects of future healthcare and human enhancement by utilizing individualized data on molecular makeup, physiology, lifestyle, and diet. Comparing Digital Twins across populations helps differentiate between health and disease, shaping the therapy-enhancement debate. Digital Twins have the potential to identify effective routes for therapy and enhancement, allowing individuals to define their well-being preferences. However, ethical, legal, and social concerns arise, including challenges to equality and the risk of discrimination based on compiled information. Governance is necessary to ensure transparency, data privacy, and fair access to this data-intensive technology.

- In [59], the authors presented the benefits of using digital twins in manufacturing Six core cognitive capabilities (perception, attention, memory, reasoning, problem-solving, and learning) were described along with their ability to influence complex manufacturing decisions and future autonomy.

- The research paper [60] presents a novel framework for anomaly detection in digital twin-based Cyber-Physical Systems (CPS). The framework includes two main components: a discrepancy detector based on the Gaussian Mixture Model (GMM), and an anomaly classifier utilizing the Hidden Markov Model (HMM).

  Initially, the discrepancy detector analyzes data from two sources: one from the physical plant and the other from the digital twin. It assesses if there are any anomalies present by comparing the data from both sources. The generated signatures from this detector are then used by the anomaly classifier to classify different types of anomalies, employing the HMM.

  To validate the effectiveness of the framework, experiments were conducted using the Tennessee Eastman process model.

  In future endeavors, the researchers aim to enhance the framework by integrating correction mechanisms. These mechanisms would be designed to maintain system stability based on the classification results obtained from the anomaly classifier.

- In the paper [61], a pioneering approach is introduced for constructing a dynamic digital replica, or digital twin, of an additive manufacturing system utilizing retrofitted low-end sensors found in IoT devices. By leveraging side-channels like acoustic, vibration, magnetic, and power signals, the system can be indirectly monitored. These signals are then processed using a clustering algorithm to generate a comprehensive fingerprint library that accurately represents the physical state of the system, essentially creating a physical twin in the digital realm. The digital twin serves the purpose of detecting and pinpointing anomalous physical emissions that may lead to variations in product quality.

With an average accuracy of 83.09%, the digital twin successfully localizes errors by comparing the detected emissions to the established fingerprint library. Furthermore, an algorithm is presented for updating the digital twin and deducing any deviations in quality. To illustrate the effectiveness of the methodology, a case study is conducted using an additive manufacturing system.

In comparison to existing methods that disregard the liveliness of the model, their created approach outperforms them by dynamically updating itself, accurately inferring quality deviations, and precisely localizing abnormal faults within the additive manufacturing system.

4 Digital Twins architecture

The objective of the article "Towards a Requirement-driven Digital Twin Architecture", as it is mentioned in its title, is to propose a new architecture for DT that is driven by requirements. Since DT can be used to simulate, predict, and optimize the behavior of the physical systems in real-time, its development and realization of an architecture independently of the use case can be challenging especially due to the need for accurate data, modeling, and simulation.

To address these challenges, the authors propose a requirement-driven approach to the design of DT architectures. This approach emphasizes the importance of understanding and defining the requirements of the physical system before developing the DT. The authors suggest that a set of requirements can serve as the basis for the DT architecture, and that this architecture can be designed to meet these requirements.

The provided comprehensive and practical approach to the development of DT architectures that can be used to support a range of applications and industries will be presented in the next subsection.

As mentioned, The paper proposes a conceptual framework for the development and deployment of CDT in production systems. This framework provides a systematic approach for integrating DT technology into production systems and can help practitioners and researchers to implement DT in a systematic and effective manner.

This conceptual framework is presented in Figure 3.11.

