VTT Technical Research Centre of Finland

Machine learning in safety critical industry domains

Linnosmaa, Joonas; Tikka, Petri; Suomalainen, Jani; Papakonstantinou, Nikolaos

Published: 20/10/2020

Document Version
Publisher's final version

RESEARCH REPORT

VTT-R-01124-20

Machine learning in safety critical industry domains

Authors: Joonas Linnosmaa, Petri Tikka, Jani Suomalainen, Nikolaos Papakonstantinou

Confidentiality: Public

VTT

Report's title Machine learning in safety critical industry domains	
Customer, contact person, address SAFIR2022 programme, Steering Group 1, Jari Hämäläinen	**Order reference**
Project name Admire_22_2020	**Project number/Short name** 125844-1.3
Author(s) Joonas Linnosmaa, Petri Tikka, Jani Suomalainen, Nikolaos Papakonstantinou	**Pages** 28
Keywords machine learning, artificial intelligence, safety critical industry, nuclear power	**Report identification code** VTT-R-01124-20

Summary

This report is an introduction to machine learning, and it focuses on topics that can be seen as challenging on safety critical domains such as nuclear power industry. The requirements for designing, developing and testing a safety-critical system for operation are stricter and more regulated than in many conventional domains because of the high emphasis on correct and predictable system behaviour for the protection of the public. While machine learning techniques and the science behind them is developing in an unparalleled fast pace, there are still many fundamental challenges that need special attention when they are to be used in safety-critical context, where there are no room for errors.

In the report, the three main machine learning paradigms are briefly presented and the basic steps in their use are explained. Then, the use of machine learning components in safety critical systems and industrial domains is discussed, while highlighting different fundamental properties and challenges. Finally, it goes through some of the most interesting machine learning topics that were identified in informal discussions with Finnish nuclear stakeholders as potential current or future directions for research and development.

There is clear motivation in nuclear safety engineering to look for and carefully adopt technologies that can improve safety in a proven and measurable manner. The authors, based on this study and the discussions, recognize the growing interest to utilize artificial intelligence based systems to support the everyday work of engineers over the different lifecycle phases of plant design and operation. Topics, which raised most interest were related to using Natural Language Processing for document and requirements, different predictive maintenance topics and supporting the work of operators in control rooms.

Confidentiality	Public

Tampere 20.10.2020

Written by	**Reviewed by**	**Accepted by**
Joonas Linnosmaa	*Emmanuel Ory*	
Joonas Linnosmaa, Research Scientist	Emmanuel Ory, Research Team Leader	Janne Järvinen, Vice President, Data-driven solutions

VTT's contact address
VTT Technical Research Centre of Finland Ltd, P.O. Box 1000, FI-02044 VTT, Finland

Distribution (customer and VTT)

SAFIR2022 Programme
VTT Archives

Preface

This report is a deliverable of a project proposal called DEFLECT – machine learning for fault identification, however it was scoped as a small study to take a wider view into machine learning and its potential use in safety critical domains, especially in nuclear, and to include discussions with Finnish nuclear stakeholders about the topic in general. It was funded under The Finnish Research Programme on Nuclear Power Plant Safety 2019-2022 (SAFIR2022) as ordered by its Steering Group 1. It is meant as an introduction and a discussion opener for the SAFIR Programme members towards utilizing more data driven methods for nuclear safety-related applications. We try to give the readers basic understanding relating to the industrials use of machine learning methods.

The authors would like to thank Fortum, STUK, TVO and Fennovoima for their support and their participation to the discussions during the project and for sharing their valuable domain experience and feedback.

Tampere 20.10.2020

Authors

Contents

1 Introduction

Machine learning is a method of data-analysis, which tries to automate the process of learning from the past to gain real-time and accurate knowledge to solve problems of today. Businesses and engineering sectors all over the world are trying to utilize the benefits from the swift and revolutionary progress this field has made in the recent years, to gain advantage in the ever-increasingly competitive and technical market. Currently, the science of machine learning (or artificial intelligence in general) advances even so rapidly that it is impossible to follow the latest breakthroughs, and even more challenging to exploit them. Recent studies have shown that companies struggle to productize these data-driven methods without strong practical knowledge and experience. However, these techniques are here to stay and already disrupting many traditional sectors with new, smarter, data-driven ways of making decisions. They have the capability to back-up their decisions using databases with dozens of years of data about successes, failures and different scenarios encountered through the history, which they can often process in a flash compared to their human counterparts. When machine learning based system are being deployed into real production, they face new mature problems arising from the industrial applications such as explainability, uncertainty, reproducibility, correctness, security and privacy to mention just a few.

1.1 Goals and process of the study

This report is an introduction and short literature review on the use of artificial intelligence, especially machine learning techniques, to implement, guide or supervise critical control actions and mission critical decision making. This report should give the reader a better picture of the status of machine learning, together with an assessment of the present-day applicability of its techniques in the safety critical domains. This short report will not extensively cover the details of the methods or the theoretic background of machine learning as a science. Instead, it aims to introduce the relevant concepts, identify the challenges and put the applications of machine learning in safety critical industrial domains into perspective.

The rest of this report is structured as follows: The second chapter introduces the reader to the fundamental ideas of machine learning and its three basic paradigms. The third chapter highlights and discusses properties and challenges related to machine learning that are relevant when thinking about using it for safety critical industrial applications. The main findings of the stakeholder discussions and short introductions to some selected topics are presented in Chapter 4. Finally, Chapter 5 summarizes the conclusions of this small study.

An important part of this report originates in the informal discussions with Finnish nuclear stakeholders (STUK, Fortum, TVO, and Fennovoima) aiming to identify current needs or challenges that may benefit from the use of machine learning or other data-driven methods.

In addition to this small study that focuses on machine learning applications on safety engineering over the lifecycle of safety-critical systems, the SAFIR Steering Group 1 has ordered another small study focusing on the use of artificial intelligence as a regulator decision support technology. These reports will be published at the same time. There is a minor overlap between these two reports since machine learning is part of the artificial intelligence family of tools and the same basic principles apply to both. However, this report tries to focus more on industrial applications of machine learning while the other report will focus more on regulatory use of artificial intelligence.

1.2 Terms and abbreviations

Table 1. Abbreviations

Abbreviation	Description
AI	Artificial Intelligence
DL	Deep Learning
FTA	Fault Tree Analysis
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
MBSE	Model Based Systems Engineering
ML	Machine Learning
NLP	Natural Language Processing
NPP	Nuclear Power Plant
RL	Reinforcement Learning
SE	Systems Engineering
V&V	Verification & Validation

2 About machine learning

“Machine Learning (ML) is the art and science of letting computers learn without being explicitly programmed” [1]. The concept of machine learning is based on the development of a model by a computer of some abstract principle from data alone and applying this gained “knowledge” to another, yet unseen, situation to make predictions. This basic idea of ML has been around at least since the 1950s, when the first neural network models were invented. Even before that, modelling methods like Bayesian statistics and Markov chains have been used to accomplish similar tasks. However, the advent of growth of data availability and computational power, combined with the arrival of novel learning methods has increased the scientific activity as well as several breakthroughs in many scientific areas, including the use and exploitation of different machine learning paradigms [2]. Recent advances are due to deep learning (DL), which is a subfield within ML that can deal with complex model architectures trained with large, often incomprehensive to human, datasets to learn dependencies and form models, which were unattainable in the past because of technical insufficiencies.

The traditional approach of modelling the problem (or fitting a model to a problem) is to specify a model that can describe the observed data, while a more modern, but resource demanding, approach specifies no explicit model, but lets the computer be responsible for identifying associations in the observed data and create the model independently. This approach has led to breakthroughs in many applications where describing a fitting model has been too difficult or impossible for a human (maybe because of sheer amount of data or complicated correlations in parameters), but leads to another kind of challenge: the complexity of the generated model. This so called “black box” problem comes from the high number of free parameters, the correlations/associations between them and the complex interactions of a self-learning model, which tend to make them difficult for humans to interpret logically. However, significant progress has been made over the last years in the interpretability of machine learning models and today many aspects of the old “black box” models can be interpreted using modern tools. [2].

Machine learning is, in its very core, a data analysis tool and selecting correct algorithms is critical for the realization of the desired results. Achieving results depends on the used paradigm, algorithm and data. Commonly, machine learning has been divided into three main domains, or paradigms:

- Classification and regression belonging to supervised machine learning
- Clustering and dimensionality reduction belonging to unsupervised machine learning.
- Additionally, data can be analysed with reinforcement learning, where algorithm processes unlabelled data and learns by trial and error using either value- or model-based learning.

Figure 1, is one way to show this division of functionalities between supervised, unsupervised and reinforcement learning paradigms.

