International Journal of Science and Research Archive
eISSN: 2582-8185
Cross Ref DOI: 10.30574/ijsra
Journal homepage: https://ijsra.net/

(Review Article)

Machine learning for predictive maintenance in self-healing software services

Nagaraj Bhadurgatte Revanasiddappa *

Individual Researcher, Engineering Technology Leader, USA.

International Journal of Science and Research Archive, 2022, 05(01), 162-181

Publication history: Received on 17 December 2021; revised on 25 January 2022; accepted on 28 January 2022

Article DOI: https://doi.org/10.30574/ijsra.2022.5.1.0027

Abstract

Automated self-repairing systems, catalyzed by predictive maintenance and ML, are the new formative model in today's software industry. Predictive maintenance is used widely for anticipating and avoiding system failures, and it forms a critical element in improving reliability and performance of software services. This article focuses on machine learning and application like predict and prevent maintenance and self-healing system that helps minimize downtimes, increases overall system performance of a system and selects optimal use of resources. The first part gives an overview of the development of software systems whereby an analysis was done on the general transition from simple static systems to autonomous systems that can self repair. It talks about predictability maintenance as a way of foreseeing failures, therefore avoiding undue disasters which are likely to inconvenience end users. The article then proceeds to explain the underlying ideas of predictive maintenance and differentiating it from the preventive and reactive kinds. This paper elaborates the methodology of how predictive maintenance integrates with self-healing and also with focus on without human intervention techniques used in autonomous systems wherein faults are corrected in real time. The discussion goes on to Machine learning methods ubiquitous in the predictive maintenance. Fault detection is tackled in supervised learning, and anomalous patterns in system logs are detected in unsupervised learning; reinforcement learning is applied to form recovery models to enhance self-healing mechanisms. The article goes on to discuss some of the difficulties in data quality, scaling of models and the combination of ML systems into existing structures, with solutions put forward including the use of cloud and the use of mix-model solutions. The use of predictive maintenance in the real world includes; In the healthcare sector and in cloud computing to name but a few. These case studies illustrate how organizations deploy these technologies for dependable and high performing operation in critical applications. Last but not least, the article brings vision for the future, with new interesting enhancements in self-healing solutions and the progression of AI/ML making the shift towards autonomous and self-optimizing systems.

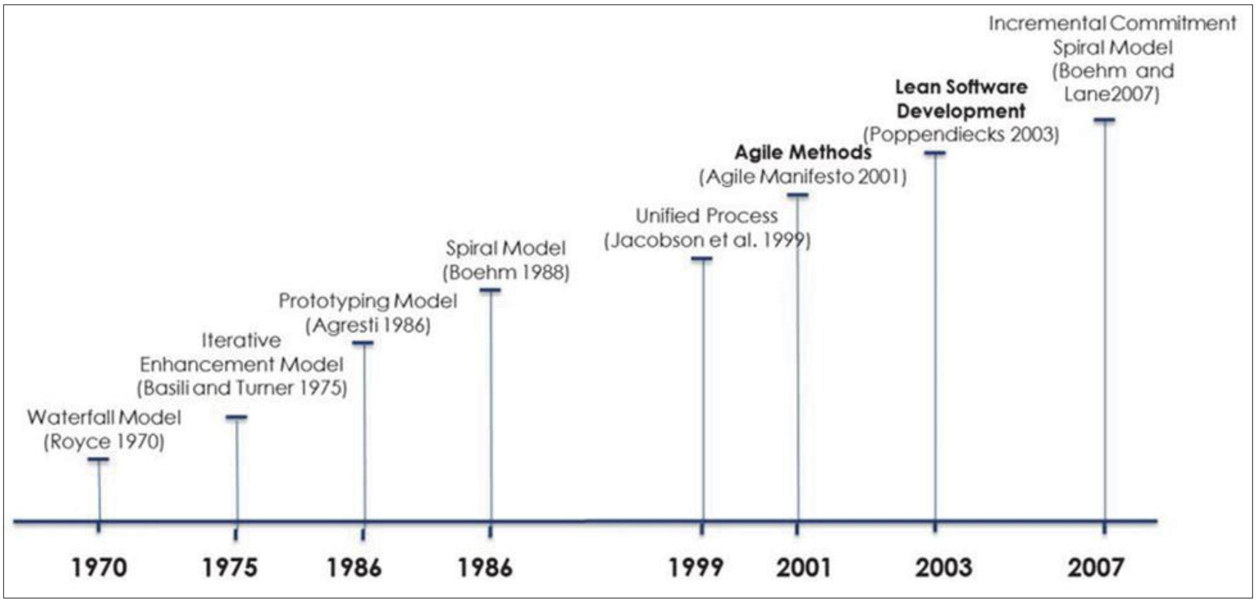
Keywords: Self-healing software; Predictive maintenance; Machine learning; Fault detection; Anomaly detection

1. Introduction

The evolution of software systems from being static monolithic entities to self healing systems is evidence of the complexity of managing today's complex computing environment. In early systems, alarms depended mostly on the involvement of specialized engineers for fault identification, diagnosis, and correction in the system. This approach/procedure, however, fail as systems became complex and demands for reliability enhanced [1]. However, the beginning of Autonomic Computing in 2001 should be considered a turning point. IBM initiated the variability aspect of self-management with the MAPE-K loop of Monitor, Analyze, Plan and Execute based on knowledge in the environment [2]. This framework allowed the software systems to control their state, which led to the concept of the self-healing systems. Healing systems are those that are able to detect, at runtime, that there are problems with the system, analyze them, and automatically repair the problem area. They include policies – goal policies for attaining objectives, or utilitarian policies for fulfilling more objectives at once [1]. While first Self healing systems where built to detect faults and perform recovery operations, current approaches stress unobtrusiveness, making sure that user interactions are

* Corresponding author: Nagaraj Bhadurgatte Revanasiddappa

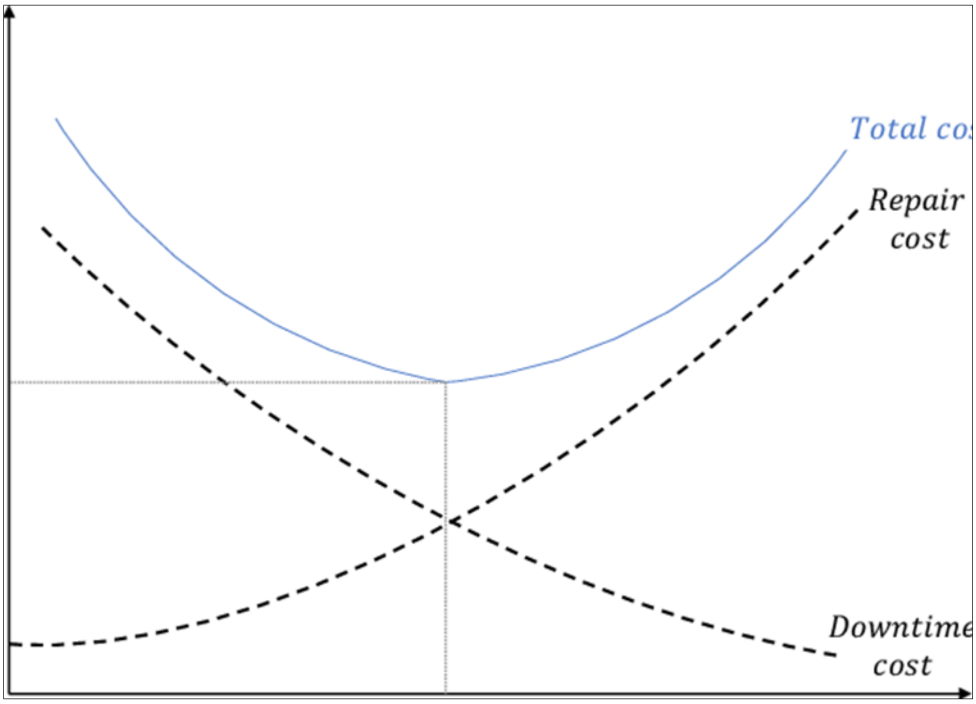
not a problem while enforcing availability, reliability and stability [2]. Nonetheless there is still room for difficulties. The currently existing systems and structures need incremental introduction of self-healing capabilities as they may be incompatible with the concept. Furthermore, the assessment of self healing systems is rather challenging because it implies parameters like the speed of healing, accuracy of fault detection, and adaptability under tougher conditions [1] These systems are tested by using simulators and failure trace methodologies so that the actual complex environment can be mimicked to fix the processes over multiple cycles [2]. This evolution shows a clear transformation from the exception-based to the proactive approach in systems management, which is a never-ending process as the goal of complex software environments is to be more reliable, robust and optimised.



Adapted from ResearchGate

Figure 1 Software development methods timeline

When it comes to self healing software services, PM is hugely significant in keeping services running while promoting system dependability. Machine learning techniques can be used in the development of predictive maintenance methodologies since this approach is used in developing techniques, which could likely predict when system failure is most probable hence ensuring less down times of software systems. The advantage of self-healing systems which detect errors and attempt to correct themselves without the need for intervention from human operators is that the system gets to be corrected before it leads to a catastrophe, given that Durations allows constant monitoring and predictive capabilities. Machine learning combined with predictive maintenance in context of self-healing software services has number of benefits over conventional approaches. While traditional methods involve waiting for system failures to happen, or performing occasional preventive checks, predictive maintenance constantly evaluates system logs, performance parameters and history for patterns which may cause system failure. This approach also increases the reliability of the system since problems are often solved before they affect the performance of the system, it also ensures that software services are optimal. For instance, when applied in self – healing software, predictive maintenance eliminates the need for engaging human intervention and guarantee the swift execution of maintenance action. This leads to increase in system availability, thus minimum system downtime and enhanced user experience. Lastly, predictive maintenance also leads to the overall durability of self-healing software services with predicting abilities, as well as allows them to recover from issues, which occurred due to changing conditions without requiring external intervention. Using the tools of machine learning, it becomes possible to support the constant and smooth operation of self-healing software systems while at the same time keeping the costs and resulting downtimes at a minimum [3].

