Foundations of Artificial Intelligence and Machine Learning

Früh, Alfred; Haux, Dario

Erstveröffentlichung / Primary Publication
Arbeitspapier / working paper

This work has been funded by the Federal Ministry of Education and Research of Germany (BMBF) (grant no.: 16DII121, 16DII122, 16DII123, 16DII124, 16DII125, 16DII126, 16DII127, 16DII128 - "Deutsches Internet-Institut").

Empfohlene Zitierung / Suggested Citation:
Früh, A., & Haux, D. (2022). *Foundations of Artificial Intelligence and Machine Learning.* (Weizenbaum Series, 29). Berlin: Weizenbaum Institute for the Networked Society - The German Internet Institute. https://doi.org/10.34669/WI.WS/29

weizenbaum
institut

WEIZENBAUM SERIES \29

WORKING PAPER \

JULY 2022 \

Alfred Früh and Dario Haux

Foundations of Artificial Intelligence and Machine Learning

Foundations of Artificial Intelligence and Machine Learning*

Alfred Früh \ University of Basel \ alfred.frueh@unibas.ch
Dario Haux \ University of Basel \ dario.haux@unibas.ch

ISSN 2748-5587 \ DOI 10.34669/WI.WS/29

EDITORS: The Managing Board members of the Weizenbaum-Institut e.V.
Prof. Dr. Christoph Neuberger
Prof. Dr. Sascha Friesike
Prof. Dr. Martin Krzywdzinski
Dr. Karin-Irene Eiermann

Hardenbergstraße 32 \ 10623 Berlin \ Tel.: +49 30 700141-001
info@weizenbaum-institut.de \ www.weizenbaum-institut.de

TYPESETTING: Roland Toth, M.A., Luisa Le van, M.A.

WEIZENBAUM INSTITUTE: The Weizenbaum Institute for the Networked Society – The German Internet Institute is a joint project funded by the Federal Ministry of Education and Research (BMBF). It conducts interdisciplinary and basic research on the changes in society caused by digitalisation and develops options for shaping politics, business and civil society.

This work has been funded by the Federal Ministry of Education and Research of Germany (BMBF) (grant no.: 16DII121, 16DII122, 16DII123, 16DII124, 16DII125, 16DII126, 16DII127, 16DII128 – "Deutsches Internet-Institut").

Abstract

Today, artificial intelligence and machine learning play a crucial role in various fields of application. In one way or another, they influence our everyday lives. This current state of affairs and the suggestive power of these terms have triggered fundamental discussions in society. However, the technical basics have not received the attention they deserve – and need. This is especially true from a legal perspective, where groundwork on both the fundamental functionality as well as all the relevant terms surrounding the technology seems to be almost non-existent. This paper aims to fill this gap. We examine the technical background of artificial intelligence and machine learning from an interdisciplinary perspective and aim to develop common definitions that can be used for further research in legal academia. These findings provide a common starting point for a more differentiated treatment of legal (and technical) questions surrounding artificial intelligence and machine learning and allow legal academia to make reliable legal statements as well as to advance legal research in this field.

* This working paper is part of a research project on technical and legal countermeasures against deep learning perturbations. The authors thank Sophie Zimmermann, MLaw and Anna Brand, BLaw for helpful research.

Table of Contents

1 Introduction

The increasing importance of Artificial Intelligence for systems used in our everyday lives[1] has, unsurprisingly, produced an abundance of literature in various fields regarding possible definitions and notions as well as related concepts, techniques and technologies. However, from a lawyer's perspective, most of them are not a suitable starting point for making reliable legal assessments about these new technologies. Statements on the 'explainability' of AI *in general* are, for example, misleading, false or useless, considering the plethora of different technologies in the realm of AI. The same is true for other legal deliberations on systems, be it on their robustness against cybersecurity threats[2], the possibility to protect them under intellectual property laws[3], their autonomy[4] when assessing 'their' harmful actions, etc.

In this paper, we draw from (some of) the abundant literature in Computer Science. Then, we identify core concepts and technologies, sharpen their contours and relate them to each other as well as to terms that are already being used in the legal sphere. This yields a detailed and coherent picture, which, in itself, is novel. Computer Science generally does not have an interest in these systematic aspects and rather focuses on specific issues. As a consequence, we suppose and hope that this work of 'translating' technical aspects to the legal sphere is useful for other legal scholars in the field.[5]

2 Groundwork

2.1 Concepts and Related Terminology

2.1.1 Artificial Intelligence (AI)

The term «*Artificial Intelligence*» or briefly «*AI*», refers to the attempt to replicate understanding and learning by means of an artifact, focusing primarily on thinking[6] or acting, as well as aiming for a rational ideal or replica of human capabilities.[7] In general, «*AI system*» denotes a computer environment applying AI and can also be described as a structured contextualized combination of

1 For a good overview on AI use cases in different fields see *Virdee*, p. 44 – 49.

2 This paper is inspired by (and the starting point for) a more comprehensive research project that aims to develop legal remedies for so-called adversarial attacks, deliberately misleading AI Systems with small perturbations. For a general introduction into the topic see *Goodfellow/Shlens/Szegedy* et al., passim.

3 *Hilty/Hoffmann/Scheuerer*, p. 50 ff.

4 For the corresponding risk of autonomy (*Autonomierisiko*) and further references, see *Zech*, Risiken, p. 27 ff.

5 For a comparable approach see *Zech*, Risiken, passim.

6 At the same time, however, *Surden* underlines the importance of not referring to AI as «thinking machines», see *Surden*, 89 Wash. L. Rev. 2014, p. 89.