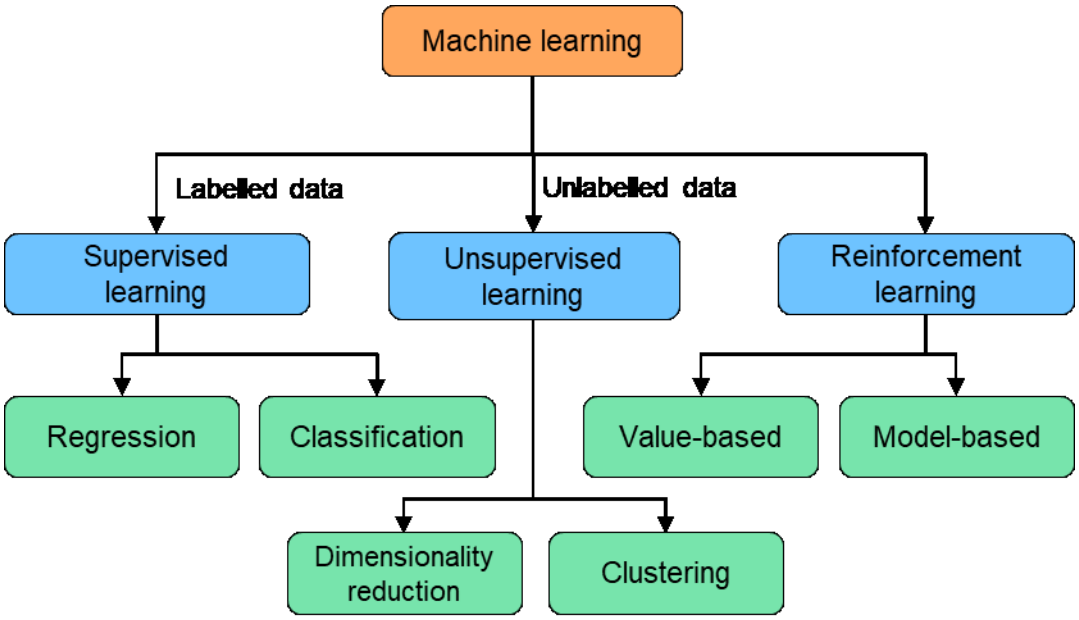

Figure 1. Taxonomy and overview of main ML paradigms and algorithms, modified from [2].

There is common unawareness in terms around machine learning topics, mainly with machine learning (ML), deep learning (DL) and artificial intelligence (AI). The topmost blue branch of Figure 2 tries to convey the hierarchical relation between these terms. Machine learning is only a subset of techniques that are considered to form the domain called artificial intelligence, and deep learning only a subset of machine learning that uses ‘deep neural networks’ [3]. However, machine learning is not an independent application domain within AI, rather a tool for the other domains. In fact, machine learning-based algorithms are used as a learning method in almost all the other main domains of AI too, for example Natural Language Processing (NLP), speech recognition or machine vision.

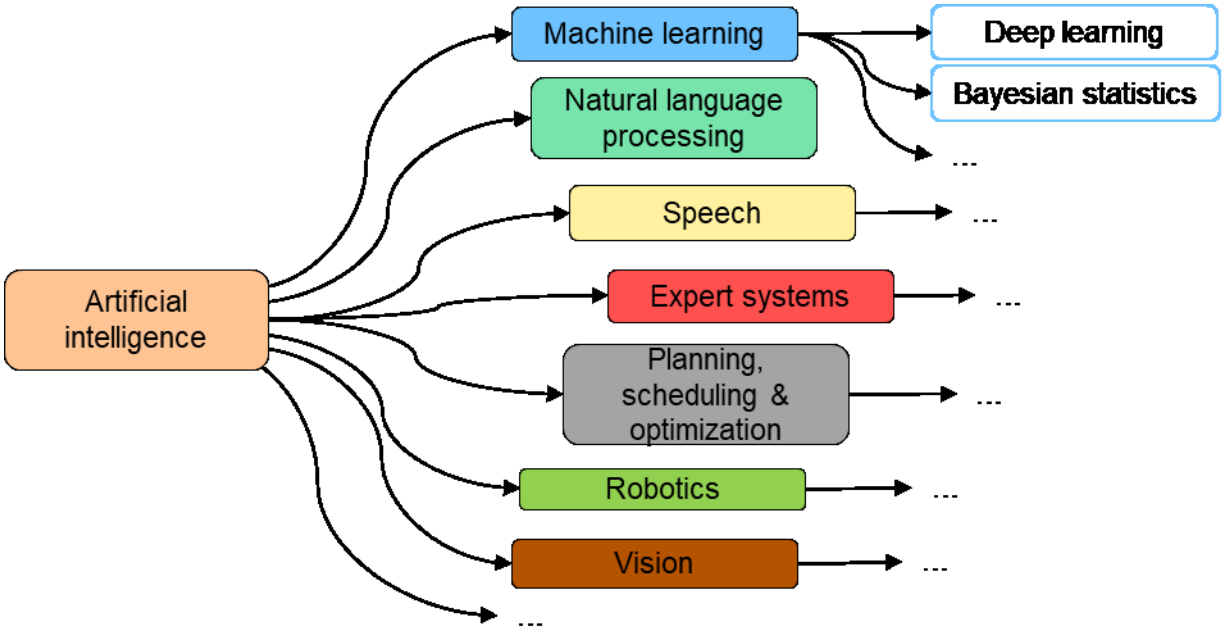

Figure 2. Example domains within artificial intelligence, modified from [4].

Figure 2 also shows how diverse is the field of AI. Usually each domain contains its own field of science with its own domain knowledge, research communities and various techniques used to accomplish tasks within the category. This report, due to its scope, cannot dive deeply in these domains or their domain-specific techniques, there is a wealth of extensive knowledge available from formal (universities etc.) and informal/practical sources. Instead, in the following subchapters, we will introduce to the reader, three main paradigms of machine learning algorithms: namely supervised, unsupervised and reinforcement learning, explaining their key aspects and steps. Almost all the modern machine learning techniques have their roots from one of these paradigms. Each of them can then use deep learning or other more classical algorithms to find their suitable learning information from the training data.

The core idea of all these current data-driven trends (e.g. machine learning, artificial intelligence or big data) is about supporting the user making better, safer and more well-informed decisions based on the data available. Data itself does not lead to better decisions, but it makes possible to analyse the challenge and gain insights that is not easily available otherwise. Analytics can be grouped based on the support they can offer, for example, to describing, diagnosing, predicting and prescribing levels:

- Describing level helps to understand what happened.
- Diagnostic level helps to understand why it happened.
- Predictive level helps to understand what will happen next.
- Prescribing level helps to understand what should be done next.

As the analytic levels mature from simple informative to more advanced and controlling, so does the potential value of support offered by these systems. Unfortunately, similarly does also the amount of skills, techniques and capabilities needed from the organizations wishing to use them. Machine learning system, which are analytical at heart, follow the same scheme. Figure 3 tries to depict this relation of value versus effort of different analytical levels.

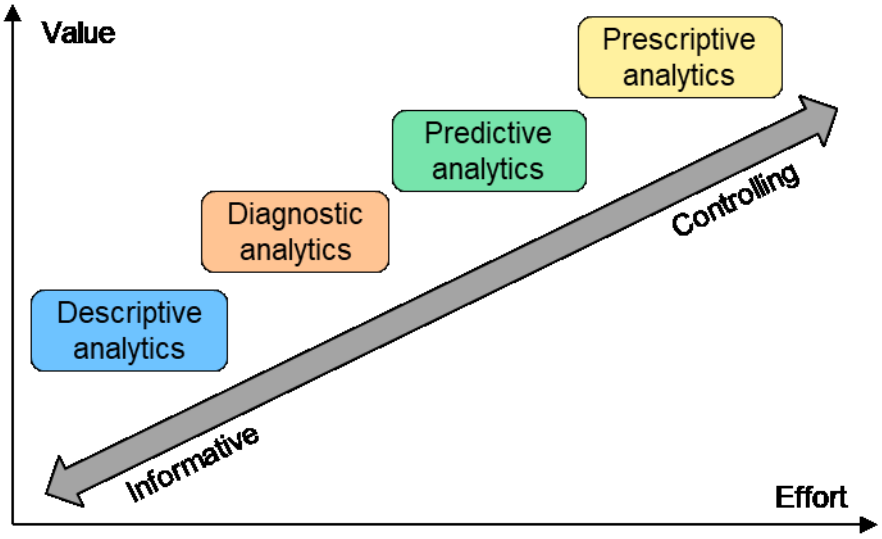

Figure 3. Four levels of analytics, modified from [5]

The higher one moves on the grey arrow, the more value the analytics can offer, but at the same time more effort is needed to gain the full benefits of a correctly working analytical system.

2.1 Supervised learning

Supervised learning in brief

In supervised learning, a computer is tasked to learn how to predict a class or value of a yet unobserved data point based on a concept that has been derived from a training dataset [2]. Prediction target can be numeric values or string labels, depending on the required output and used supervised learning algorithm. Used training datasets need to be labelled, categorized or classified in order to train the algorithm. This is usually one of the most demanding tasks within supervised learning, the labelling of training data. Collecting huge amounts of data can be easy to do under the right circumstances, but, actually, categorizing this data can be very time consuming, it is, however, necessary, if supervised learning is to be used. Many big players in the domain have crowdsourced this labelling effort to large groups of individual people over the internet [6].

Supervised learning is an algorithm, which learns from known datasets and associated target responses. The algorithm, consisting of classification or regression, is objected to learn a mapping function from the input variable to the output variable. The mapping function is approximated to predict as accurately as possible output variables when new input variables are presented. Difference between classification and regression is that classification produces qualitative or discrete output variable while that for regression is numerical or continuous.

Classification algorithm in machine learning tries to predict the value of a single or several conclusions, for instance, into discrete categories: value is over or less than a fixed benchmark. Then again, regression algorithm tries to predict an integer or a floating value, for instance, in quantities: predicted value is the continuation of historical values and hence a continuous output.