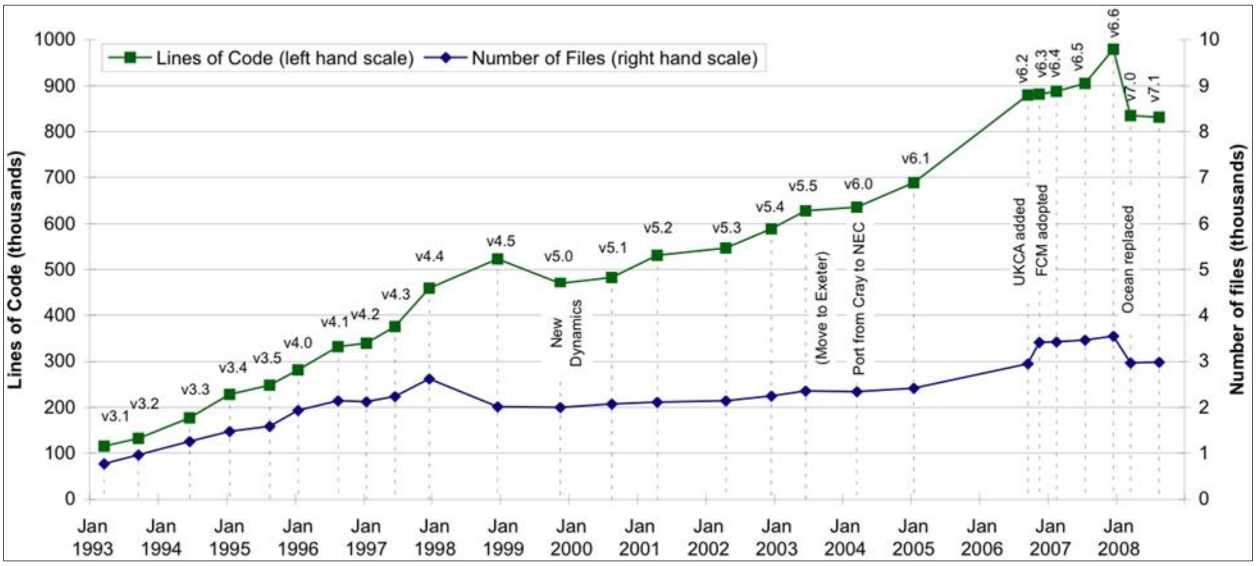


Adapted from ResearchGate

Figure 2 Predictive maintenance cost model

1.1. Current Challenges in Software Services

When dealing with IT infrastructures where the general level of complexity is rapidly growing, a task of coordinating the software components in various environments is not an easy one. In self healing software services there are many components and frameworks involved in service construction, from third party libraries to custom built applications and services which cause error and down time and failure in systems. The problems stated above are solved through machine learning in predictive maintenance by constantly assessing system's status and warning of possible failure [4]. Here it is possible to mention a few advantages of using machine learning, namely, anomaly detection with help of regression models allowing predicting times when intervention is necessary. This minimizes on time required and increases the reliability of the total system. Also, as different software architectures are being developed, machine learning aids in the coordination of multiple teams in different locations or the consistency of design documentation with the actual code implementors. inline with self-diagnosing that happens in predictive maintenance, issues are solved by systems themselves without involving individuals. In the end, machine learning can improve IT more exact and efficient structures as modern technologies get more complicate [4]. The increasing costs and potential consequences of the downtime continue to be another important factor for organizations encountering the complexity in relations to IT systems. Any unexpected interruption in the system puts the organization at risk of losing customer revenue, efficiency and trust. In the context of self-healing software services, such risks are magnified by the fact that self-healing has to happen across distributed systems. Machine learning for PM can bring a disruptive change, and it allows early identification of possible failures and timely adjustment of the system to reduce outage times [5]. Consuming real-time system data as well as histories of how events have occurred in similar systems, machine learning algorithms predict future system weaknesses even before they transform into major incidents. However, these systems need to be protected against any risks that may prevail in its environment such as data leak, bad integration, or excessive resource utilization. It cannot be business as usual for ensuring maximum availability of services after the dotted line, but such optimal availability cut with the need to factor in cost in service provision and business risk requires a best of both worlds solution where there is predictive maintenance complemented by very watertight contractual terms and fail safe back-up mechanism. This approach also makes it possible for organizations to sustain competitiveness as it is designed for cooperative models in dynamic technological contexts [5].



Adapted from Easterbrook

Figure 3 Growth of code complexity

1.2. Objectives and Scope of the Article

This article looks at how the concepts of, machine learning, predictive maintenance and self-healing Software systems are interconnected. Predictive maintenance has become more efficient due to an integration of the machine learning algorithms to allow the real-time, and historical data, to be analyzed with computer-like accuracy to determine possible failures. This capability increases system dependability, reduces time off, and synchronizes maintenance. Self-healing software adds these advantages by detecting as well as repairing problems independently of other people. The use of such technologies can be said to open up for a new era in the improved resilience of operating systems in functionalities in various industries. Specifically, the exploration concerns how machine learning techniques including anomaly detection and classification, bolster predictive maintenance frameworks as well as enable self-healing mechanism confronting difficulties on its own. Thus, by synchronizing these progressive techniques the article tries to unearth the trails to create more stable, effective and encapsulated systems that can address the changing operational requirements in an effective manner. It also provides implementable techniques for integrating predictive maintenance into autonomous systems so that they work optimally and with great efficiency. Some strategies are the graphical integrated sensor network that assembles high-quality data and the cloud-based big data analysis platform. According to the article it is crucial to apply machine learning models to the tasks like anomaly detection, root cause analysis and fault prediction. In addition, the discussion points to the importance of distributed architectures in providing coverage for decentralized decision-making needs, flexible structures that can continue to operate regardless of the occurrence of fault within the systems. This is accompanied by integration with self-healing software that embeddeds other recovery mechanisms. These strategies are particularly germane to those segments of the industry that require highly reliable systems such as the aerospace, healthcare, and manufacturing industries. Through outlining the content of the article on the direct prediction of maintenance in autonomous operational conditions, the authors create a solid plan to help organizations practice efficient maintenance and decrease expenses to reflect the systems' accuracy in operating optimal systems.

2. Predictive Maintenance in Software Services

2.1. Definition and Scope

Predictive Maintenance or PdM, sometimes called condition-based maintenance (CBM), is an approach that identifies and forecasts equipment failures and schedules maintenance appropriately. While there is the reactive maintenance where maintenance is performed after equipment breakdown, preventive or planned where maintenance is done based on the recommended periodicity and there is the PdM which uses current data of the real condition of the systems and components. From sensors like vibration, temperature, ultrasonic and others PdM draws relation and correlation and forecast when maintenance is required. The fundamental concept behind the PdM concept is to carry out maintenance in the context of operations and maintenance (O&M) minimally. This decrease the time that is needed to carry out maintenance thus reducing the cost and also increases the service life of equipment. Consumers receive fewer unexpected plant shutdowns, lower inventory costs and fewer reports of products failing to meet design standards while manufacturers experience fewer forced outages, reduced inventory costs and reduced calls for premature component replacement. But the shift to implementing PdM could be costly and requires the use of many resources. Among them, the high cost of instruments including state of the art sensors, and the issue of data acquisition, analysis, and decision making. However, the fact that PdM allows organizations striking a perfect balance between the frequency of maintaining equipment and the costs incurred is something that is a big plus for many industries that are looking to improve reliability and efficiency in today's world. They herald a more efficient and intelligent system of maintenance approaches [6].

Differences Between Predictive, Preventive, and Reactive Maintenance

Tertiary maintenance plans include Reactive maintenance, Preventive maintenance, and Predictive maintenance; all of which have disparities in their approaches, usefulness, and price implications on the maintenance of processes.

- **Reactive Maintenance (RM),** also known as run-to-failure maintenance, is a traditional method of equipment maintenance founded on the consistent idea of equipment repair upon failure. This approach is simple since no upkeep is done on the system unless it is down. Despite the fact that RM ensures optimum use of equipment, it has so many demerits. In this case, leaving the potential problems unsolved may result in expensive repair work, aggravation of other parts, or even complete machine downtime. Also, RM leads to high operation expenses because it demands firms to stock many spare parts on hand [6].
- **Preventive Maintenance (PM)** seeks to minimize the probability of failure through planned maintenance activities by time and utility cycle. PM is done on the equipment even if it is working well with the intention of averting breakdowns on the equipment. Nevertheless, the main disadvantage of PM is that such maintenance generates excessive work. Equipment may be maintained or refurbished well before the expiry of its useful life which comes at an extra cost. On the other hand if maintenance is done when it is due, then equipment may still be prone to wear related failures, implying higher repair costs, or even total failure [6].
- **Predictive Maintenance (PdM)** provides a process of enhancing Maintenance that is much more advanced than preventive maintenance. It employs various detective data including the vibration, thermal and ultrasonic data taken from other sensors and monitoring systems for real time equipment health status. Probabilistic models on this information to know when a piece of apparatus or a system is probably going to fail, maintenance is done on the apparatus or system just a bit sooner that it is most likely to fail. PdM eliminates or at least reduces sometime of failure and in the process, minimizes time downtime and costly repairs. Even though PdM implies higher initial expenditures to invest in sensors and analysis tools, it pays off in the long run by improving plans for periodic check-ups, decreasing ad hoc downtimes, and obtaining equipment service life increases [6]. While on the other hand, while PM is more generalized which leads to the general maintenance of equipment without necessarily having to give recommendations based on the status of the equipment, PdM allows maintenance that is done specifically and hence costs less.