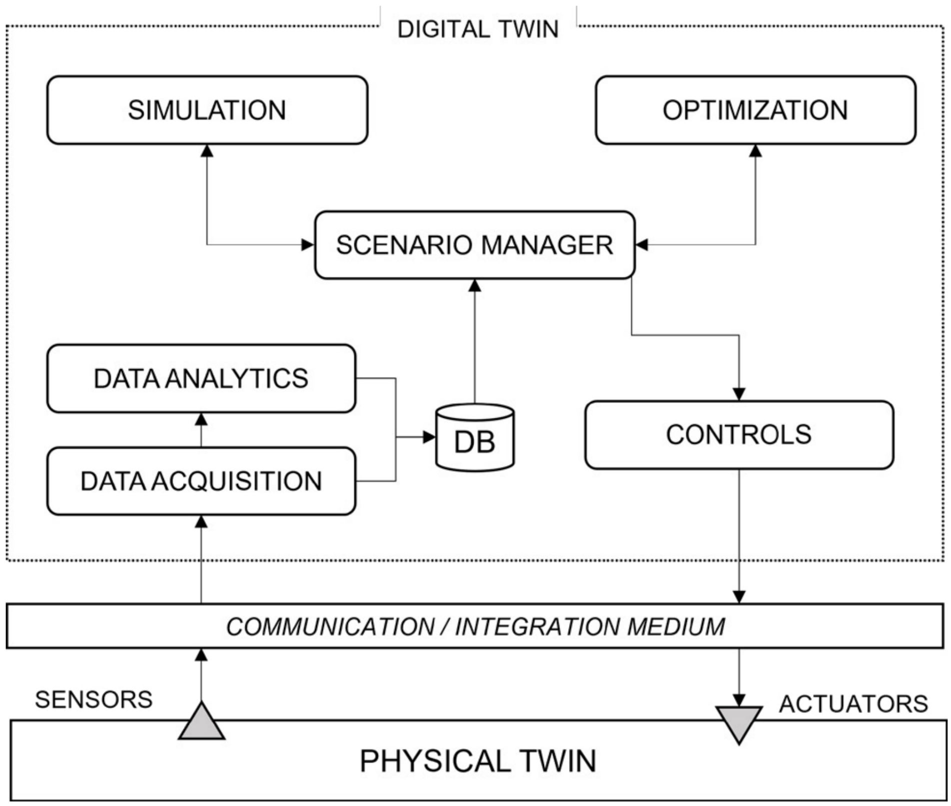

Figure 3.11: Exemplary DT architecture.

5 Framework CSDT

Taking into consideration the following mentioned points:

- Level of granularity: System Twin.
- Use AI for Diagnostic and Predictive Analytics.
- The creation of a static DT.

By remaining faithful faithful to the above points, a CSDT can be created, this term that is created to represent the framework that can not only replicate the PT actions, what makes it a super-DT is its ability to generate disturbed data and functions to make it more resilient to future problems rather waiting for it to occur.

5.1 Conclusion

In conclusion, this state-of-the-art analysis and different reviews on the present articles has shed light on the power Digital Twins and Cognitive Digital Twins. Through an exploration of definitions, current research, and practical applications, it is evident that Digital Twins offer significant benefits for resilience in various domains.

Further experimental work will be emphasizing on :

- Create an Edge/Fog System of Sytems Architecture adapted to the problem and use-case presented.
- Use RNN for a classification problem.
- Prove that Decision Trees perfom better o tabular data.
- Create a Digital Twin adapted perfectly to the physical twin.
- Generate perturbations and anomalies to make it a super-DT.

Overall, this state-of-the-art analysis highlights the need for continued research, innovation, and investment in Digital Twin technologies. With further advancements and integration into real-world applications, Digital Twins have the potential to revolutionize maintenance and resilience practices, leading to improved operational efficiency, reduced downtime, and increased system performance.

Part IV

General Conclusion

Chapter 4

General Conclusion

In this paper, an exploration of the power and potential of Digital Twins as a preventative and resilience tool was presented, along with relevant definitions and a comprehensive state-of-the-art analysis. However, it is crucial to acknowledge that while Digital Twins offer numerous benefits, they are not a one-size-fits-all solution for all maintenance and resilience challenges.

Implementing Digital Twins requires significant investments in terms of data collection, analytics capabilities, computing power, and the recruitment and training of skilled personnel. These resources are necessary for ensuring the proper operation and maintenance of Digital Twins. It is also important to note that while Digital Twins can provide valuable insights and analysis, they cannot entirely replace human intuition and expertise in decision-making processes. Instead, they should be viewed as a complementary tool that enhances and augments traditional maintenance practices.

Looking ahead, the forthcoming paper will focus on the realization of a cognitive super-Digital Twin. This advanced version of the Digital Twin concept not only replicates the physical IoT Twin with precision but also possesses the unique capability to detect and generate perturbations for enhanced prevention and resilience strategies. By introducing controlled disruptions and analyzing their impact, the super-Digital Twin aims to improve the overall performance and preparedness of systems.

While the cognitive aspect of the super-Digital Twin is a significant component of its capabilities, the emphasis of the forthcoming work will be primarily on the perturbations and disruptions introduced by the twin. By investigating the effects and responses to various perturbations, a deeper understanding of system behavior and the development of more effective preventive and resilience measures can be achieved.