Main use areas

Prediction and classification. Supervised learning provides means to determine and predict average values and distinguish values according to a specific classification task. Therefore, supervised learning helps to produce output data according to previous experience. Previous

experience can furthermore be used to optimize the performance of the algorithm. By comparison, supervised learning is less complex to apprehend.

Commonly used classification algorithms include Naive Bayes, decision tree, K Nearest Neighbours and logistic regression. Commonly used regression algorithms include regression trees, Support Vector regression and linear regression.

Basic steps of supervised learning

For the supervised machine learning to be successful, extensive data selection is required. The data set can be compiled from historical data and/or synthetic data (e.g. based on simulation model results). A data processing step can be taken to extract “features” from the data, e.g. calculate statistical values from time series and use these for the further development of the AI system. The data set is labelled, meaning that data entries are associated with a specific state or class (e.g. a time series of sensor measurements is associated to a specific fault in the process). This data set can be split to form a training data set (used for the development of the AI tool) and a testing data set (used to test the success rate of the AI tool). The selection and development of a supervised learning-based algorithm requires considering the structure of the learned function. The algorithm must work accordingly with the given training set to produce the expected output later with the test set. Figure 4 describes these basics steps.

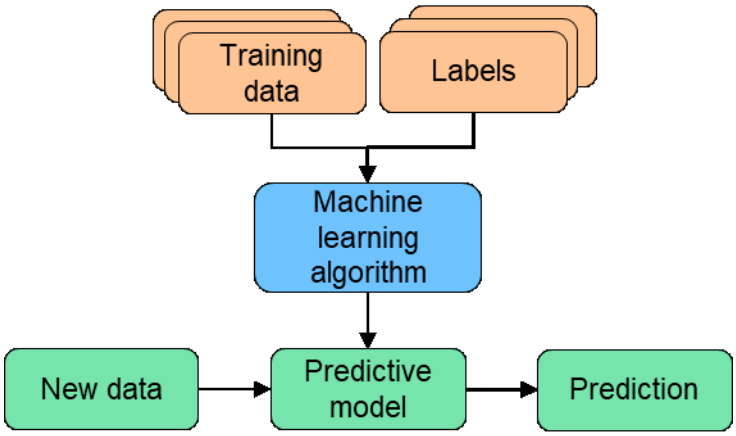

Figure 4. Supervised learning model [7]

Challenges in supervised learning

Supervised learning requires labelled data, which is often either not available or requires preparation and pre-processing. If data is not in-line with the required output, irrelevant features among training set could produce inaccurate results. Therefore, each class should have sufficiently good examples for the classifier to work expectedly. Accuracy problems can also manifest if the training set includes incomplete or improbable values. Learning process with big data requires a lot of computation time making classification a challenging task.

Enough training data, relating to all cases the algorithm needs to handle, is needed, and this data needs to be labelled. For example, fault prediction algorithms need annotated data of the failures it needs to predict, and it would be beneficial, if this data were recorded all the way to failure (depending on the expected prediction function of the AI tool). As can be imagined, in many cases, this kind of data is not available or there are just a few recorded cases. Safety system are designed and built to have a very low frequency of failures, high fidelity simulation models can be used to simulate the response of the process when specific events are triggered and enhance the data set to be used for the development of the AI system.

2.2 Unsupervised learning

Unsupervised learning in brief

Labelling of data sets is often very difficult or impossible. In unsupervised learning, a computer is tasked to identify yet unknown patterns in data without any pre-existing knowledge/labels (like groups or classes) [2]. The training algorithm has the possibility to discover information such as unknown patterns in the data. The unsupervised training algorithm is designed to learn from a known dataset without corresponding target responses thus utilizing unlabelled data. The algorithm restructures this data for the identification of new features that can represent a class or a new series of uncorrelated values. Unsupervised machine learning algorithms are grouped into clustering and association or dimension reduction problems.

The clustering algorithms identify relevant sub-groups as structures or patterns in a collection of uncategorized data without having predefined understanding of a subgroup's property characteristics. The granularity of these subgroups can be adjusted if the number of clusters is known. Clusters are a collection of subgroups of the data, which are similar to one another. There are numerous approaches in clustering for grouping data points by similarity: exclusive partitioning with K-means, agglomerative with Hierarchical clustering, and density-based clustering with Density-Based Spatial clustering.

Association-based algorithms discover associations between data points and reduce the dimensionality of the dataset. Often data includes correlated information, which occurs as unnecessary redundancy, potentially harmful for algorithms performance and training. Dimension reduction decreases the complexity of the data and provides means for avoiding overfitting. Furthermore, uninformative features can be located and removed thus improving the algorithm's performance and convergence time (training time). Dimensionality reduction can be accomplished with feature selection and feature extraction. In this case, information is either selected from the original dataset or derived to formulate new subspace. Furthermore, commonly used dimensionality reduction methods include principle component analysis (PCA), t-distributed stochastic neighbour embedding (tSNE) and uniform manifold approximation and projection (UMAP).

Main use areas

The unsupervised learning algorithms provide insight into the structure and meaning of data. Found patterns can be then utilized as inputs to supervised machine learning algorithms. This technology also provides means for anomaly detection or outlier analysis, which can discover unusual data points from the dataset. Although supervised learning is the most used machine learning paradigm, annotating datasets takes resources and as such, unsupervised learning can be beneficial with potentially locating useful new features for categorization. Additionally, the amount and parameters of classes the data is divided into is not always known, which may enable unsupervised algorithm to search for naturally emerging patterns. This can be beneficial for instance in data mining since clustering automatically splits the datasets into groups of some similar parameters. Unsupervised learning can be also utilized for data pre-processing by either reducing the number of features in a dataset or by decomposing dataset into components.

Basic steps of unsupervised learning

For unsupervised learning to be useful, the dataset has to include related observations. Furthermore, unsupervised learning relies on the assumption that the dataset includes meaningful patterns, which can enlighten new aspects of the data. Collecting diversely relevant data and inspecting the quality is crucial for the algorithm to work as intended. With unsupervised learning, one tries to find underlying structure of a dataset and group it advantageously. Grouping may include reducing dimensions of the data. Whether the dataset is compressed with clustering or by reducing dimensionality, the effective way of representing

the data needs to be considered. The effectiveness of unsupervised learning is observed with metrics that support decision making regarding tuning parameters of the algorithm. However, how well a specific unsupervised learning algorithm performs depends on the context of the desired goal. Hence, metrics can include performance evaluation of clustering or evaluation of probability measures like log-likelihood if the algorithm is probabilistic. Also, by adjusting the parameters of the model according to specific metrics, the accuracy of the algorithm can be increased. Figure 5 shows these basic steps.

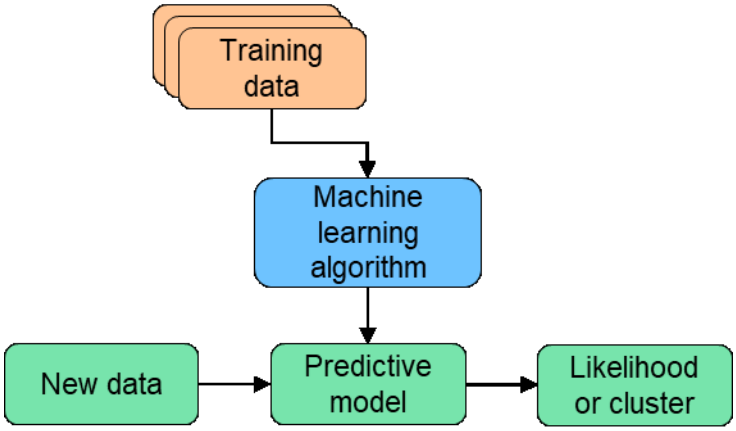

Figure 5. Unsupervised learning model [8]

Challenges in unsupervised learning

Unsupervised machine learning can be more difficult to utilize than supervised machine learning, since the algorithm itself is trying to come up with patterns from the data. This makes the algorithm less accurate as it may or may not find patterns. Because there are no labels to compare with the output, it is left to external evaluation to comprehend what results may be meaningful. Although, unsupervised learning algorithms allow performing more complex processing tasks, the learning process can be unpredictable. Hence, obtaining precise information of data sorting is not possible.

2.3 Reinforcement learning

Reinforcement learning in brief

Reinforcement learning utilizes unlabelled datasets like unsupervised learning. On the other hand, data includes information about the current state of the operative environment with a reward signal, which make reinforcement learning as a field related to supervised learning. Difference to other paradigms is that reinforcement learning proposes deep learnings methods that concern software agents and produced actions in an environment where the agent attempts to achieve maximized cumulative reward through interactions with the environment. During numerous steps, the algorithm learns from feedback to achieve complex objectives or to maximize some specific dimension. The feedback from environment to the agent is not the ground truth label or value, but a measure of how well the action was measured by a reward function. Used algorithms vary according to the decision process and used reward function, but the principle of exploratory trial-and-error or deliberative planning approach is the same.

Main use areas

Reinforcement learning provides the means to locate situations, which require actions. Process-wise, reinforcement learning discovers which actions yield highest reward over a

specific timespan. Algorithms have no supervisor, and they learn according to a specific reward function.

Basic steps of reinforcement learning

There are numerous algorithms to perform reinforcement learning, but a general idea is that the agent tries to maximize the reward by a series of actions in the environment. The reaction of an agent is an interactive action with the environment, and the algorithm policy is a method of selecting an action given a state in expectation of maximizing the outcome. Each state is associated with a positive or a negative reward, which incorporates how the algorithm has accomplished from the overall goals' perspective. “State” refers to the current situation of an agent in an environment. Figure 6 shows these basic steps.