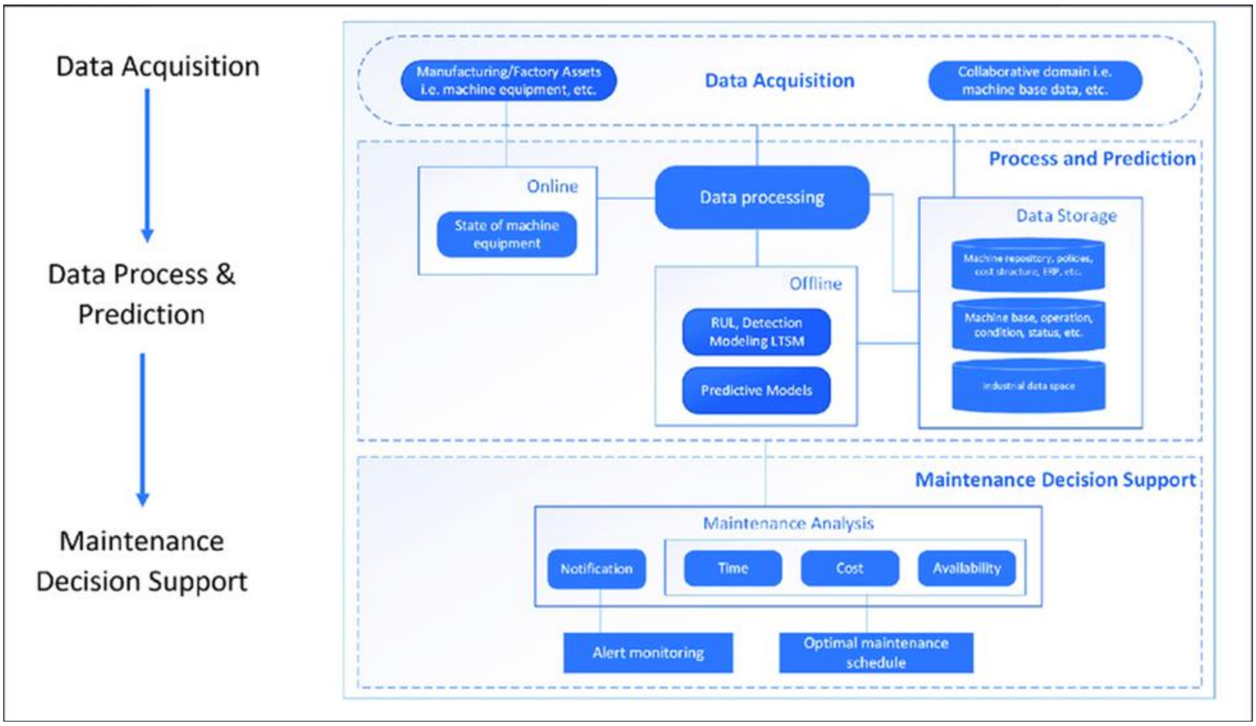
Table 1 Benefits, Challenges, and Applications of RM, PM, and PdM [6]

Maintenance Type	Benefits	Challenges	Suitable Applications	Unsuitable Applications
RM	• Maximum utilization and production value • Lower prevention cost	• Unplanned downtime • High spare parts inventory cost • Potential further damage for the equipment • Higher repair cost	• Redundant, or non-critical equipment • Repairing equipment with low cost after breakdown	• Equipment failure creates a safety risk • 24/7 equipment availability is necessary
PM	• Lower repair cost • Less equipment malfunction and unplanned downtime	• Need for inventory • Increased planned downtime • Maintenance on seemingly perfect equipment	• Have a likelihood of failure that increases with time or use	• Have random failures that are unrelated to maintenance

PdM	• A holistic view of equipment health • Improved analytics options • Avoid running to failure • Avoid replacing a component with useful life	• Increased upfront infrastructure cost and setup (e.g., sensors) • More complex system	• Have failure modes that can be cost-effectively predicted with regular monitoring	• Do not have a failure mode that can be cost-effectively predicted

2.2. Relevance to Self-Healing Systems

Predictive Maintenance (PdM) and self-healing are two separate principles but aim at improving system reliability and avoiding time losses. PdM uses real time information obtained from sensors the same as vibration, temperature and pressure to estimate chance of failure [6]. This predictive evaluation points you to the right time that should be taken for maintenance, preferably before the equipment fails thus cutting down on time that is not needed and increasing the time when maintenance should be carried out. Self-healing systems in software, on the other hand, are designed to be able to identify problems, counter them and rectify the same before the intervention of any human being. The primary element in problem detection and management and self-healing systems is contingency. In the same manner that PdM keeps an eye on physical systems in order to know when equipment will fail and addressing it before it happens, self-healing software systems perform similar goals in accomplishing system health and avoiding disturbances, by repairing code or reconfiguring resources if problems are detected. In both approaches, monitoring as a continuous process remains important. In PdM sensors give continuous feed back of health of the machinery while in case of self-healing systems, monitoring tools lookout for software glitches or hardware failures. Both strategies intend to make optimum use of data, keep interventions timely so that errors do not accumulate over the time. When using both predictive maintenance along with self-healing principles, the system can be further optimized. For instance, while PdM can pick up physical equipment problems, self-healing software can handle the system's response for software or network problems. This kind of combination provides a complete option that helps decrease operation interference, lengthen system longevity, and strengthen the depth and sturdiness of intricate systems.



Adapted from ResearchGate

Figure 4 Predictive maintenance process and framework

2.3. Benefits of Predictive Maintenance in Software Services

Some of the key areas that include Predictive maintenance (PdM) is a trending topic that is even applicable in software services industries to decrease response time and increase infrastructure availability. In software services, PdM uses algorithms and technologies of the Fourth Industrial Revolution to predetermine system failures and represents a great value to maintenance management. As reported in the analysis section, PdM therefore enables an organization to assess the overall health of the system and make the right decisions to make any necessary adjustments before the system fails. Moving from response-based management to prediction improves dependability and productivity of services, offering significant cost advantages over conventional maintenance strategies. First, PdM is extremely effective in decreasing the overall cost of maintenance in the field of software services. Conventional methods of maintenance like the preventive maintenance are characterized by regular interferences with or without concerning the status of the system in question. This may lead to unrequired repairs work or even the early replacement of equiment which are not efficient. PdM, on the other hand, employs actual-time data from the diverse system components to anticipate failures, minimizing proactively interruptions and(parseInt(Dr)#) ultimately decreasing general maintenance costs. They also reduce operation cost and make it possible to address issues that affect systems causing them to be a one-stop solution to most problems.

PdM also increases efficiency with regard to service delivery. PdM simplifies service supply chains and the overall operation as a result of automating maintenance schedules. It creates the capability for service providers to respond more efficiently and proactively to failures that are likely to diminish the customer experience and reduce overall customer loyalty. The above processes when automated makes it possible for businesses to Be in a position to deliver their services well without having to complicate the operations that they offer to their clients. Therefore, the companies implementing PdM activities enjoy increased competitiveness with regard to offering more efficient service and increasing customer loyalty, and thus securing enhanced prospects to increase market share. In addition, it enhances decision making by giving decision makers real-time information with which to make their decisions. The problem with conventional maintenance approaches is that they are based on guesses or fixed plans such that an action may not be carried out when it needed or conversely, a system may be worked on unnecessarily. PdM solves this problem by giving decisions based on real data about the state of a system using predictive analytical tools. This also increases response rate to emerging problems hence reducing possible lengthy breakdowns and enhancing operational hours. Contemporary information decision making guarantees effective resource allocation so that the sustainability of the organization is accorded the much-needed boost.

In the last, PdM helps in data integration and present a single unified view of the health of the systems. Unlike some PdM's predecessors that call for data collection from multiple sources, the system brings all data to one application, making the monitoring process much easier. This is easier because such an approach gives a bird's eye view of things, and organizations are well placed to identify issues which need to be addressed before they escalate. The end result has instituted a more efficient maintenance process as more time and resources can be saved. Thus, expanding the participation of PdM in software services should bring definite advantages, such as cost reductions, increased effectiveness of methods, improvements to decision-making processes, and service quality. With the help of real-time analytics and data integration, companies are ready to achieve even higher results in an environment that is becoming gradually more competitive. [7]

Figure 5 Benefits of predictive maintenance. Adapted from FasterCapital

3. Machine Learning Techniques for Predictive Maintenance

3.1. Supervised Learning for Fault Detection

Supervised learning is widely used as one of the main stances in fault detection since it relies on labeled data to educate the models, which allow finding failures. Based on past occurrences and achieving similar result set outcomes, net based supervised learning techniques of classification and regression models offer solutions to numerous fields. These methods have become absolutely essential for anything from predicting business failures to checking the reliability of storage systems and power plants. The next sub section undertakes an analysis of the specific ways in which the prediction of failure has been made possible by classification and regression, as important tools which add value to business in operational critical operations.

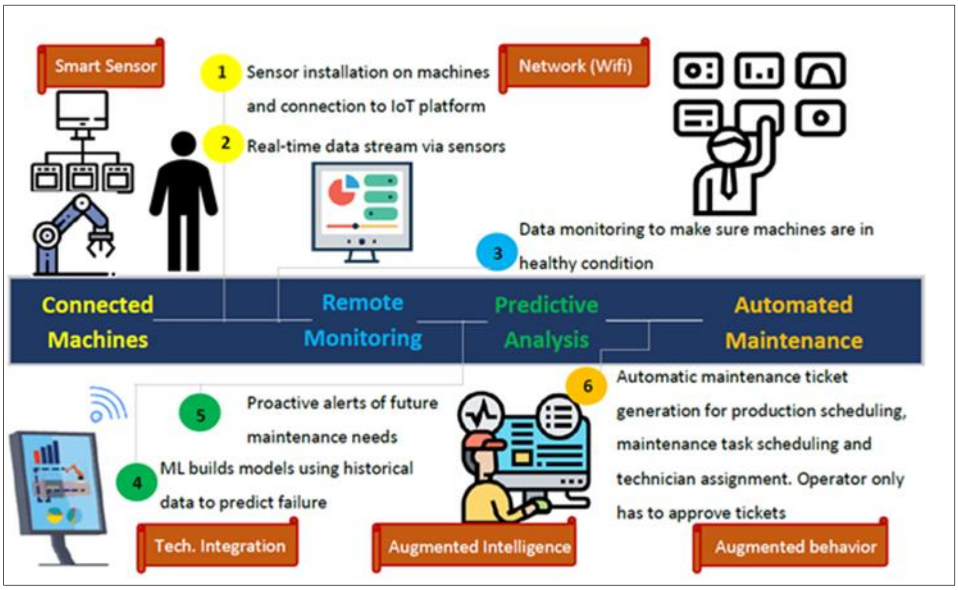
Classification and regression are widely used tools for failure prediction, as well as used in many fields to improve equipment performance and decision making. The effectiveness of these techniques is in the analysis of the accumulated and current data, as well as in the pattern recognition of potential failures. Its presence in business, storage systems, as well as in the power industry, prove their efficiency.