Bibliography

[1] I. Zhou, I. Makhdoum, N. Shariati, M. A. Raza, R. Keshavarz, J. Lipman, M. Abolhasan, and A. Amalipour, "Internet of things 2.0: Concepts, applications, and future directions," vol. 9, pp. 1–2, May 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9427249

[2] R. Minerva, A. Biru, and D. Rotondi, "Towards a definition of the internet of things (iot)," pp. 17–19, May 2015. [Online]. Available: https://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Issue1_14MAY15.pdf

[3] K. K. Patel, S. M. Patel, and C. Salazar, "Internet of things-iot: Definition, characteristics, architecture, enabling technologies, application future challenges," vol. 6, p. 2, May 2016. [Online]. Available: https://www.researchgate.net/publication/330425585_Internet_of_Things-IOT_Definition_Characteristics_Architecture_Enabling_Technologies_Application_Future_Challenges

[4] F. I. Suny, M. Rahman, M. Roshed, and N. T. Newaz, "Iot past, present, and future a literary survey," p. 3, Jul. 2021. [Online]. Available: https://www.researchgate.net/publication/353004403_IoT_Past_Present_and_Future_a_Literary_Surveys

[5] S. DuBravac, "The evolution of the internet of things (iot)," *Harvard Business Review*, vol. 93, pp. 72–75, Nov. 2015. [Online]. Available: https://hbr.org/2015/11/the-evolution-of-the-internet-of-things-iot

[6] S. N. Swamy and S. R. Kota, "An empirical study on system level aspects of internet of things (iot)," vol. 8, pp. 19–20–21, Oct. 2020. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9218916

[7] (2021) Difference between cloud, fog and edge computing in iot. [Online]. Available: https://www.digiteum.com/cloud-fog-edge-computing-iot/

[8] Iot predictive maintenance. [Online]. Available: https://www.aspentech.com/en/apm-resources/iot-predictive-maintenance

[9] 5 benefits of iot-based predictive maintenance. [Online]. Available: https://www.itconvergence.com/blog/5-benefits-iot-based-predictive-maintenance/

[10] J. P. G. Sterbenz, "Smart city and iot resilience, survivability, and disruption tolerance: Challenges, modelling, and a survey of research opportunities," Nov. 2017. [Online]. Available: https://ieeexplore.ieee.org/document/8093025

[11] J. D. F. J. H. G. H. Christian Berger, Philipp Eichhammer, "A survey on resilience in the iot: Taxonomy, classification and discussion of resilience mechanisms," Sep. 2021. [Online]. Available: https://arxiv.org/abs/2109.02328

[12] IBM, "What is machine learning?" [Online]. Available: https://www.ibm.com/topics/machine-learning

[13] E. Burns, "Machine learning," 2019. [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/machine-learning-ML

[14] Javatpoint, "Machine learning algorithms." [Online]. Available: https://www.javatpoint.com/machine-learning-algorithms

[15] J. point, "Unsupervised machine learning." [Online]. Available: https://www.javatpoint.com/unsupervised-machine-learning

[16] S. Bhatt. Reinforcement learning 101. [Online]. Available: https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292

[17] Wikipedia. Reinforcement learning. [Online]. Available: https://en.wikipedia.org/wiki/Reinforcement_learning

[18] geeksforgeeks. Decision tree. [Online]. Available: https://www.geeksforgeeks.org/decision-tree/

[19] T. Yiu. Understanding random forest. [Online]. Available: https://towardsdatascience.com/understanding-random-forest-58381e0602d2/

[20] Wikipedia. Naive bayes classifier. [Online]. Available: https://en.wikipedia.org/wiki/Naive_Bayes_classifier

[21] Introduction to convolution neural network. [Online]. Available: https://www.geeksforgeeks.org/introduction-convolution-neural-network/

[22] A. A. A. M. S. J. E. T. Chukwudi Nwogu, Giovanni Lugaresi. (2020) The digital twin paradigm for smarter systems and environments: The industry use cases. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/multilayer-perceptron

[23] A. Biswal. (2023, Apr.) Recurrent neural network(rnn) tutorial: Types, examples, lstm and more. [Online]. Available: https://www.simplilearn.com/tutorials/deep-learning-tutorial/rnn

[24] M. Saeed. (2022, Sep.) An introduction to recurrent neural networks and the math that powers them. [Online]. Available: https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/