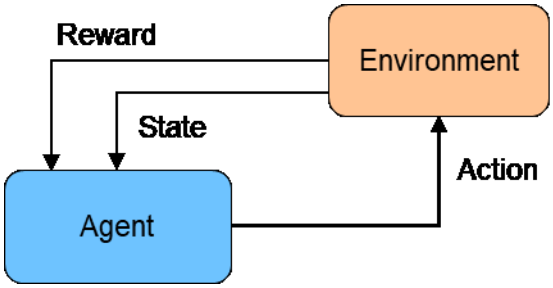

Figure 6. Reinforced learning model [7]

Reinforcement learning can be approached at a higher level with values-based, policy-based or model-based methodologies. Each method can be either positive or negative oriented, depending on how the strengthening of behaviour is defined. As mentioned, there are numerous algorithms to perform reinforcement learning, but couple of widely used learning models are Markov Decision Process and Q-learning.

Application areas vary from industrial automation to data processing, robot motion control to business strategy planning and aviation control.

Challenges in reinforcement learning

Designing a reinforcement learning system requires careful reward function design. The environment, states and rewards include many parameters, which may hinder the speed of learning if misapplied. In addition, realistic environments are often dynamic, which requires additional computational power, and the observability of such environment can be difficult. If there exists plenty of data regarding solving the problem, reinforcement learning might not be the paradigm to go for.

3 Machine learning and safety

Systems are called safety critical, when the failure or malfunction of such system can result in serious injuries, death, or severe damage to equipment or environment. The challenge is, how can these systems be designed, constructed, operated and maintained to keep all the inherent and external risks at an acceptable level, knowing that too excessive focus on safety, will quickly render the whole project economically infeasible. The use of safety critical systems is regulated and governed with strict design and operation requirements, given by a regulating authority or international safety standards. The fulfilment of these requirements needs to be ensured and demonstrated in unarguable, unbiased, comprehensive and transparent way by the designer. Confidence needs to be built for the regulator, but also the system owners need to convince themselves that the system behaviour is as intended.

Thus, 'black box' type of approaches to safety related applications are rarely acceptable. Generally, complex new technologies with inherent dependencies and uncertainties, combined with human beings and natural environment, can lead, especially in unexpected cases, to situations where there is no final answer to the question about systems safety. The big question is, whether machine learning applications are already past this point and if they can they be used to increase the safety of a system, instead of introducing new unacceptable failure mechanisms. The answer is, unfortunately, difficult to give presently. Research efforts do aim for more transparency and verifiability of AI systems operation.

Conventionally, the creation of new state-of-the-art machine learning models with ever-increasing performance has been somewhat ad-hoc in scientific communities. However, various international efforts have been initiated to standardize artificial intelligence (and therefore machine learning to some extent). These efforts include issues related to privacy, trustworthiness, safety and public wellbeing [9]. In addition to local and regional efforts, there are two big international organizations, which are leading the global standardization effort in AI and ML. IEEE has a Global Initiative on Ethics of Autonomous and Intelligent Systems to address some of the societal concerns relating to AI, which include areas like data governance, privacy, algorithmic bias, transparency, ethically driven robots and autonomous systems, failsafe design and wellbeing metrics [9]. In addition, the ISO has created a new technical subcommittee (SC) in the area of artificial intelligence, ISO/JTC 1 SC 42, which the scope lies into foundational standards and issues related to safety and trustworthiness.

The next chapter reviews requirements, which would be relevant for ML applications, if they were to be used in critical industrial domains. Examples of such domains are nuclear, automotive and medical.

3.1 Challenges

Fundamentally, ML application are a specialized type of computer software. Thus, their use in safety critical domains can be related to the use of highly specialized software. At same level, similar kind of software development requirements will apply to machine learning systems as to other software-based applications. Machine learning applications will also have their own specific properties, which cannot be handled by current safety critical software standards, for example testing and validation which will need additional requirements, as well as different metrics to assess their validity. More of the these ML specific challenges are presented later in this chapter. Currently, developing software complying with safety criteria requires rigorous engineering practices enforced by standards. However, problems will arise when the standards were not designed for tasks like ML [1]. As stated by a group of nuclear regulator's and safety authorities' experts in Common Position [10], the assessment of software cannot be limited to verification and testing of the end product, the computer code, but also to other factors, such as the quality of the processes and methods for specifying, designing and coding will have an important factor on the implementation as well as during operation.

Training and using ML models can be convenient and simple on a theoretical level (where training and testing datasets are reconstructed perfectly with very limited complexity). However, when combined with the limitations of real world, and especially industrial setting, many surprising challenges will need solving before the model is fit for safe operation. Unfortunately, how to assure that all the challenges have been sufficiently cleared, and the proper behaviour and safety of an advanced ML technique is certain, is in practice still in many ways an open question. Next are presented some of these fundamental properties of ML, which often will have safety related effects and cause problems in assuring the correct behaviour.

A study about impacts of ML against de facto safety standard ISO 26262 from the automotive industry [11] identified five main areas that are affected when using ML approaches. Authors believe that similar challenges await other safety critical domains too. Many of the topics below

come from the automotive study. However, there are few other practical challenges listed which raise safety related question.

Challenge of identifying new types of hazards

ML will create new types of hazards that are not due to the malfunctioning of a component to be identified with safety analyses. For example, an automated operator might give the real human operator false sense of security when they presume that the algorithm is smarter than it actually is, or a RL component can try to exploit flaws in the environment in a very unintuitive way for a human to gain better results. [11]

Challenge of new types of faults and failure modes

There are ML specific faults and failure modes during their development lifecycle, which need to be explicitly addressed and require the use of specialized tools and techniques that are customized for ML software lifecycle. However, most of them are still closely related to conventional software faults, like incorrect output for some input, but can still cause novel failure modes. [11]

Challenge of traceability

Traditionally, the left side of the V model has the assumption that the component behaviour against a certain hazard is fully specified and can be verified and traced back to its specification. Instead of such specification, ML algorithms use training sets of data, and since these sets are necessarily incomplete, it is not always clear how to create assurance that the corresponding hazards are always mitigated. However, high-level requirements for the ML component can be expressed, and detailed data requirements (“data specification”) can be specified to ensure that an appropriate training, validation and testing data sets are obtained. “Curse of dimensionality” is still a problem, and the training set cannot be increased without limits to try to take care of all possible events. [11]

Challenge of abstract system hierarchy

Typically, complex safety systems implement some hierarchical architecture making it easier for a human to understand its functionalities and interactions of its components. It also permits the use of compositional analysis techniques such as fault tree analysis (FTA). However, ML (especially DL) can be used to implement so called ‘end-to-end’ approach, consisting an entire software-based system, including its architecture. For example, making control decisions directly from raw sensor data. Thus, a traditional sense of architecture no longer exists in these solutions. Salay et. al. suggest that ML should not be used with ‘end-to-end’ level, if the assumption is the need of a stable hierarchical architecture of software components. [11]

Challenge of software development

Safety-critical software standards (such as part 6 of ISO 26262 or Annex D of 60880) are biased towards imperative programming languages (e.g. C, Java), which ML components are often not. However, other programming paradigms are already mature and specifying requirements based on intent and maturity rather than specific details can help addressing this gap. Data scientists make usually the critical decisions relating to development of ML components such as selection and preparation of data and the data quality inspection, which could be argued to belong to the domain of requirements engineering. A recent questionnaire survey on difficulties in engineering of machine-learning systems even notes how requirements engineering activities are one of the most difficult activities to cope within ML systems [12]. To be more adjusted towards ML-based systems, requirements engineering methodology is suggested to be broaden with new types of requirements such as explicability, freedom from discrimination or specific legal requirements [13]. A publication on requirements engineering for machine learning also highlights how data scientist or requirements engineers need to understand the quantitative ML measures to specify good functional requirements [13]. This is

an important aspect in order to relate the developed application and results of ML-based methods to the customer context. [11]

Challenge of interpretability

Interpretability vs. performance of ML algorithms (Figure 7). As stated in [2] the spectrum of available ML methods is wide, and they range from simpler and more interpretable to more advanced, with potentially higher performance, but with less interpretability. Interpretability can depend on number of free parameters, model complexity, data types, etc. When increasing the expressive power of the model, it is typically done at the expense of transparency. Non-transparency is a problem to safety assurance as it makes it more difficult for the assessor to gain confidence that the model is operating as intended [11].