In business failure prediction (BFP), the classification models preferred are support vector machines (SVM), classification trees, and naïve Bayes among others. These models work with predicting FIR and AFR with the use of financial ratios and other similar measures and place companies in “failure” or “non-failure” categories. It is essential for financial institutions, governments and investors because of this categorization to minimize on risk. The effectiveness of these models depends on their ability to learn from existing case and be able to draw such knowledge to formulate results with new cases. For instance, the study establishes that such models are capable of detecting ominous signs of financial distress that, in turn, offer useful warnings to the relevant stakeholders. This is especially the case most especially during the period of financial vagaries for instance financial crises, wherein business failures ripple throughout the whole economy [8].

They are used widely in storage systems, as regression and classification techniques are used for predicting hardware failures, especially for hard drives. Contemporary hard drives contain features known as Self-Monitoring, Analysis, and Reporting Technology (SMART) attributes that record working parameters. These attributes help identify the drive failures with impressive accuracy with models such as Classification Trees (CT), Regression Trees (RT). For example, the practical reliability assessment of a CT model yields more than 95% failure prediction accuracy with false alarm probability of less than 0.1%. Regarding predictive accuracy, these models surpass prior tools such as artificial neural networks while being easier to interpret. Furthermore, the RT models offer a health score of the part, the relative crucial of which determines its maintainability with respect to probable failures. This capability noticeably increases the dependability of storage systems and also decreases operational expenses since problems are dealt with before they become glaring [9].

In power industry, classification and regression techniques is used in the maintenance and repair systems for enhancing the performance of power equipment. The predictions about individual sectors are as follows: The power plants are affected by difficulties associated with the increased capital intensity of equipment and the vulnerability of becoming outdated on machinery. Predictive models, based on information collected from numerous sources, reveal failure probabilities; in result, operators are capable of focusing on the failures that require the most attention and should allocate resources. These techniques enhance reliability, that is, low rates of failure, with reduced time off line and maintenance expenses. For example, regression models can estimate the rate of decline of critical parts, so that operators of the plant can be able to prepare for when the products of the plant will fail so that they can get prepared to have them serviced. Such measures result in the provision of constant energy and minimal losses occasioned by the outage than when the power is expected without any prior notice [10].

Being precise and flexible, the classification and regressions become critical to failure prediction. These models offer an understanding directing organizations towards changing from repair centered maintenance to models that predict failure. This shift does not only scale back costs but in addition improve operational effectiveness and reliability across various industries. It will and help to develop and continue to apply their methods and solutions to improve the failure prediction and provide long lasting and reliable and complex systems.



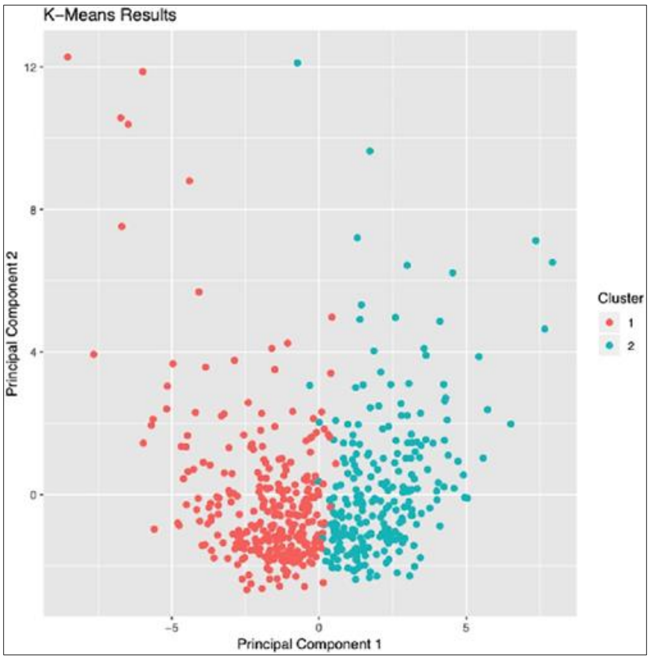
Adapted from Medium

Figure 6 Equipment failure prediction using machine learning

3.2. Unsupervised Learning for Anomaly Detection

Anomaly detection is ingrained with unsupervised learning since the manner of identifying unusual patterns in datasets without employing labeled data. This finding is particularly valuable in situations where abnormal cases are few and far between or where getting labels is costly. Clustering and outlier are two friendly applied methods which used to detect anomalies. In clustering the objective is compacting data points into clusters and the point that does not fit in any cluster are considered as outliers. With regards to segmentation, k-means or DBSCAN etc. are used wherein the isolated points are possible anomalies [11][12]. Clustering in network anomaly detection efficiently differentiates regular traffic pattern from anomalous activity by segregating out less familiar feature spaces [12].

While, Outlier detection involves determination of individual data points that are quite distinct from other data points. They effectively use distances or densities as metrics to raise alerts. For instance, density-based strategies pop points in sparse areas as outliers, which is useful in real-time monitoring for aural networks or embezzlement [13]. Each of the two approaches has its advantages. Clustering is more profound at finding set pattern in data that makes it ideal in studying group behavior while, outlier detection is ideal in identifying isolated incidences. These types of techniques are sometimes blended together to increase the reliability in developing anomaly detection systems [11][13]. Examples of such application shown in network anomaly detection are proving the efficiency of these methods in dynamic and large-scale networks. Clustering enables the identification of 'typical' traffic that outlines the formation of standard traffic of the system while outlier detection separates outliers that point to systems threats [12]. Thus, by using these methods together, an even higher accuracy of detection is achieved without the emergence of a large number of false-positive results. The real-world experience displayed by the authors their capacity to massive change in the data environment, guaranteeing sound anomaly detection [11][12]. However, some issues are still exist, especially for the high dimensional or time-varying data. New studies which involve the use of clustering and outlier detection together with other techniques in machine learning are revealing as solutions to scale and flexibility [11]. These methods increase the rate of detection without reducing the degree of freedom required to deploy it in practice. In conclusion, among the loosely supervised learning methods like clustering and outlier detection, are central to performing an anomaly detection. Because of the possibility to work with the unlabeled data, explore the relationships between the variables and update the findings depending on the situation, these algorithms are further important for various domains, including the network security and fraud analysis [11][12][13].



Adapted from ResearchGate

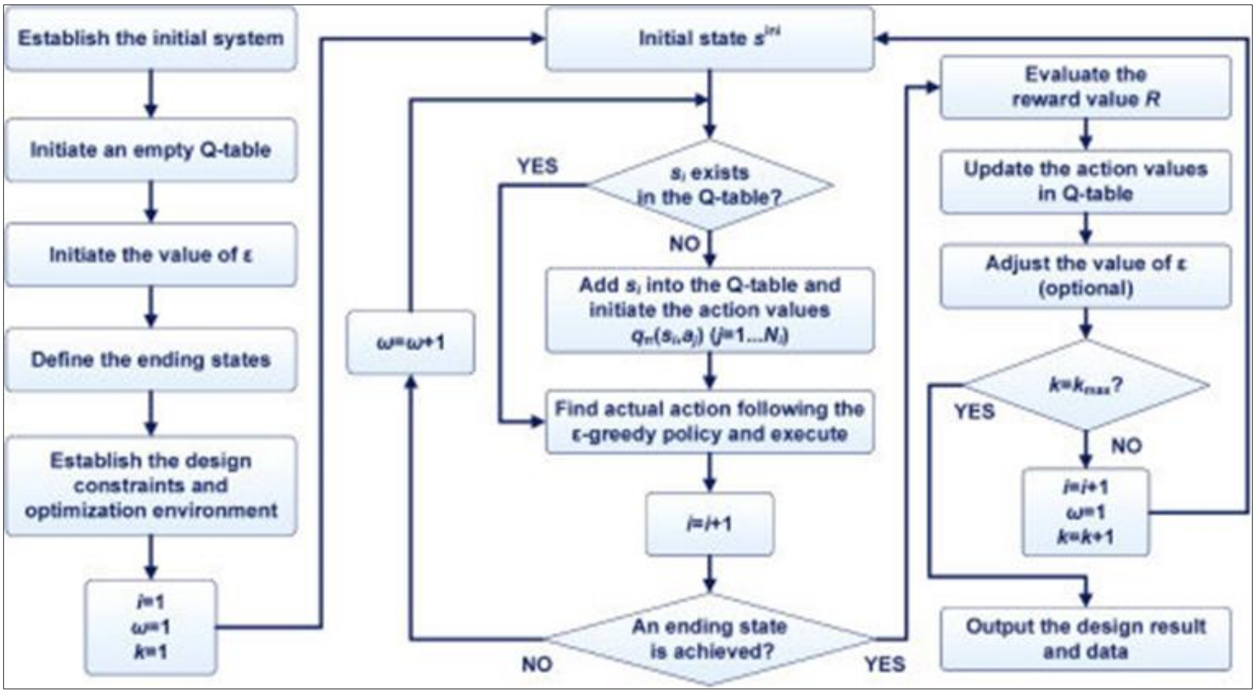
Figure 7 Scatter plot showing the result of machine learning clustering

3.3. Reinforcement Learning for Self-Healing: Adaptive Models for Automated System Recovery

Supervised learning is still widely popular for self-diagnostic systems and automated yield in cloud management and other extensive IT systems, but reinforcement learning or RL has been receiving more attention as one of the main revolutions in self-healing systems. In its simple form, RL allows systems to discover beneficial behaviors through engagements with an environment, and with corresponding incentives or punishments. This makes RL an ideal candidate for applications where the system must itself find problems, learn changes to conditions, and resume from failures in real-time especially in real-time environment such as cloud computing [14].

This is perhaps one of the more profound problems in cloud systems, the problems of providing “always on” services while containing the impact of failures and disruptive events. Self-healing systems based on RL seeks to address these issues as they monitor system performance, make diagnosis on anomalies or system failures and make appropriate corrective actions on their own without human interferences. Every failure can be seen as the result of a suboptimal path in the RL agent space; therefore, slowly but surely, an RL agent can teach the system resilience to failure, at least based on the attempted procedures. These adaptive models are especially advantageous in such a volatile cloud environment, characterized by the rapid turnaround in infrastructures and required concomitant changes in the system [15]. The architecture of RL-based self-healing models typically includes three key components: three components of reinforcement learning namely environment, agent, and reward function. They refer to all the system factors such as the servers, networks, and application which interrelate with one another. The agent is the decision maker in the system which monitors the state of the failure, takes the appropriate actions when a failure is detected, and will experience some level of feedback based on the consequences of the action which was taken. In RL, the reward function specifies the way the agent is incentivized for desirable action or otherwise. Periodically, the agent updates its policy, which is the means of selecting the most appropriate action for implementation from among those identified by the agent as potentially effective, given the agent’s sense of the environment and of what has happened [14]. Another advantage of RL in self healing systems is that it can suit unforeseen failures. Classic methods of addressing faults are usually a set of rules or a specific predetermined fault recovery process. However, these methods are good when there is stability and they fail when there is new or emergent conditions. RL-based systems, however, learn from its interactions with the environment hence they are very robust and elastic to respond to new form of faults. For example, in cloud environments, RL can help the system to forecast situations where resources are likely to be overloaded or to fail and improve the workloads on backup servers or take early remedial action when system performance is affected [15].