[25] A. Chugh. (2021) Deep learning | introduction to long short term memory. [Online]. Available: https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/

[26] Q. H. M. Imtiaz Ullah. (2022, Jun.) Design and development of rnn anomaly detection model for iot networks. [Online]. Available: https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/

[27] Javatpoint. Difference between machine learning and deep learning. [Online]. Available: https://www.javatpoint.com/machine-learning-vs-deep-learning

[28] K. Lamb, "Principle-based digital twins: a scoping review," pp. 19–20–21, Dec. 2019. [Online]. Available: https://www.cdbb.cam.ac.uk/files/scopingreview_dec20.pdf

[29] X. D. Z. L. J. T. Weifei Hu, Tongzhou Zhang, "Digital twin: a state-of-the-art review of its enabling technologies, applications and challenges," *Journal of Intelligent Manufacturing and Special Equipment*, pp. 19–20–21, Jul. 2021. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/JIMSE-12-2020-010/full/html

[30] A. Sharma, E. Kosasih, J. Zhang, A. Brintrup, and A. Calinescu, "Digital twins: State of the art theory and practice, challenges, and open research questions," pp. 19–20–21, Dec. 2020. [Online]. Available: https://arxiv.org/pdf/2011.02833.pdf

[31] E. H. Glaessgen and D. S. Stargel, "The digital twin paradigm for future nasa and u.s. air force vehicles," p. 7, 2012. [Online]. Available: https://ntrs.nasa.gov/api/citations/20120008178/downloads/20120008178.pdf

[32] J. D. Hochhalter, W. P. Lester, J. A. Newman, E. H. Glaessgen, V. K. Gupta, V. Yamakov, S. R. Cornell, S. A. Willard, and G. Heber,

"Coupling damage-sensing particles to the digital twin concept," p. 5, Apr. 2014. [Online]. Available: https://ntrs.nasa.gov/api/citations/20140006408/downloads/20140006408.pdf

[33] A. Parrott and L. Warshaw, "Industry 4.0 and the digital twin," 2021. [Online]. Available: https://www2.deloitte.com/us/en/insights/focus/industry-4-0/digital-twin-technology-smart-factory.html

[34] F. Tao, M. Zhang, Y. Liu, and A. Y. C. Nee, "Digital twin driven prognostics and health management for complex equipment," vol. 67, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0007850618300799

[35] M. N. K. Boulos and P. Zhang, "Digital twins: From personalised medicine to precision public healths," Jul. 2021. [Online]. Available: https://www.mdpi.com/2075-4426/11/8/745

[36] W. Purcell, T. Neubauer, and K. Mallinger, "Digital twins in agriculture: challenges and opportunities for environmental sustainability," *Current Opinion in Environmental Sustainability*, vol. 61, p. 1, Apr. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S187734352200104X

[37] P. Brosset, T. Matins, M. Watson, and S. Williams, "How digital twins enable autonomous operations," Feb. 2022. [Online]. Available: https://www.accenture.com/us-en/insights/industry-x/manufacturing-systems-architecture

[38] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," p. 3, Aug. 2017. [Online]. Available: https://www.researchgate.net/publication/306223791_Digital_Twin_Mitigating_Unpredictable_Undesirable_Emergent_Behavior_in_Complex_Systems

[39] M. Grieves, "Digital twin: Manufacturing excellence through virtual factory replication," p. 3, Mar. 2015. [Online]. Available: https://www.researchgate.net/publication/275211047_Digital_Twin_Manufacturing_Excellence_through_Virtual_Factory_Replication

[40] A. Badach, "Digital twins in iot," pp. 19–20–21, Dec. 2022. [Online]. Available: https://www.researchgate.net/publication/366167747_Digital_Twins_in_IoT

[41] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Internet of things 2.0: Concepts, applications, and future directions," p. 8, Mar. 2020. [Online]. Available: https://www.researchgate.net/publication/339802823_Characterising_the_Digital_Twin_A_systematic_literature_review

[42] J. Shepard, "What's a digital shadow and how does it relate to a digital twin?" Sep. 2022. [Online]. Available: https://www.analogictips.com/whats-a-digital-shadow-and-how-does-it-relate-to-a-digital-twin-faq/

[43] W. SA, "Difference between digital twin, digital model, and digital shadow." [Online]. Available: https://www.wizata.com/knowledge-base/difference-between-digital-twin-digital-model-and-digital-shadow