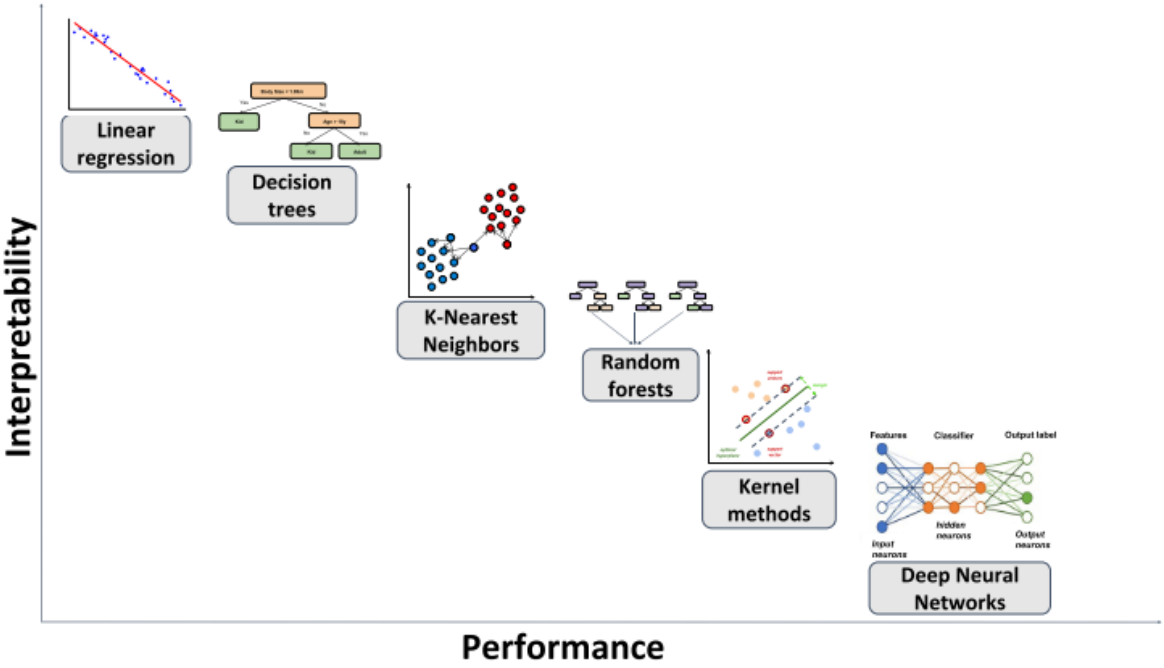

Figure 7. Interpretability vs. Performance [2]

All ML paradigms (supervised, unsupervised, reinforcement) contain the knowledge about the behaviour of the model in an encoded form. However, as can be seen from Figure 7, some types are harder to interpret for humans than others [11].

Challenge of training data

Independent of the ML paradigm used, data is the most crucial characteristics for a successful ML algorithm. Training data needs to represent the actual properties of the situation the algorithm is going to face while in operation. As a rule of thumb, more data equals better results. The phrase is especially true when using neural networks. It is crucial to use the available domain knowledge and expertise as much as possible to come up with the most relevant events and features of the system the data is representing, and at the same time identify the events, which will be problematic for the algorithm. Data should be an unbiased random subset of the system it is representing, without any single case/event overrepresenting over the rest, even rare events. However, as the amount of possible training inputs is only finite, it is possible that some input that could be encountered while in operation may not be available and thus leading to a critical situation. The amount of data is not the only key to success, also the quality of data plays an important role in ML. Unprocessed data can include oddities challenging the performance of algorithms, such as extreme values, and outliers or

missing data, which need to be handled somehow. How to do this depends on the data itself and the algorithms used.

Challenge of generalization

Relating to the topic above, if the data selection is not carefully engineered to represent the system under consideration, it is possible that the ML model overfits the data presented to it in the training phase. In this case, it will lose its ability to predict unseen cases in implementation, often because they are too complex for the model (when overfitting the model captures the details only incidental to the training set rather than general to all inputs [11]). Figure 8 gives an example of under- and overfitting on a time series data. This creates uncertainty about how the model will behave once deployed. This problem relates very closely to the availability/quality of training and testing data and the accuracy/error rate of the ML model.

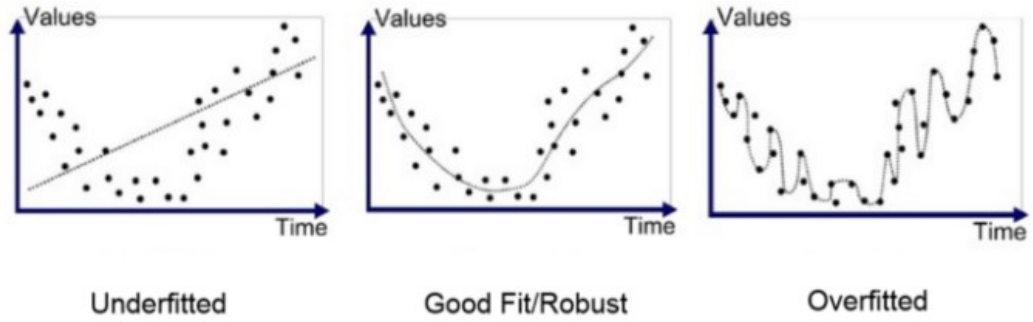

Figure 8. Examples of underfitting and overfitting [14]

Challenge of uncertainty

ML models are typically trained based on accuracy, which is a measurement of how often the correct option is chosen, and loss, i.e. a statistic of how far off the model is. They are both used during the training phase, but the results are only an estimate and do not necessarily correspond to the actual performance or reliability of the model while it is in operation and using full input domain. [1]. The selection of representative training data and the specification of operation range of ML model is important to address these uncertainties.

However, no matter how well the ML model is planned and trained, it typically always contains some error rate (e.g. the classification accuracy is 97%). Thus, it must be assumed that the model will periodically fail. The accuracy estimate gained during the training and testing process is still a statistical guarantee about the reliability of the model based on the finite set of inputs used during the training. The true accuracy might be different as it is based in infinite set of samples, which might change or drift from those originally shown to the model. [11].

Challenge of local optima

Especially Deep Neural Networks operate using local optimization algorithms, and in many cases, there are many local optima. It is not guaranteed, when trained multiple times, that the process produces the same optima each time, even if the training set is the same. Thus reusing parts of previous models is difficult, especially considering safety assessments. [11].

Challenge of spurious correlations

Machine learning systems are highly dependent on the data, luckily there is usually a lot of data (big data) available from the industrial processes being measured, transferred, analysed and stored by the instrumentation and control systems of the plant, which can be fed into machine learning algorithms to train them. However, the data and correlation found by the

algorithms cannot be trusted blindly. If not specified and analysed by someone, who actually knows what the data means during the development and training of these algorithms, the outcome might be something else than originally wanted. Thus, the development of machine learning algorithms needs to include as much as possible the domain expertise from whatever domain the algorithms are applied to. They have the knowledge to specify and analyse what sort of data will be needed and are the outputs of the algorithms meaningful in any way. Figure 9 is one funny example of data that seems to be correlating, but probably does not contain any meaningful information; it is just something that the algorithm has found while blindly processing the data.

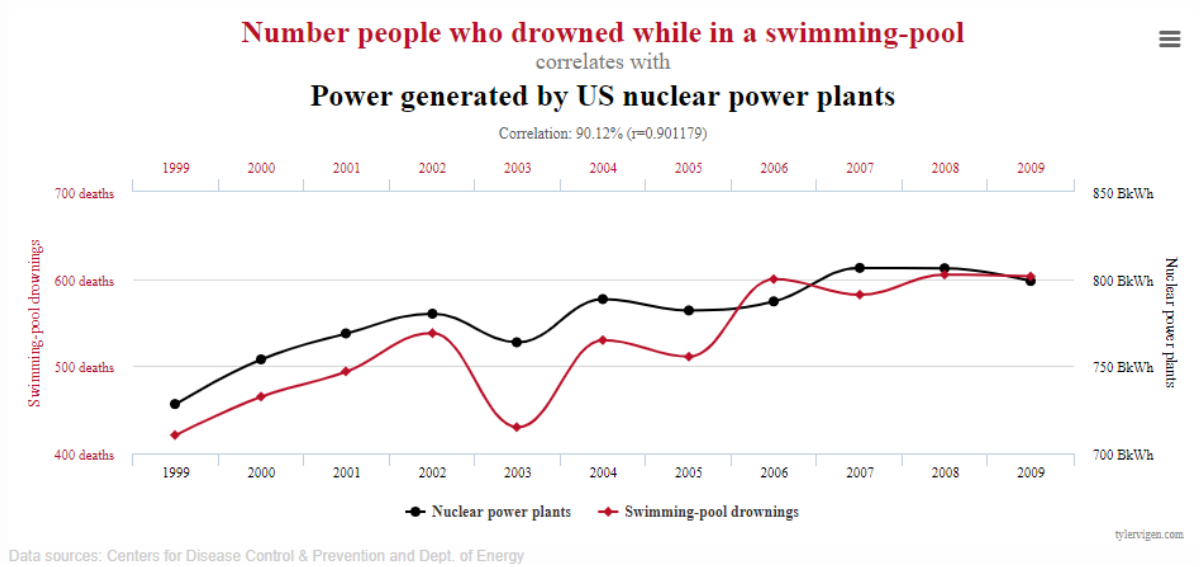

Figure 9. Not all correlations in data are actually meaningful for the system in question [15]

Challenge of security

Machine learning opens networks and systems for a multitude of new cyber-attacks as illustrated in Figure 10. ML systems are vulnerable for adversaries that can input bogus or tampered data during the learning time - poisoning - or testing time - evasion [16], [17]. Furthermore, ML systems may also leak information [18] and contain backdoors [19]. Within industrial domain that is using ML systems these attacks may then yield different consequences. Company or customer specific secrets may leak, if an adversary learns operational, organizational, or business critical information from data, which is collected or inferred, or from ML models. Devices may become broken and services unavailable due to fabricated or spoofed configurations, or overload situations. Security incidents and misconfiguration may follow, if ML is prevented from detecting and generating signals from events, cyber-attacks, or failures. Resources may be stolen, used unfairly, or extra burden may be caused to the victims.