However, the models based on RL improve the effectiveness of the recovery processes. This is due to the fact that in a normal restoration process, cloud systems would normally require several operations hence the virtue to be performed as an action. The RL agent is designed to be a time-aware agent which acts to achieve the fastest possible recovery in the least amount of time with the least interference thus achieving Optimal recovery with resource constraint. This is especially true in cloud computing since management of resources is central to the optimization of costs when delivering reliable services [14]. Another strength of RL is the possibility of its scaling up. As cloud systems become more complex self-repair models based on RL are able to increasing loads while not prompting human intervention. Due to the presence of failure scenarios, when faced with new failure conditions, the ability of the RL agent increases with experience. This capability of operation makes RL an important means of building scalable and flexible self-healing systems [15]. Therefore, to summarize, the developing application of reinforcement learning implies multiple opportunities for effective automation of system recovery as well as the growth of self-healing capacities. As opposed to traditional enforcement methods that require systems to avoid repeating their past mistakes when facing new situations, RL-based models are a smart approach to the problem of maintaining constant system availability at a large scale. With the increasing complexity and dynamism of cloud environments, the resourceful unease seems not admittedly, but a RL will most certainly be expected to be the key to providing end –to –end functional self-repairability for systems and thus enhance their reliability, reduce possible downtimes and optimize operational costs [14][15].



Adapted from ResearchGate

Figure 8 Flowchart of the reinforcement learning method

3.4. Feature Selection and Data Preprocessing: Techniques to Improve ML Model Accuracy and Efficiency

Feature selection and data preprocessing are among the important stages in ML pipelines since they help improve model performance and reduce computing time. Therefore, and by addressing such risks effectively as noise and overfitting for specific input features, more accurate and comprehensive ML models can be developed, by virtue of integrating high-quality subject data [16].

3.4.1. Feature Selection Techniques

Feature selection targets at the identification of the elements of the dataset which has the greatest potential of improving the performance of the ML model. Techniques for feature selection are generally divided into three categories:

- **Filter Methods**: These techniques are performed to assess features separately from the model through the mathematical test of correlation, mutual information, and variance thresholds. The independent variables that have high correlation with the target variable or which identifies more variance are selected, and the unimportant variables are excluded. These methods are also computationally efficient; they are also commonly employed in a preliminary manner [17].
- **Wrapper Methods**: In wrapper methods, the whole process involves training the model for each feature subset and then choosing the one with the best performance with other features. Skilled scientific strategies, for

instance recursive feature diminishment (RFE), forward or backward selection are normally embraced. Though highly efficient, wrapper methods are cost-time intensive especially when dealing with large datasets [16].

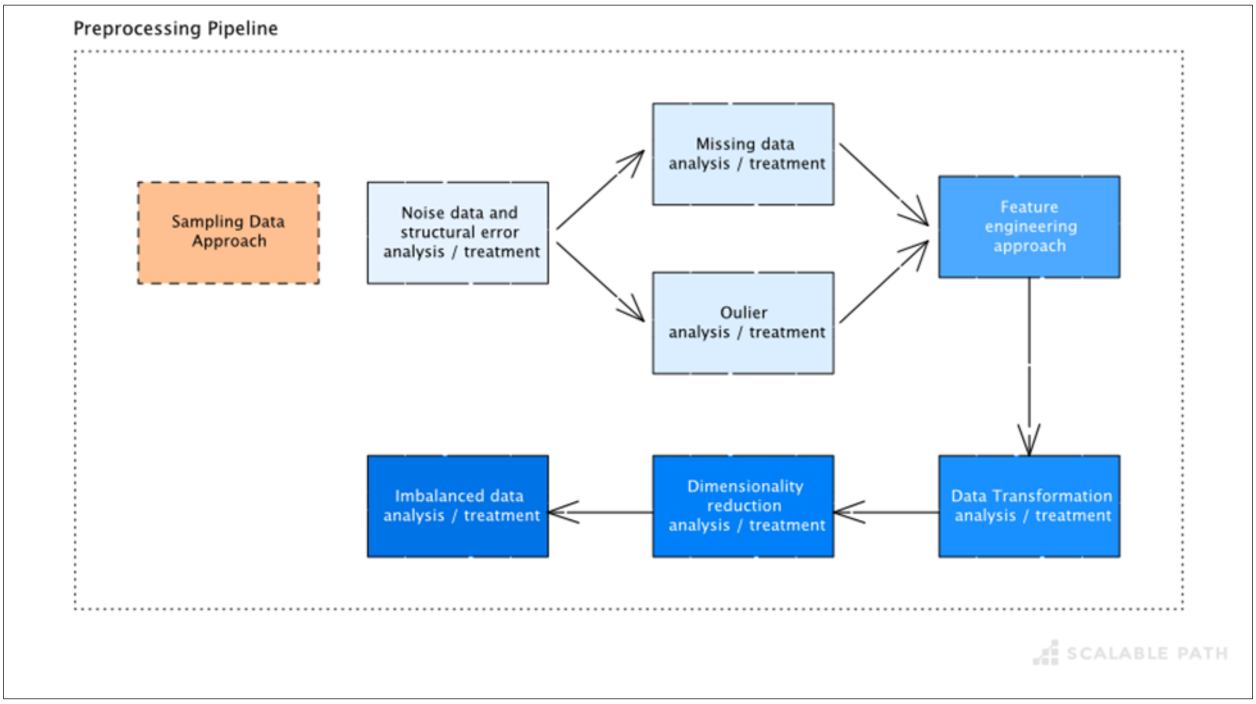
- **Embedded Methods**: The latter include the so-called embedded methods, which are designed to select the features concurrently to the model training process. In general, regularization methods such as Lasso and Ridge regression reduce contribution of less important features while training model, therefore, performs feature selection processo. These methods work with reasonable accuracy as well as reasonable computational time [17].

3.5. Data Preprocessing Techniques

Preprocessing of data makes the data set fully ready for training the ML models because it solves some of the big problems like missing values, outliers, or inconsistent formatting of data.

- **Data Cleaning**: Handling of missing values is done through imputation either through mean, median or mode estimate. Outliers are dealt with using statistical tools, such as Z-scores or interquartile ranges not to distort the model [18].
- **Normalization and Standardization**: These techniques level of feature scales to be parity. Normalization scales features onto a given range, usually [0 1], while on the other hand standardization scales feature so that its mean is zero and standard deviation is one merely. These Above steps are especially helpful where feature scaling is critical like – Support Vector Machines and models based on Gradient Descent [14].
- **Dimensionality Reduction**: Data dimensionality is reduced by most of its variance using techniques such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). This is not only efficient in terms of a reduced amount of calculations needed to be made but also minimizes the problem of overfitting [18].

This way, using a proper feature selection and data preprocessing, a lot of space for improvements in accuracy and efficiency of ML models can be opened. These steps keep model size minimized, accelerates training time and offers better generalization on unseen data which is the basis of reliable and scalable solution in ML [16][17][18].



Adapted from ScalablePath

Figure 9 Data preprocessing phase

4. Challenges and Solutions in Implementing ML-Based Predictive Maintenance

Introducing the methods of machine learning (ML) for predictive maintenance in IT systems is accompanied by a number of issues which should be solved to enable the proper functioning of predictive maintenance. They include data quality, data availability and data integration as well as model reliability and scalability when making decisions in constantly evolving ecosystems. However, the implementation of ML solutions with the existing environment poses major challenges most of the time. This section elaborates on these hurdles offering more information on data issues, models, and systems integration. This work also describes possible solutions for these challenges, including the use of Cloud-based ML platforms and Hybrid solutions for solving PdM problems.

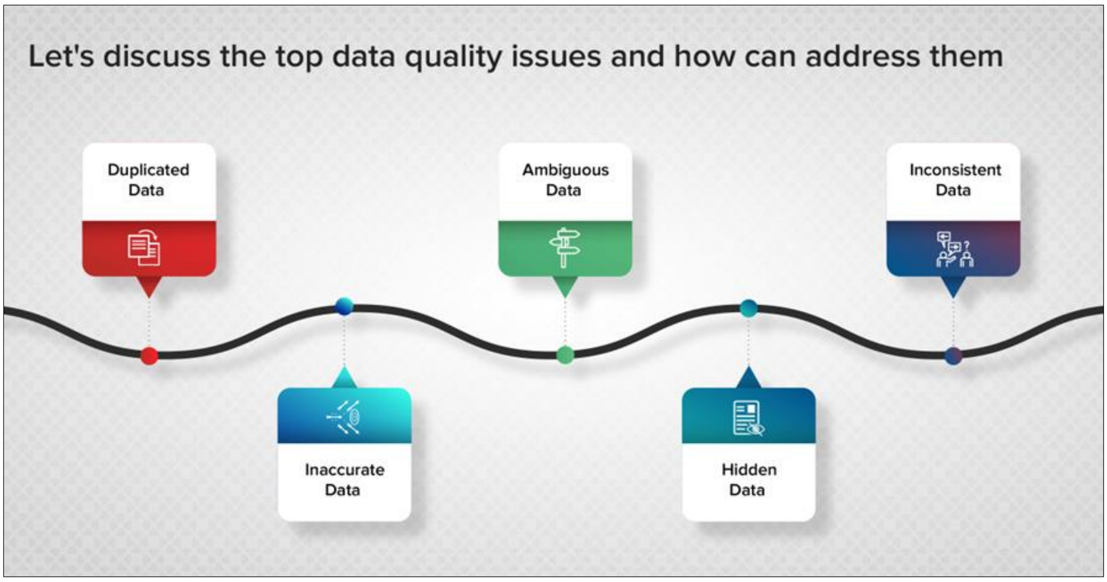
4.1. Data Challenges

There are great data hurdles which are encountered when deploying predictive maintenance systems that are based on ML technologies. The most frequently reported problem is data quality. In order for the predictive maintenance models to be effective they require historical data on the performance of the equipment and previous failures. However, this data is usually unclear, irregular, or contains significant missing information. Faulty data or data which is incomplete is normally detrimental in predicting outcomes resulting in poor maintenance decisions. Data availability adds to this challenge even more, especially in industries where data collection from the machines is either limited or not well checked frequently. In other situations where there is insufficient data to be given to an algorithm and other data they not capable of modeling well and or giving accurate predictions.