[44] C. W. Wil Van der Aalst, Oliver Hinz, "Resilient digital twins: Organizations need to prepare for the unexpected," Sep. 2021. [Online]. Available: https://www.researchgate.net/publication/354764507_Resilient_Digital_Twins_Organizations_Need_to_Prepare_for_the_Unexpected

[45] L. Seppälä, "Digital model, digital shadow, or digital twin – what is at the core of data-driven shipbuilding?" pp. 19–20–21, Aug. 2020. [Online]. Available: https://www.cadmatic.com/en/resources/blog/digital-model-digital-shadow-or-digital-twin/

[46] T. A. A. K. D. S. T. A. Thomas Bergs, Sascha Gierlings, "The concept of digital twin and digital shadow in manufacturing," vol. 101, Sep. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827121006612

[47] G. Fuller, "Digital model, digital shadow, and digital twin: their places in eiot," Sep. 2021. [Online]. Available: https://www.psa.inc/company/news/digital-model-vs-digital-shadow-vs-digital-twin-and-their-place-in-eiot/

[48] D. D. Teams, "4 types of digital twins (basic overview with examples)," Nov. 2022. [Online]. Available: https://digitaldirections.com/4-types-of-digital-twins-basic-overview-with-examples/#what-are-the-main-types-of-digital-twins

[49] IBM, "What is a digital twin?" Jun. 2022. [Online]. Available: https://www.ibm.com/topics/what-is-a-digital-twin

[50] M. Kor, "Integration of digital twin and deep learning for facilitating smart planning and construction: An exploratory analysis," Jun. 2021. [Online]. Available: https://hj.diva-portal.org/smash/get/diva2:1569258/FULLTEXT01.pdf

[51] S. Software, "Machine learning supercharges real-time digital twins," May 2021. [Online]. Available: https://www.scaleoutsoftware.com/featured/machine-learning-supercharges-real-time-digital-twins/

[52] S. A. Mergen Kor, Ibrahim Yitmen, "An investigation for integration of deep learning and digital twins towards construction 4.0," Mar. 2022. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/SASBE-08-2021-0148/full/html

[53] P. Sarkar, "Digital twin modeling using machine learning and constrained optimization," Oct. 2022. [Online]. Available: https:

//towardsdatascience.com

[54] R. Dolphin, "Lstm networks | a detailed explanation," Oct. 2021. [Online]. Available: https://towardsdatascience.com/lstm-networks-a-detailed-explanation-8fae6aefc7f9

[55] L. Grinsztajn, E. Oyallon, and G. Varoquaux, "Why do tree-based models still outperform deep learning on tabular data?" 2022. [Online]. Available: https://arxiv.org/abs/2207.08815

[56] F. Flammini, "Digital twins as run-time predictive models for the resilience of cyber-physical systems: a conceptual framework," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 379, no. 2207, p. 20200369, 2021. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2020.0369

[57] P. Eirinakis, S. Lounis, S. Plitsos, G. Arampatzis, K. Kalaboukas, K. Kenda, J. Lu, J. M. Rožanec, and N. Stojanovic, "Cognitive digital twins for resilience in production: A conceptual framework," *Information*, vol. 13, no. 1, 2022. [Online]. Available: https://www.mdpi.com/2078-2489/13/1/33

[58] v. d. H. J. Bruynseels K, Santoni de Sio F, "Digital twins in health care: Ethical implications of an emerging engineering paradigm," Feb. 2018. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/29487613/

[59] T. Mortlock, D. Muthirayan, S. Y. Yu, P. P. Khargonekar, and M. A. A. Faruque, "Graph learning for cognitive digital twins in manufacturing systems," *IEEE Transactions on Emerging Topics in Computing*, vol. 10, pp. 34–45, 2021.

[60] C. Gao, H. Park, and A. Easwaran, "An anomaly detection framework for digital twin driven cyber-physical systems," in *Proceedings of*

the ACM/IEEE 12th International Conference on Cyber-Physical Systems, ser. ICCPS '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 44–54. [Online]. Available: https://doi.org/10.1145/3450267.3450533

[61] S. R. Chhetri, S. Faezi, A. Canedo, and M. A. A. Faruque, "Quilt: Quality inference from living digital twins in iot-enabled manufacturing systems," in *Proceedings of the International Conference on Internet of Things Design and Implementation*, ser. IoTDI '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 237–248. [Online]. Available: https://doi.org/10.1145/3302505.3310085