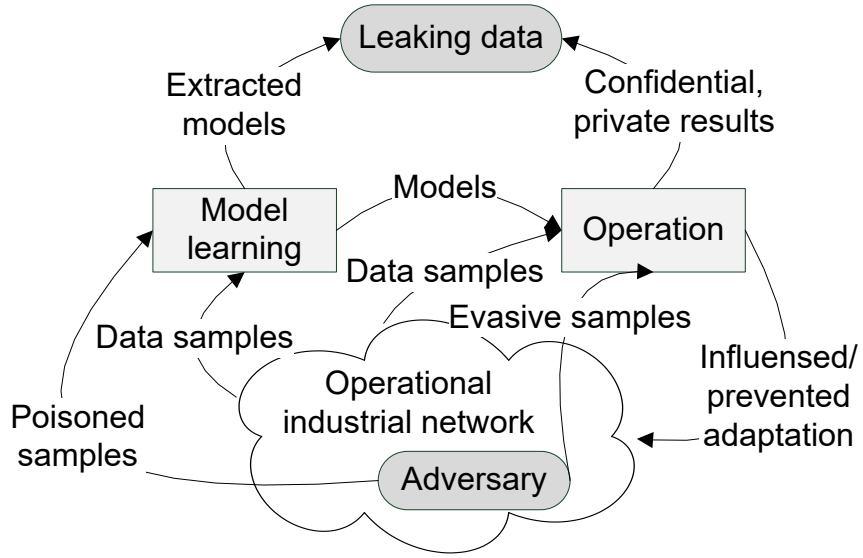

Figure 10. Security threats against machine learning

Solutions for hardening ML systems against threats, include traditional perimeter defences, which protect the confidentiality, integrity, and availability of ML systems, models, and data sources as well as algorithmic and data-oriented defences [20], [21]. For instance, adversarial training approaches increase the robustness of algorithms by including the attack data into learning data sets. Security assessments and penetration testing may be applied to verify ML systems robustness. Reactive defences include detection of adversarial samples from the data streams, concept drift (monitoring performance of ML model). Privacy preserving techniques, such as e.g. secure aggregation, cryptographic means, and differential privacy may be applied to mitigate risk of information leakage.

4 Machine learning in practice

4.1 Thoughts of Finnish nuclear stakeholders

Part the study was about discussing the topic with Finnish nuclear stakeholders from the Steering Group. Informal meetings were held with STUK (Finnish nuclear regulator), Fennovoima (licence holder in design phase), TVO (licence holder in licencing and modernization phase), and Fortum (licence holder in modernization phase). During these meetings, a draft of this report was presented to introduce the participants to the topic. They were mostly experts in various nuclear engineering domains rather than ML. Participants were keen to learn about the topic and delivered interesting conversation about the different properties of ML, how it could be applied to various industrial (nuclear) topics and what challenges they might raise. As much as possible of the ideas and comments from the conversations were integrated as part of this report. During the meetings, the potential use cases of machine learning, specifically in nuclear domain, were discussed, and potential research and development topics were presented from both sides (researchers and industry). The novel main points of the discussions are gathered below, divided in more general comments related to the use of ML or AI and to interesting current or potential topics of support systems utilizing some AI-based technology:

General comments:

- All the stakeholders agreed that requirement management is a constant challenge and would benefit from some sort of automated assistance (e.g. based on machine learning); especially NLP methods have high potential in text processing and labelling.
- How can different ways of doing Systems Engineering be assisted by AI (e.g. document-centric vs. SE vs. MBSE)?
- There were comments on both directions of the issue that humans might start to rely too much on AI-based systems.
- Loss of skills due to AI-based system adoption was not seen as a high risk.
- When it can be demonstrated that AI improves the overall safety (e.g. it makes less errors than humans do in similar situation), then we can give more value to AI's opinion.
- While humans are still in charge of decisions, the chance of reducing safety should be low.
- There is always the need for continuous improvement and the AI-based methods might be the way forwards, we need to make sure it is applied in a positive way.
- Machine learning should be acceptable if the algorithms and the data are validated, and the development process is good.
- There are huge amounts of data available or produced in nuclear power plants (process data, used history data, operating experiences, manuals etc.) this data should be categorized and labelled with metadata where possible to be able to use it properly.
- There exist initiatives to identify potential data sources in running plants for implementing different data-driven methods.
- Process simulators exists and could be used for generating synthetic data for the rare events that would be interesting to identify or possibly act as a training platform for different algorithms.
- The quality of data is important; there are broad spectrum of possible events and data sources. Thus, domain expertise is very important for interpreting inputs and outputs.
- Is there is any useful applications for Watson in nuclear?
- Discussing and learning the concepts of AI and/or ML in nuclear applications is useful for sharing knowledge and discovering novel ideas.
- EU and BF projects are a good possibility to collaborate on machine learning topics.

Potential research/development directions for ML based systems:

- Predictive maintenance/fault identification systems for different applications, some systems are already in use with positive experiences. These could include, using simulator data, control room information, alarm data to help plant personnel to make decisions in operation, maintenance or accident scenarios.
- During complex projects, the amount of emails, documents and decisions tend to get large. Handling references between documents and automatically fetching or suggesting information would be handy.
- Using BIM/PIM (Building/Plant information models) to check or analyse requirements, e.g. separation.

- Using different sources of data to get indications of possible security related issues, either cyber-, site- or personnel.
- Genetic algorithms have been tried for fuel assembly to find (at least) local optima, but there are still optimizations to be done for improving the safety and financial usage.
- Planning, scheduling and optimizing the order of different tasks, usage of different tools and workforce during annual outages. Also, the alarm prioritising.
- Operator support in various situations. How to identify faults? How to foresee possible combined effect of faults? How to find the initial event in case of quickly evolving accident scenario?

4.2 Overview on selected machine learning applications

Security applications

Security applications of machine learning provide a promise of more rapid and autonomous security responses within operational industrial environments. Further, ML systems can play a role in adaptive collection of security information as well as optimization of security services. Examples of security applications - threat detection, testing, and honeypots - are illustrated in Figure 11.

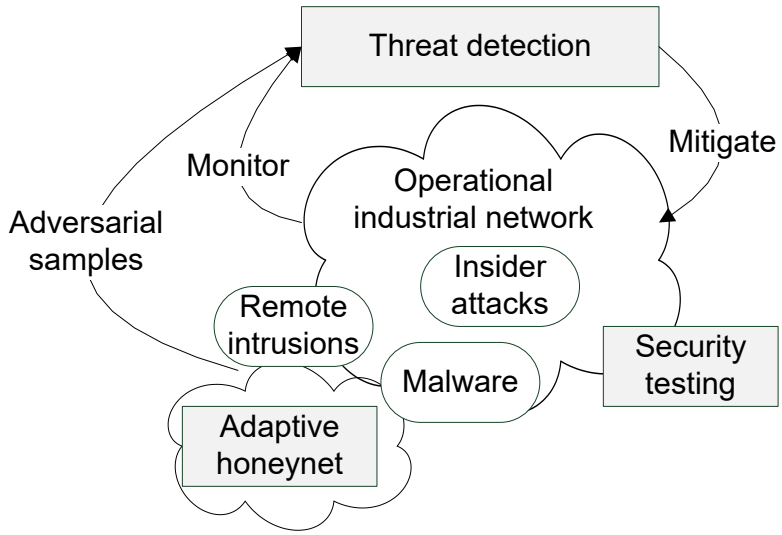

Figure 11. Examples of security applications benefitting from machine learning

Cyber threat - attack and intrusion - detection and prevention systems are prominent users of machine learning [22]. Both signature as well as anomaly-based threat detection approaches can benefit from the automation provided by ML. In the first case, ML can have a role in modelling the adversarial behaviour and, in the latter case, in modelling the normal behaviour. Examples of security applications in the industrial domain include detections of network intrusions [23], [24] as well as malware [25]. In one hand, industrial systems can greatly benefit from the added security provided by the ML-assisted threat detection. Industrial systems have traditionally been for closed environment and are thus poorly secure. Reactive defences can partly fill this gap, while also providing additional layers of security. On the other hand, there are several challenges limiting the adaptation of ML-based attack detection. Real-time nature of systems and existence of legacy technologies limits the possibilities to add new processing and data collection features. Within industrial control systems settings, the applications and protocols are rarer or customized and thus, there has been less research efforts and

commercial tools. Further, for customized industrial settings, realistic adversarial training and testing data is especially difficult to acquire. Means to learn to detect new attacks are poor when attacks, producing new training data, are relatively rare, and when existing limited public data sets [26] are not been easily applied to industrial use cases.

In addition to run-time security for operational networks, there are also other security applications for ML. Honeypots and -nets are environments, which are isolated from the operational technologies and which have been deployed to the network to lure and capture adversaries [27]. In addition to providing a defence by capturing some attacks, honeypots are a mechanism to collect data for learning and building models on adversarial behaviour. ML techniques have been proposed [28] for building more intelligent and realistic honeypots that adversaries cannot easily distinguish from real operating environments.

Further, machine learning can be utilized when testing security properties of industrial system and devices. Security scanners can utilize ML to recognize vulnerable software [29] or to adapt scanners functionality [30]. In operational environments, scanning is typically restricted to prevent production breaks. Instead, industrial organizations test individual products and non-operational digital twin environments [31].