Another big factor is how to fuse data together from disparate source systems for subsequent analysis. The information generated for predictive maintenance might be stochastically dispersed and heterogeneous in terms of its nature; it may include sensor data, equipment logs, maintenance records, environmental data, and many others. It is not always easy to bring these various datasets together into one seamless platform for analysis. Often legacy systems or 'normal' databases do not possess the capability to deal with the volume and variety of data that the ML systems afford. This results to slow down learning and decision making processes, and the solutions cannot easily be replicated at scale within an organization.

In this regard however, several solutions can be implemented to address the above challenges. For data quality, more strict data cleaning and preprocessing also commonly employed to eliminate the errors in the data sets. If a number of values in the set are missing, it is possible to use data imputation techniques to delete or get rid of them; if there are observations that are too extreme within the data set, they can be eliminated with the help of anomaly detection techniques as well. For data availability, a number of companies can increase their investments in real time monitoring systems and IoD, so that they track additional, more specific and up-to-date data. Edge computing can also be applied for data analysis to facilitate storage of important data which may be needed when in areas that lack or have restricted internet connection. Last, regarding data integration, businesses can adopt, for instance, a cloud-based solution to integrate a number of dissimilar sources of data, as seen in 5.3, which then makes it possible to feed the right data into the ML models.

Overcoming these data challenges result in better prediction in the area of condition based maintenance, resulting in the following benefits; cost reduction, minimal system downtime and enhanced operational efficiency [19][20].



Adapted from Amurta

Figure 10 Common data quality issues

4.2. Model Reliability and Scalability

It is the question of achieving model reliability in two conditions, as well as the model scalability as the significant factor affecting ML-based PdM success in dynamic IT landscapes. Being able to maintain that the models are fit for purpose is critical to their long term use since the operating environment is bound to change. In evolving contexts like Information Technology, conditions like new material condition, changing workloads and types of data can affect the performance of Machine Learning models. For instance, the models can give good results when tested on the past data; however, the system's efficiency and reliability can deteriorate if the operational environment changes or some new conditions appear. This results in the problem of model robustness whereby looks at the possibility of the model to continue predetermining with stable and accurate results notwithstanding the changes. A possibly less robust model might be incapable of recognizing new patterns of failures or might over- or underestimate the requirement for maintenance and, therefore, could miss potential improvement opportunities. In order to respond to these challenges, it is important to create models which allow for on-going learning processes. These methods can also be used for an incremental learning model and online learning methods are beneficial in the context of updating the models based on the changing system environment. Moreover, an understanding of system behavior from a broader perspective, which is offered, for instance, by ensemble learning, the technique that has multiple models combined in order to make predictions, can also contribute to reliability.

Another factor that complicates the process of deploying ML models in large complex IT environment is scalability. When the number of transactions is high, conventional approaches may not be efficient enough to handle larger data quantities. This mandates the use of efficient algorithms of large scale data that does not impede the performance of the developed system. The distributed computing frameworks like, Apache Spark or TensorFlow used for the parallel processing of data volumes of data and implement the parallel models to match the real time demands of predictive maintenance in large infrastructures. To sum up, developing solutions where models could remain accurate and efficient at the same time, staying productive when conditions are changing, some solutions have to be implemented: the application of the adaptive model learning and using the scalable infrastructure.

4.3. Integration with Legacy Systems

The introduction of ML into current Business Systems is a major issue, even more, so in industries that have lengthy complex structures that were not built to contain contemporary artificial intelligence solutions. Such barriers can prevent effective adoption of predict and maintain solutions and other innovations based on ML. The first one is a problem of data integration where data from one system cannot be used in another system. Traditional systems tend to either save information in less suitable structures or else employ separate database management systems, which complicates the extraction, preprocessing, and fusion of the data with the contemporary ML process. Such systems can be missing APIs or normalized data interfaces, therefore requiring interventions or data interpretation and translation. When it comes to feeding legacy systems into the current ML models, it is usually challenging to do that in real-time because of the absence of integration tools to streamline the process. One more key concern is the technical debt that links to such a legacy legacy misconfiguration infrastructure. These were developed using older technology software and hardware, which might not adequately support the computations needed in current practice of ML algorithms. The scalability of the architecture to accommodate the resource-consumption nature of many ML models can mean heavy investment in hardware or deploying in the cloud. Furthermore, there can be a shortage of qualified staffing to implement AI and to understand the older systems and newer technologies simultaneously to integrate both integrations.

There are also two issues related to system stability and risk management that give the industry serious challenges. Indeed, in many cases today, corporations that operate in sectors such as energy, finance, and healthcare cannot pay the price in production losses due to systems overhauls or intricate integration projects. The integration of the gross existing systems with the modern ML models may cause new types of risks connected with data security, compliance, or stable operation. Nevertheless, hybrid architectures and cloud-based solutions seem to be the suitable vehicle for integration. Thus, using cloud platforms, organizations can execute ML models as a third-party solution while retaining a traditional infrastructure, which significantly reduces risk and guarantees effective interaction between current and previous frameworks. Thus, in order to implement ML systems into enterprise-level legacies, the following key considerations have to be taken into account: data compatibility issues, technological debt, and possibilities in terms of further implementing advanced AI features to the current infrastructure.

4.4. Proposed Solutions

In order to address the problems related to the application of machine learning solutions in the traditional IT environment some general and hybrid approaches are the following. Cloud application platforms are flexible, adaptable and economic for hosting and executing of ML models. These platforms reduce the pressure on the members of an organization to purchase expensive computer equipment to host the computation as the cloud infrastructure varies depending on the computational needs of the group. Some of today's well-developed cloud service platforms, which include AWS, Azure, Google Cloud, include powerful built-in ML tools that enable the integration with the current conventional systems and thus support data storage, preprocessing, and training of models.

Another good solution is built-in infrastructure and outsourced resources, which is half and half between on-premises and cloud. With paying special attention to the need for facilities for data handling and model training on the cloud side while keeping old structures for regular activities, organizations can obtain a clear and effective transition. This approach is less invasive and greatly limits the chance of interfering with ongoing business activities. It also gives flexibility for a business organization to phase change its conventional systems to more advanced systems with cloud structures. All of these proposed solutions do not only make the adoption of ML models easier but also confirm that organizations may advance their levels of effective preventive maintenance without encountering major barriers on data integration or system compatibility, or incur high initial costs.

5. Real-World Applications and Case Studies

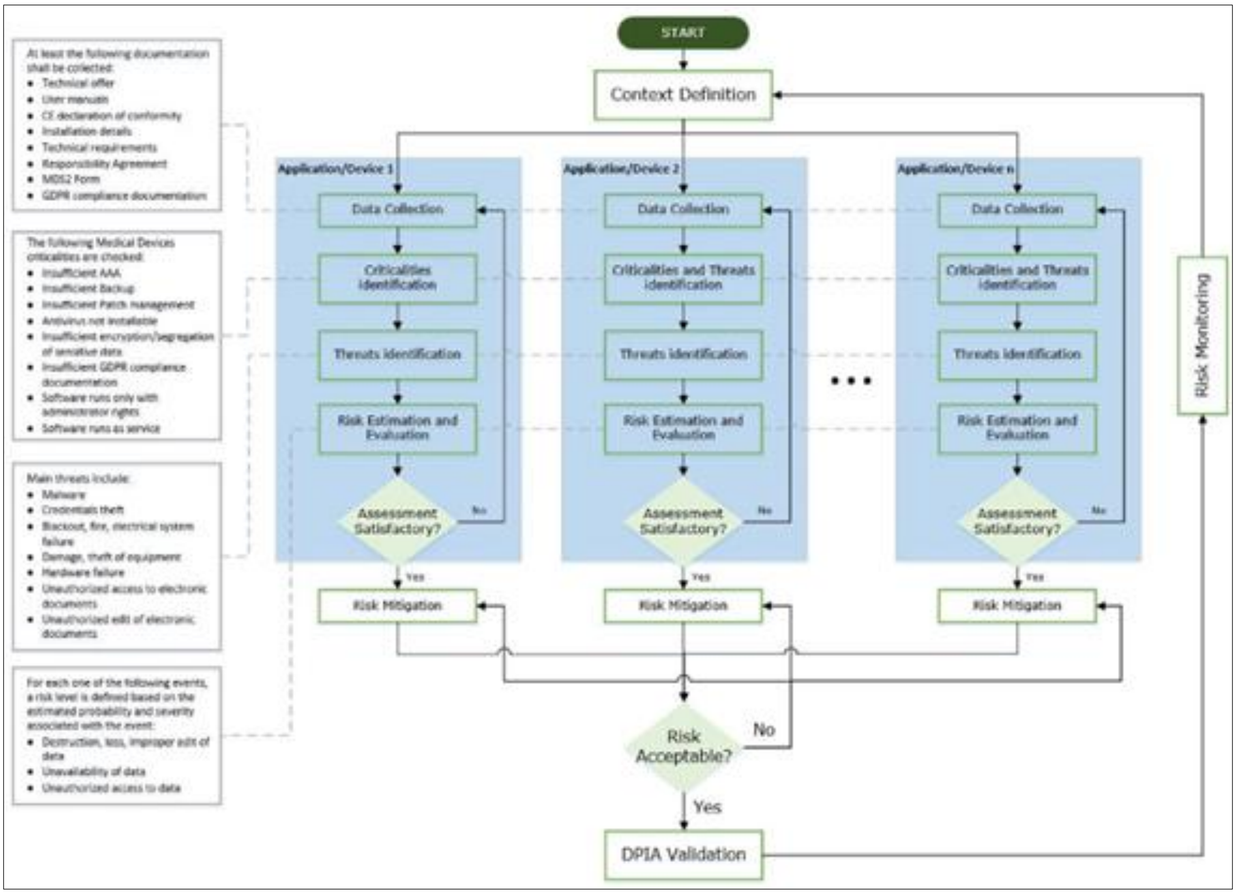
The concept of ML is no longer a theory anymore and it has become an envy of innovation in today's world where technology is changing with a speed of light. High-risk industries ranging from manufacturing to healthcare are revolutionized by the presence of ML: predictive maintenance is now possible. In this chapter, the author aims to present examples of ML as applied to solving real-life problems as well as demonstrate how it may look like when it unfolds to address the most disparate issues, transform the business environment, and activate new possibilities. Studying these particular cases will help us recall what real ML contribution in various fields looks like.