Fault identification / predictive maintenance

Predictive maintenance applications strive to determine the condition of engineering systems to estimate the requirement of maintenance. Machine learning offers a variety of approaches for exploiting sensory data that can be utilized in maintenance purposes. These methods are often built on existing history datasets of operation of the systems. Now considering ML, the algorithm has the possibility to learn efficiently and detect aberrations among the data only, if the datasets include and cover all the possible scenarios the system can be exposed to. Incorporating the volume of stored and processed data requires a big data framework, which the ML algorithm can access for analyzation [26]. Although there are a variety of equipment dynamically producing data in an NPP, there are rarely sufficient data of situations where specific equipment have failed during plant operation. Additionally, to develop accurate and robust predictions, a model requires a combination of domain knowledge and statistical expertise.

Though considered as a black box, deep learning-based techniques have also been utilized especially for automatic feature extraction. Yet, traditional machine learning algorithms like Support vector machines are still mainly used because of their transparency and simplicity. A recent nuclear related research paper [32] presents a fault prediction architecture and presents an example case of a turbofan engine. The paper presents several research studies on anomaly detection in a NPP environment, such as [33], [34], [35]. However, the paper does not have appropriate real data gathered from an NPP, but instead, results from a dataset from the National Aeronautics and Space Administration (NASA). The case demonstrates the difficult situation of utilizing predictive maintenance in an environment where there is no pre-collected data of the equipment in its relevant environment (an NPP in this case). Nonetheless, the paper displays how a predictive maintenance framework for nuclear infrastructure can be utilized.

A successful demonstration of predictive maintenance in an industry use case can be seen from Mathworks with a developed pump health monitoring system for Baker Hughes [36]. In order to notice a truck with a pump failure a large dataset was collected and extensively analysed from valves and valve seats. For accurate prognosis, Mathworks had compared different machine learning models, such as classification and regression-based models from supervised learning. Using supervised learning-based models required data to be labelled, which can be accomplished, for instance, with unsupervised learning.

Requirements engineering (processing/categorization)

Quality of the requirements is an important factor in the design of any system, especially of systems with safety requirements. In complex safety critical designs, the number of these requirements can go up to tens of thousands. Requirements engineering is a discipline that puts great effort into ensuring that the quality of these requirements is high. Machine learning and especially Natural Language Processing (NLP) techniques have been used in this domain to help expert review and develop better requirements. The challenge is to create a pattern recognition for the given task and manage a classification accordingly.

An example of requirements quality classification with ML-based methods is displayed on paper [37]. The proposed methodology evaluates the quality of requirements written in natural language with a decision tree-based classifier. Classification trees are also used for separating functional and quality-focused concerns to facilitate further assessment of the system on paper [38]. The aforementioned examples display how challenges in requirements engineering can be approached with classification techniques whether for validation or categorization purposes or even for requirements traceability [39].

Operator decision support:

Operating an NPP is a challenging task [40] on its own and when different complex fault situations with multiple possible initial sources and quickly evolving situations are introduced to the mix, things get increasingly difficult. Human mind can only focus on a limited number of stimuli and process only a limited amount information during a certain timespan; when on the other hand, a machine can process huge amount of real time data and, at the same time, remember decades of history data. Informative, diagnostic and predictive ML-based techniques offer a possibility to ease the amount of tasks at hand, for instance, by providing predictions of the behaviour of the system or by providing technical information from data to help decision makers perform appropriate and timely actions. Prescriptive systems can even help controlling plant from a state to another.

An example of ML usage in behaviour and decision-making is displayed in paper [41]. Applied methodology is based on neural networks for predicting how a multi-application small light water reactor is performing. The operator who is watching over the sensors provides the utilized data. The results display how algorithms were able to learn from complex data the average of most of the sensors and thereby give predictions of processes; although most of the sensors behaviour was successfully captured, analysing the facility as a whole system could not be accurately achieved. However, the paper concludes on methods to potentially overcome this issue in the future with additional transients.

Another example of process industry using ML support for the plant operation is the Napcon Advisor [42], which has been successfully used a trained neural network to predict the changes in a process (oil refinement) and to control the process in some state changes (changing from one oil quality to another). The system is said to be predictive and prescriptive. Similar AI assisted or autonomous control trials and research are going on, for example, in paper [43] and in chemical [44] industries. Current hot domain for the autonomous control is the automotive industry, where several big players are already testing their implementation in practice. It is the leading force of standards and legislation concerning the use of ML in safety critical decision making. Similarly, in all domains, networks are trained using the history data of operations, simulations and observing the running process. Difficulties come from adjusting the network to unforeseen changes in the operation environment and diverse incoming measurement data, which the algorithm would need to understand without ever encountering data of that particular format.

An example of automatic alarm for fault management in cellular networks is displayed on paper [45]. Though not concentrating on the nuclear domain, the paper introduces a ML-based technique for automatic prioritization of alarms in a network according to the need of key personnel. The novelty of the study lies in the comparison of different classifier algorithms

within the cellular network domain. The proposed methodology of supervised learning offers possibilities considered also for other domains.

5 Conclusions

Based on the study and the discussions with the Finnish nuclear stakeholders, the authors conclude that there is an increasing interest in utilizing different machine learning methods to offer support for the work of various engineering domains across the wide spectre of plant lifecycle phases. The highlighted domains were, for instance, cyber security, requirements engineering and plant maintenance, while the most potential topics identified related to supporting decision making of operators and utilization of predictive models. However, this report and the discussions were only able to scratch the surface of the vast potential of various industrial applications, which different algorithms of machine learning and the various domains artificial intelligence could offer to nuclear industry.

Machine learning, and artificial intelligence in general, is a complex and quickly evolving domain of different techniques, practices and challenges. It has the capability and the potential to offer support to wide and diverse set of problems in versatile sectors of industry, including the safety critical ones. However, these methods and techniques still have some fundamental challenges that need to be properly addressed when these methods are implemented in industrial settings, some of which are presented in this report, such as quality and suitability of training data and the interpretability of used models. Many of these challenges are, in a sense, part of the maturing process of machine learning methods finally making their big breakthrough as a part of daily business life of traditional industry domains. Yet, researchers around the globe are actively seeking, and succeeding, to solve these challenges. The study of the suitability of different machine learning techniques to Finnish nuclear scene is already going inside stakeholder organizations, in national programs, and should be even further extended through common research topics utilizing the good current AI knowledge identified in Finland.

In the end, machine learning is only a data processing tool, which usefulness completely depends on the quality and the properties of the used data. To get functional, reliable and safe support from the AI system, developers need to first have clear comprehension of the operational context themselves to understand what the algorithm is supposed to do. To get there, significant domain expertise is required to select and explain the diverse conditions needed from the algorithm and data.

Further work on the topic could consider a more specific application area to concentrate on, and to apply there a targeted machine learning paradigm with the help of domain experts. Discussions with the nuclear stakeholders brought forward many interesting views and questions, which could be more accurately specified with a more thorough study and a possible demonstration use case.

References

[1] J. Henriksson, M. Borg, and C. Englund, “Automotive safety and machine learning: Initial results from a study on how to adapt the ISO 26262 safety standard,” in *Proceedings - International Conference on Software Engineering*, May 2018, pp. 47–49, doi: 10.1145/3194085.3194090.

[2] S. Badillo *et al.*, “An Introduction to Machine Learning,” *Clin. Pharmacol. Ther.*, vol. 107, no. 4, pp. 871–885, Apr. 2020, doi: 10.1002/cpt.1796.

[3] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553.

Nature Publishing Group, pp. 436–444, May 27, 2015, doi: 10.1038/nature14539.

[4] C. Kumar GN, “Artificial Intelligence: Definition, Types, Examples, Technologies,” *Medium.com*, 2018. https://medium.com/@chethankumargn/artificial-intelligence-definition-types-examples-technologies-962ea75c7b9b (accessed Jun. 12, 2020).

[5] “Four levels of Analytics/Data Science.” https://koopingshung.com/blog/four-levels-of-analytics-data-science-descriptive-diagnostic/ (accessed Sep. 15, 2020).

[6] J. Song, H. Wang, Y. Gao, and B. An, “Active learning with confidence-based answers for crowdsourcing labeling tasks,” *Knowledge-Based Syst.*, vol. 159, pp. 244–258, Nov. 2018, doi: 10.1016/j.knosys.2018.07.010.

[7] J. P. Mueller and L. Massaron, *Machine Learning for Dummies*. Hoboken, UNITED STATES: John Wiley & Sons, Incorporated, 2016.

[8] “Tutorial Diagrams — scikit-learn 0.11-git documentation.” https://ogrisel.github.io/scikit-learn.org/sklearn-tutorial/auto_examples/tutorial/plot_ML_flow_chart.html (accessed Jun. 12, 2020).

[9] F. Rudzicz, P. A. Paprica, and M. Janczarski, “Towards international standards for evaluating machine learning,” in *CEUR Workshop Proceedings*, 2019.

[10] Common Position, “Licensing of safety critical software for nuclear reactors - Common position of seven European nuclear regulators and authorised technical support organisations,” 2018.

[11] R. Salay, R. Queiroz, and K. Czarnecki, “An Analysis of ISO 26262: Using Machine Learning Safely in Automotive Software,” Sep. 2017, Accessed: May 06, 2020. [Online]. Available: http://arxiv.org/abs/1709.02435.