5.1. Healthcare Industry

The technology of machine learning has dramatically changed the healthcare industry, especially the issue of system reliability for significant patient information. Since the increasing use of electronic health records, wearable devices, and better diagnostic tools, the data reliability is the main concern for patient safety and throughout the entire care continuum. This task is somewhat simplified by the use of ML because it increases the credibility of data, identifies any deviations, and provides data for real-time decisions [38][39]. One of the simplest applications of ML into healthcare is to improve the dependability and precision of EHR systems. Electronic health records or their acronym EHRs are a multifaceted tool containing patient history, diagnostics, and treatment plans. Problems, such as inaccuracies, in this data can result in wrong diagnoses, early treatment, or sometimes death. Automated machine learning algorithms enable the identification of deviations that exist from a high level of normality in EHRs allowing for error detection and rectification. For example, in natural language processing (NLP) methods, unstructured clinical notes are utilized to check the correctness of structured data and maintain accuracy in the patient's record [23].

Wearable health devices produce metrics in real–time that provide information regarding the state of the patient. However, this data is usually dirty which means that it contains noise, the entries are incomplete or some of them are outliers. In the integration of wearables into the healthcare systems ML models are used for data cleansing and elimination of noise from the data collected from wearables. Methods like exploratory data analysis and streaming allow clinicians to recognize essential changes in the patient's condition quickly; which is very beneficial for chronic disease and home monitoring [21]. It also increases system reliability through predictive analysis that is embedded in the ML algorithm. A common clinical issue that can be identified in healthcare systems is the problem of equipment breakdown and work processes disruptions. Maintenance prediction models based on ML are used to study the operational data generated by the medical instruments to identify when a device is likely to develop a problem. This preemptive measure helps to avoid long breakdown times, lower the cost of repair, and guarantee that essential apparatus, including ventilators and imaging equipment, will be always ready to use [22]. Moreover, cybersecurity applications based on ML safeguard sensitive and vital patient information against breaches and other unauthorized access. Such activities are viewed by advanced ML algorithms of the network traffic, as well as signs of a potential cyber attack. This capability is crucial in the preservation of health care information security, and HIPAA among other regulations [23]. Real-life examples can help to understand how ML helps in healthcare. For instance, an AI platform used in a top-tier hospital system decreased the inaccuracy rate of EHR by 40% concerning procedural data reliability. Another success story can be realized in a wearable health monitoring system that adopted ML algorithms to get 98% accuracy of arrhythmias for timely interferences [22][23]. However, the introduction of the MLS styled system for use in the health care setting, has its drawbacks as follows. To come up accurate models, quality data is essential and in the healthcare sectors they are faced with data with gaps, and fragmented data. Furthermore, by the interpretability issue, clinicians must be confident as well as be capable to comprehend the output calculated by the ML system. All these challenges require cooperation between data scientists, health care workers and decision makers [77].

Therefore, healthcare industry is benefitting from ML by guaranteeing the accuracy of crucial patient information. With advanced analytics, anomaly detection and predictive maintenance, ML optimises the precision, security and functioning of healthcare systems. As the use of ML is added to the systems, it promises to enhance the quality of care for patients and reliability for the systems as well as opening up the prospects for building smarter and safer healthcare system[22][23].



Adapted from ScienceDirect

Figure 11 Data quality assessment framework

5.2. Cloud and SaaS Providers: Managing Uptime and Performance in Large-Scale SaaS Systems

The cloud computing has brought a great turning point on how software application are deployed and being used especially through SaaS. SaaS providers are being expected to operate programs that would serve a growing customer base, across multiple geographies, and across multiple service levels. The uptime and performance are important concerns to the SaaS providers since even slight interruption in the service delivery cycle results in customer loss, stain on the company's image and loss making. Management of these factors is therefore critical in retaining competitiveness and lastly the customer confidence.

Uptime is the measure of how much percent of the time, a particular system is up and running and is completely available to the public in case of SaaS applications. Minor service disruption can cause significant adverse effects on the provider and the service consumer who relies on it for business-critical work. To ensure the availability of the subscribed service at a premium level SaaS providers have put in place number of measures that include second level servers, balancing algorithms and online monitoring tools. In distributed environments, workload is spread over several servers of different geographical location, hence they can overcome challenges related to server crashing or very high traffic situations [24]. The operational trend of SaaS platforms is becoming complex and therefore requires the enhancement of robust cloud management tools. These tools enable the providers to keep track of request response rates, delay, and failure rates among others. In real-time data analysis and alerting engines are applied to identify bottlenecks and degradation in service delivery to enable the providers to make prompt corrections before the situation becomes worse. Another important application area is predictive analytics, implementing machine learning algorithms, which is also growing popular in finding performance problems when they're still potential, which, in its turn, provided stronger support for managing them in advance and decreasing the chance of performance degradation [25].

Another major importance that SaaS providers strictly follow is scalability; an element that contributes immensely to performance and system availability. With the increasing population of users over time SaaS applications have to deal with higher traffic and number of transactions without losses in the quality of service provided. That is, while cloud platforms themselves are inherently scalable, SaaS providers must minimize overhead and efficiently scale their cloud environments in an effort to achieve cost optimization. These include methods of how the SaaS provider can allocate the required resources depending on how the traffic flows; thereby make it scalable in that the demand during high traffic is addressed and the demand during low traffic is not wastefully catered for [24]. Apart from uptime and performance, the SaaS provider must ensure he/she meets disaster recovery and data integrity issues. Cloud infrastructures are always dependable in a natural way; however, account loss, degeneration, or violation can still happen. To avert such mishaps, most SaaS companies employ robust backup solutions and retain data mirroring centers in different geographical regions in the event that something goes wrong. Furthermore, systematic tests are performed at set intervals to confirm the validity of disaster recovery strategies, and to eliminate legal and regulatory risks consistent with compliance with industry standards [26]. One of the major issues in Saas industry is nature and vast heterogeneity of the end-User environment. Since the customers can use the SaaS applications through the devices, operating systems, and networks, it becomes critical for the SaaS providers to deliver high quality of service irrespective of the environment. This is made do through proper systems design, incorporation of Shear hosting architectures and a very thorough testing to gain compatibility over the many environments. Therefore, it can be seen that overall uptime and service performance management in large scale SaaS systems can be very complex. Therefore, by using the efficient functions of the cloud infrastructure, proper monitoring and scaling, as well as predictive analysis, SaaS providers are able to consider their platforms as reliable and performant. This is logical as cloud consumption will persist to rise, and customers' expectations for consistently reliable service and high performance will heavily impact the success of SaaS vendors [24][25][26].

6. Conclusion

6.1. Key Takeaways

The continuous monitoring ad forecasting of system condition is vital in the formation of self-healing software applications, where system failures are detected ahead of time and addressed consequently to minimize disruption of service. Through the use of machine learning and real time analytics, predictive maintenance enables systems to detect areas of likely failure before such failures occur to full scale. This cuts on time required to allow users to wait during maintenance, increases the reliability on the tools, and enhances the user experience. Amarthya self-organizing capability of software systems to diagnose, prognosis, and perform remedial measures without any human intervention is revolutionary incorporation, especially in reliability-sensitive contexts such as high-availability systems. In self-healing systems, the early warning redundant structure is substituted by the predictive maintenance function. Applying artificial intelligence helps find the existing trends, patterns, and abnormality that may suggest development of problems. "All these predictive measures provide the system with opportunity to take appropriate action such as redirecting the traffic, reallocating resources or altering the system parameters to make sure that the system does not fail." It also somehow assists towards the reduction of its maintenance costs and the general efficiency and durability of the software systems.

However, in self-healing software, self-predictive maintenance goes beyond failure detection. As is described below it opens up opportunities for the highly effective dynamic approach to the management of the system where the software is capable of adjusting control actions to varying circumstances and thus enhances the system's stability during usage. With a growing trend in the use of cloud services and advanced structures in organizations, predictive maintenance becomes a crucial element of self-repair to maintain operating effectiveness, reduce the incidence and reliance on human input as well as guaranteeing continuity of service delivery. In conclusion, predictive maintenance is critical to self-healing systems and hence crucial a method of ensuring that you effectively manage software applications' performance and risks at scale. Its effectiveness in preventing the occurrence of hazards before they gain out-of-control status make it possible for organizations to maintain system-high availability and reliability, while at the same maximizing resource productivity and minimizing operational costs.

6.2. Future Directions

This coming evolution of self-healing technologies will transform ways through which organizations address their software systems. With new improvements of artificial intelligence and machine learning we are likely to see even more intelligent and independent self-healing systems in the near future. When deep integration of AI decision making with edge computing and real time data processing will be applied, systems which are able to predict and avoid failure would be combined with the systems that self-optimize on their own without the involvement of human operators. In the future, we are likely to witness improvements in the design of intelligent systems, which gets better at handling each failure in an incident in order to become more robust. Increasingly complex environments will also be managed by autonomous systems with much greater comfort to the end users. They said predictive maintenance must advance and adapt to provide finer detail of how IoT gadgets and cloud systems as well as distributed architectures are functioning. Also, the progress realized in the sphere of quantum computing might lead to the appearence of the new opportunities in the sphere of real-time analysis and decision making on the basis of which the self-healing system has to operate even more effectively.

Self healing systems are expected to take a cognitive turn by developing completely autonomous environment with very little or virtually no supervision for operations in the years to come will be comparatively cheaper and more secure to operate. These systems incorporated into various sectors of society, including health care and finance industries, will spur advance and indicate new distinctiveness for running organizations and resisting disruptions.

References

[1] Schneider, C., Barker, A., & Dobson, S. (2015). A survey of self-healing systems frameworks. Software: Practice and Experience, 45(10), 1375–1398. https://doi.org/10.1002/spe.2303

[2] Ghahremani, S., & Giese, H. (2020). Evaluation of self-healing systems: An analysis of the state-of-the-art and required improvements. Computers, 9(1), 16. https://doi.org/10.3390/computers9010016

[3] Putha, S. (2021). AI-driven predictive maintenance for smart manufacturing: Enhancing equipment reliability and reducing downtime. Journal of Deep Learning in Genomic Data Analysis, 2(1), 160–203. https://doi.org/10.1234/jdlgda.2022.0160

[4] Ki, Y., & Song, M. (2009). An open source-based approach to software development infrastructures. In 2009 IEEE/ACM International Conference on Automated Software Engineering (pp. 525–529). IEEE. https://doi.org/10.1109/ASE.2009.67

[5] Turner, S. (2013). Benefits and risks of cloud computing. Journal of Technology Research, 4, 1. https://doi.org/10.1234/jtr.2013.001

[6] Ran, Y., Zhou, X., Lin, P., Wen, Y., & Deng, R. (2019). A survey of predictive maintenance: Systems, purposes, and approaches. arXiv Preprint, arXiv:1912.07383, 1–36. https://arxiv.org/abs/1912.07383

[7] Gayialis, S. P., Kechagias, E. P., Konstantakopoulos, G. D., & Papadopoulos, G. A. (2022). A predictive maintenance system for reverse supply chain operations. Logistics, 6(1), 4. https://doi.org/10.3390/logistics6010004

[8] Li, H., Sun, J., & Wu, J. (2010). Predicting business failure using classification and regression tree: An empirical comparison with popular classical statistical methods and top classification mining methods. Expert Systems with Applications, 37(8), 5895–5904. https://doi.org/10.1016/j.eswa.2010.02.016

[9] Li, J., Ji, X., Jia, Y., Zhu, B., Wang, G., Li, Z., & Liu, X. (2014, June). Hard drive failure prediction using classification and regression trees. In 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (pp. 383–394). IEEE. https://doi.org/10.1109/DSN.2014.43

[10] Shcherbatov, I., Lisin, E., Rogalev, A., Tsurikov, G., Dvořák, M., & Strielkowski, W. (2021). Power equipment defects prediction based on the joint solution of classification and regression problems using machine learning methods. Electronics, 10(24), 3145. https://doi.org/10.3390/electronics10243145

[11] Syarif, I., Prugel-Bennett, A., & Wills, G. (2012). Unsupervised clustering approach for network anomaly detection. In Networked Digital Technologies: 4th International Conference, NDT 2012, Dubai, UAE, April 24–26, 2012. Proceedings, Part I (pp. 135–145). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-30507-8_12

[12] Pu, G., Wang, L., Shen, J., & Dong, F. (2020). A hybrid unsupervised clustering-based anomaly detection method. Tsinghua Science and Technology, 26(2), 146–153. https://doi.org/10.26599/TST.2019.9010035

[13] Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2012). An effective unsupervised network anomaly detection method. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics (pp. 533–539). https://doi.org/10.1145/2345396.2345475

[14] Ahmed, N., Hossain, M. E., Rishad, S. S. I., Mohiuddin, A. B., Sarkar, M. I., & Hossain, Z. (2020). Leveraging reinforcement learning for autonomous cloud management and self-healing systems. JURIHUM: Jurnal Inovasi Dan Humaniora, 1(6), 678–689.

[15] Ma, T., Ali, S., & Yue, T. (2021). Testing self-healing cyber-physical systems under uncertainty with reinforcement learning: An empirical study. Empirical Software Engineering, 26, 1–54. https://doi.org/10.1007/s10664-021-09947-4

[16] Alabdulwahab, S., & Moon, B. (2020). Feature selection methods simultaneously improve the detection accuracy and model building time of machine learning classifiers. Symmetry, 12(9), 1424. https://doi.org/10.3390/sym12091424

[17] Khalid, S., Khalil, T., & Nasreen, S. (2014, August). A survey of feature selection and feature extraction techniques in machine learning. In 2014 Science and Information Conference (pp. 372–378). IEEE. https://doi.org/10.1109/SAI.2014.6918280

[18] Sami, O., Elsheikh, Y., & Almasalha, F. (2021). The role of data pre-processing techniques in improving machine learning accuracy for predicting coronary heart disease. International Journal of Advanced Computer Science and Applications, 12(6). https://doi.org/10.14569/IJACSA.2021.0120601

[19] Bhaskaran, S. V. (2020). Integrating Data Quality Services (DQS) in Big Data Ecosystems: Challenges, Best Practices, and Opportunities for Decision-Making. Journal of Applied Big Data Analytics, Decision-Making, and Predictive Modelling Systems, 4(11), 1–12. Retrieved from https://polarpublications.com/index.php/JABADP/article/download/4/4

[20] Lwakatare, L. E., Raj, A., Crnkovic, I., Bosch, J., & Olsson, H. H. (2020). Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions. Information and Software Technology, 127, 106368. https://doi.org/10.1016/j.infsof.2020.106368

[21] Chassin, M. R., & Loeb, J. M. (2013). High-reliability health care: Getting there from here. The Milbank Quarterly, 91(3), 459–490. https://doi.org/10.1111/1468-0009.12023

[22] Arour, M., Alenezi, M., Ansari, M. T. J., Pandey, A. K., Ahmad, M., Agrawal, A., ... & Khan, R. A. (2021). Ensuring data integrity of healthcare information in the era of digital health. Healthcare Technology Letters, 8(3), 66–77. https://doi.org/10.1049/htl2.12018

[23] Abouelmehdi, K., Beni-Hessane, A., & Khaloufi, H. (2018). Big healthcare data: Preserving security and privacy. Journal of Big Data, 5(1), 1–18. https://doi.org/10.1186/s40537-017-0110-7

[24] Gao, J., Bai, X., Tsai, W. T., & Uehara, T. (2013, March). SaaS testing on clouds: Issues, challenges, and needs. In 2013 IEEE Seventh International Symposium on Service-Oriented System Engineering (pp. 409–415). IEEE. https://doi.org/10.1109/SOSE.2013.79

[25] Ghahramani, M. H., Zhou, M., & Hon, C. T. (2017). Toward cloud computing QoS architecture: Analysis of cloud systems and cloud services. IEEE/CAA Journal of Automatica Sinica, 4(1), 6–18. https://doi.org/10.1109/JAS.2017.7510349

[26] Simic, V., Stojanovic, B., & Ivanovic, M. (2019). Optimizing the performance of optimization in the cloud environment—An intelligent auto-scaling approach. Future Generation Computer Systems, 101, 909–920. https://doi.org/10.1016/j.future.2019.07.052

[27] ResearchGate. (n.d.). Software development methods timeline . Retrieved from https://www.researchgate.net/figure/Software-development-methods-timeline_fig3_275654650

[28] ResearchGate. (n.d.). Predictive maintenance cost model. Retrieved from https://www.researchgate.net/figure/Predictive-maintenance-cost-model_fig1_320441136

[29] Easterbrook, S. (2009). Growth of code complexity . Retrieved from https://www.easterbrook.ca/steve/2009/07/growth-of-code-complexity/

[30] ResearchGate. (n.d.). Overall predictive maintenance process and framework. [Figure 4] Retrieved from https://www.researchgate.net/figure/Overall-predictive-maintenance-process-and-framework_fig3_354258052

[31] FasterCapital. (n.d.). The benefits of predictive maintenance. Retrieved from https://fastercapital.com/topics/the-benefits-of-predictive-maintenance.html

[32] Medium. (n.d.). Equipment failure prediction using machine learning models. Retrieved from https://yavisankar.medium.com/equipment-failure-prediction-using-machine-learning-models-8be0206eed6b

[33] ResearchGate. (n.d.). Scatter plot showing the result of machine learning clustering. Retrieved [33] from https://www.researchgate.net/figure/Scatter-plot-showing-the-result-of-machine-learning-clustering-Unsupervised-machine_fig2_346456508

[34] ResearchGate. (n.d.). Flowchart of the reinforcement learning method. Retrieved from https://www.researchgate.net/figure/Flowchart-of-the-reinforcement-learning-method_fig2_344845432

[35] ScalablePath. (n.d.). Data preprocessing phase. Retrieved from https://www.scalablepath.com/data-science/data-preprocessing-phase

[36] Amurta. (n.d.). Top 5 most common data quality issues. Retrieved from https://www.amurta.com/blogs/top-5-most-common-data-quality-issues

[37] ScienceDirect. (n.d.). Data quality assessment framework. Retrieved from https://www.sciencedirect.com/science/article/pii/S235291

[38] Chandrashekar, K., & Jangampet, V. D. (2020). RISK-BASED ALERTING IN SIEM ENTERPRISE SECURITY: ENHANCING ATTACK SCENARIO MONITORING THROUGH ADAPTIVE RISK SCORING. INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING AND TECHNOLOGY (IJCET), 11(2), 75-85.

[39] Chandrashekar, K., & Jangampet, V. D. (2019). HONEYPOTS AS A PROACTIVE DEFENSE: A COMPARATIVE ANALYSIS WITH TRADITIONAL ANOMALY DETECTION IN MODERN CYBERSECURITY. INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING AND TECHNOLOGY (IJCET), 10(5), 211-221.

[40] Eemani, A. A Comprehensive Review on Network Security Tools. Journal of Advances in Science and Technology, 11.

[41] Eemani, A. (2019). Network Optimization and Evolution to Bigdata Analytics Techniques. International Journal of Innovative Research in Science, Engineering and Technology, 8(1).

[42] Eemani, A. (2018). Future Trends, Current Developments in Network Security and Need for Key Management in Cloud. International Journal of Innovative Research in Computer and Communication Engineering, 6(10).

[43] Eemani, A. (2019). A Study on The Usage of Deep Learning in Artificial Intelligence and Big Data. International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), 5(6).

[44] Nagelli, A., & Yadav, N. K. Efficiency Unveiled: Comparative Analysis of Load Balancing Algorithms in Cloud Environments. International Journal of Information Technology and Management, 18(2).