[12] F. Ishikawa and N. Yoshioka, “How Do Engineers Perceive Difficulties in Engineering of Machine-Learning Systems? - Questionnaire Survey,” in *Proceedings - 2019 IEEE/ACM Joint 7th International Workshop on Conducting Empirical Studies in Industry and 6th International Workshop on Software Engineering Research and Industrial Practice, CESSER-IP 2019*, May 2019, pp. 2–9, doi: 10.1109/CESSER-IP.2019.00009.

[13] A. Vogelsang and M. Borg, “Requirements Engineering for Machine Learning: Perspectives from Data Scientists,” *Proc. - 2019 IEEE 27th Int. Requir. Eng. Conf. Work. REW 2019*, pp. 245–251, Aug. 2019, Accessed: Sep. 17, 2020. [Online]. Available: http://arxiv.org/abs/1908.04674.

[14] “What is underfitting and overfitting in machine learning and how to deal with it. | by Anup Bhande | GreyAtom | Medium.” https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76 (accessed Sep. 15, 2020).

[15] “Spurious Correlations.” https://www.tylervigen.com/spurious-correlations (accessed Sep. 14, 2020).

[16] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar, “Can machine learning be secure?,” in *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, ASIACCS ’06*, 2006, vol. 2006, pp. 16–25, doi: 10.1145/1128817.1128824.

[17] B. Biggio *et al.*, “Evasion attacks against machine learning at test time,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2013, vol. 8190 LNAI, no. PART 3, pp. 387–402, doi: 10.1007/978-3-642-40994-3_25.

[18] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song, “The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks,” *Proc. 28th USENIX Secur.*

Symp., pp. 267–284, Feb. 2018, Accessed: Sep. 04, 2020. [Online]. Available: http://arxiv.org/abs/1802.08232.

[19] T. Gu, B. Dolan-Gavitt, and S. Garg, “BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain,” Aug. 2017, Accessed: Sep. 04, 2020. [Online]. Available: http://arxiv.org/abs/1708.06733.

[20] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, “Towards the Science of Security and Privacy in Machine Learning,” Nov. 2016, Accessed: Sep. 04, 2020. [Online]. Available: http://arxiv.org/abs/1611.03814.

[21] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, “A survey on security threats and defensive techniques of machine learning: A data driven view,” *IEEE Access*, vol. 6, pp. 12103–12117, Feb. 2018, doi: 10.1109/ACCESS.2018.2805680.

[22] A. L. Buczak and E. Guven, “A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 2, pp. 1153–1176, Apr. 2016, doi: 10.1109/COMST.2015.2494502.

[23] J.-M. Flaus and J. Georgakis, “Review of machine learning based intrusion detection approaches for industrial control systems,” in *Computer & Electronics Security Applications Rendez-vous (C&ESAR) Conference*, 2018, Accessed: Sep. 04, 2020. [Online]. Available: https://www.cesar-conference.org/wp-content/uploads/2018/11/articles/C&ESAR_2018_J2-12_JM-FLAUS_Detection_intrusion_par_ML_pour_ICS.pdf.

[24] A. Ayodeji, Y. Liu, N. Chao, and L. Y. Technology, “A new perspective towards the development of robust data-driven intrusion detection for industrial control systems,” *Nucl. Eng. Technol.*, 2020, Accessed: Sep. 04, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1738573320300590.

[25] S. Sharmeen, S. Huda, J. H. Abawajy, W. N. Ismail, and M. M. Hassan, “Malware Threats and Detection for Industrial Mobile-IoT Networks,” *IEEE Access*, vol. 6, pp. 15941–15957, Mar. 2018, doi: 10.1109/ACCESS.2018.2815660.

[26] J. Mchugh, “Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory,” *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 262–294, Nov. 2000, doi: 10.1145/382912.382923.

[27] E. Vasilomanolakis, S. Srinivasa, and M. Muhlhauser, “Did you really hack a nuclear power plant? An industrial control mobile honeypot,” in *2015 IEEE Conference on Communications and NetworkSecurity, CNS 2015*, Dec. 2015, pp. 729–730, doi: 10.1109/CNS.2015.7346907.

[28] W. Z. A. Zakaria and M. L. M. Kiah, “A review on artificial intelligence techniques for developing intelligent honeypot - IEEE Conference Publication,” in *2012 8th International Conference on Computing Technology and Information Management (NCM and ICNIT)*, 2012, pp. 696–701, Accessed: Sep. 04, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6268588/citations?tabFilter=papers#citations.

[29] F. Yamaguchi, F. Lindner, and K. Rieck, “Vulnerability extrapolation: Assisted discovery of vulnerabilities using machine learning,” in *Proceedings of the 5th USENIX conference on Offensive technologies*, 2011, Accessed: Sep. 04, 2020. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/2028052.2028065.

[30] X. Tian and D. Tang, “A distributed vulnerability scanning on machine learning,” in *Proceedings - 2019 6th International Conference on Information Science and Control Engineering, ICISCE 2019*, Dec. 2019, pp. 32–35, doi: 10.1109/ICISCE48695.2019.00016.

[31] A. Becue *et al.*, “CyberFactory#1 - Securing the industry 4.0 with cyber-ranges and digital twins,” in *IEEE International Workshop on Factory Communication Systems - Proceedings, WFCS*, Jul. 2018, vol. 2018-June, pp. 1–4, doi: 10.1109/WFCS.2018.8402377.

[32] H. A. Gohel, H. Upadhyay, L. Lagos, K. Cooper, and A. Sanzetenea, “Predictive maintenance architecture development for nuclear infrastructure using machine learning,” *Nucl. Eng. Technol.*, vol. 52, no. 7, pp. 1436–1442, Jul. 2020, doi: 10.1016/j.net.2019.12.029.

[33] A. Letourneau *et al.*, “Nucifer: A small electron-antineutrino detector for fundamental and safeguard studies,” in *ANIMMA 2011 - Proceedings: 2nd International Conference on Advancements in Nuclear Instrumentation, Measurement Methods and their Applications*, 2011, doi: 10.1109/ANIMMA.2011.6172942.

[34] B. Stephen, G. M. West, S. Galloway, S. D. J. McArthur, J. R. McDonald, and D. Towle, “The use of hidden Markov models for anomaly detection in nuclear core condition monitoring,” *IEEE Trans. Nucl. Sci.*, vol. 56, no. 2, pp. 453–461, Apr. 2009, doi: 10.1109/TNS.2008.2011904.

[35] B. R. Upadhyaya and F. Li, “Optimal sensor placement strategy for anomaly detection and isolation,” in *2011 Future of Instrumentation International Workshop, FIIW 2011 - Proceedings*, 2011, pp. 95–98, doi: 10.1109/FIIW.2011.6476832.

[36] “Predictive Maintenance with MATLAB: A Prognostics Case Study - Video - MATLAB.” https://www.mathworks.com/videos/predictive-maintenance-with-matlab-a-prognostics-case-study-118661.html (accessed Sep. 16, 2020).

[37] E. Parra, C. Dimou, J. Llorens, V. Moreno, and A. Fraga, “A methodology for the classification of quality of requirements using machine learning techniques,” in *Information and Software Technology*, Nov. 2015, vol. 67, pp. 180–195, doi: 10.1016/j.infsof.2015.07.006.

[38] J. H. Hayes, W. Li, and M. Rahimi, “Weka meets TraceLab: Toward convenient classification: Machine learning for requirements engineering problems: A position paper,” in *2014 IEEE 1st International Workshop on Artificial Intelligence for Requirements Engineering, AIRE 2014 - Proceedings*, 2014, pp. 9–12, doi: 10.1109/AIRE.2014.6894850.

[39] H. Sultanov and J. H. Hayes, “Application of reinforcement learning to requirements engineering: Requirements tracing,” in *2013 21st IEEE International Requirements Engineering Conference, RE 2013 - Proceedings*, 2013, pp. 52–61, doi: 10.1109/RE.2013.6636705.

[40] R. J. Mumaw, E. M. Roth, K. J. Vicente, and C. M. Burns, “There is more to monitoring a nuclear power plant than meets the eye,” *Human Factors*, vol. 42, no. 1. Human Factors and Ergonomics Society, pp. 36–55, Mar. 06, 2000, doi: 10.1518/001872000779656651.

[41] M. Gomez Fernandez, A. Tokuhiro, K. Welter, and Q. Wu, “Nuclear energy system’s behavior and decision making using machine learning,” *Nucl. Eng. Des.*, vol. 324, pp. 27–34, Dec. 2017, doi: 10.1016/j.nucengdes.2017.08.020.

[42] “Machine Learning in the NAPCON Advisor - NAPCON.” https://www.napconsuite.com/machine-learning-in-the-napcon-advisor/ (accessed Sep. 17, 2020).

[43] “Can artificial intelligence run a paper machine better than humans?” https://www.tietoevry.com/en/blog/2018/04/can-artificial-intelligence-run-a-paper-machine-better-than-humans/ (accessed Sep. 18, 2020).

[44] D. Alves Goulart and R. Dutra Pereira, “Autonomous pH control by reinforcement learning for electroplating industry wastewater,” *Comput. Chem. Eng.*, vol. 140, p. 106909, Sep. 2020, doi: 10.1016/j.compchemeng.2020.106909.

[45] A. J. García, M. Toril, P. Oliver, S. Luna-Ramírez, and M. Ortiz, “Automatic alarm prioritization by data mining for fault management in cellular networks,” *Expert Systems with Applications*, vol. 158. Elsevier Ltd, p. 113526, Nov. 15, 2020, doi: 10.1016/j.eswa.2020.113526.