7 For a recent overview see *Zanol/Buchelt/Tjoa/Kieseberg*, What is «AI»?, passim; *Drexl/Hilty/Beneke* et al., p. 3 broadly define AI as a branch of computer science; it «covers cognitive computing, machine learning (ML), evolutionary algorithms, rule-based systems, and the process of engineering intelligent machines»; *Lee/Hilty/Liu*, p. 2; for a commonly understandable definition see *Surden*, 35 Ga. St. U. L. Rev. 2019, p. 1307 – 1308: «In short, when engineers automate an activity that requires cognitive activity when performed by humans, it is common to describe this as an application of AI.»; see also *FDFA*, p. 4 describing AI as the «building or programming computers to do things that normally require human

«*AI techniques*»[8] with the goal of attaining artificial intelligence.[9] This is in line with the more specific definition of the European Commission, according to which AI systems are «software that [...] can, for a given set of human-defined objectives, generate outputs such as content, predictions, recommendations, or decisions influencing the environments they interact with».[10] With regard to the objectives the AI systems can pursue,[11] three main distinctions can be made: First, AI can be used for *classification* purposes. In this case, the AI system is used to correctly capture and label specific information[12] in a predictive manner.[13] Secondly, AI systems can be used to evaluate the information that has been extracted in order to make *predictions*. And third, AI can be used for *analytical* purposes, i.e. to understand the reasons behind certain actions.[14] However, this latter objective apparently still is elusive.[15]

For expanding the technical details, two main approaches within AI can be distinguished: Symbolic AI, describing forms of deduction and rule-based reasoning or so-called sub-symbolic AI, describing correlation-based reasoning.

Symbolic AI

Approaches based on *symbolic AI*[16] date back to the mid-1960s and are therefore often referred to as «Good Old-Fashioned Artificial Intelligence (GOFAI)».[17] This is also due to their reliance on classical logic.[18] The self-description of symbolic AI derives from the explicit manipulation of symbols, that these systems rely on.[19] Symbolic AI is based on the assumption that on the basis of the human communication through symbols, it will be possible to develop intelligent systems. The most prominent example within this field are *expert systems* that apply a predefined set of *if–then* rules to a given case. Therein, the elements and the relationships are explicitly represented using symbols. If, for example, an AI System has to learn specific properties of animals, a symbolic representation the model will consist of a *network of nodes* that are connected to each other. The nodes represent the animals, their properties, or capabilities, which are then related

---

or biological intelligence». As one of the first concrete legislations worldwide, in the Directive on Automated Decision-Making, which took effect on April 1, 2019, the Canadian Legislator defines AI as: «Information technology that performs tasks that would ordinarily require biological brainpower to accomplish, such as making sense of spoken language, learning behaviours, or solving problems.»

8 This paper aims to map these techniques by using the term AI System as a starting point. For an overview of AI techniques and approaches see also Annex I of the Proposal for a Regulation of the European Parliament and of the Council Laying Down Harmonized Rules on Artificial Intelligence (Artificial Intelligence Act) or *Braun Binder* et al., p. 3 ff.

9 See also *Deutscher Bundestag*, p. 51, highlighting the role of humans, that combine components of hard- and software.

10 Cf. Art 3 (1) of the Proposal for a Regulation of the European Parliament and of the Council Laying Down Harmonized Rules on Artificial Intelligence (Artificial Intelligence Act) – EU Commission, AI, p. 39.

11 For practical application areas of AI systems cf. *Zech*, Risiken, p. 18 – 19.

12 See *Zech*, Information, p. 13 ff. for an overview on the term as well as related concepts and legal approaches; *Zech*, Risiken, p. 8.

13 *Deng*, Antitrust Magazine 2018, p. 83; e.g. based on yes / no or 1 / 0, *Sheridan*, p. 113, table 7.2.

14 *Zech,* Risiken, p. 7.

15 *Moerland / Freitas*, p. 268 with reference to *Burgess*, p. 3 – 4.

16 *Tom Mitchell, Steve Muggleton* and *Ross Quinlan* are some of the proponents of this approach, see *Domingos,* p. 89 ff.; for an approach based on learning rules directly with rule induction see *Alpaydın*, Introduction, p. 230 f.

17 *Misselhorn*, p. 22; for a brief history of data science see *Kelleher / Tierney*, p. 6 ff.; see also *Alpaydın*, Introduction, p. 13 – 15.

18 *Domingos*, xvii; *Zech,* Risiken, p. 13 f.

19 *Saker / Zhou / Eberhart / Hitzler*, p. 1.

to each other.[20] Since its beginnings, symbolic AI has been further developed and combined with other approaches in the field. This trend, also known as neuro-symbolic AI, aims at bringing together the benefits of different approaches[21] such as the possibility to train from raw data (neural) as well as the high level of explainability (symbolic).[22]

Sub-Symbolic AI

In *sub-* or *non-symbolic approaches*, the objects are not represented in the model. Instead, the system consists of innumerable parameters which together determine the decisions and the behavior of the system. Whilst trainable parameters may evolve during the training (e.g. weights), hyperparameters (e.g. the architecture) are fixed.[23] Due to this interconnection between the parameters, this approach is also known as *connectionism*.[24]

As the human brain serves as a blueprint for the learning architecture, the most popular approaches are known as *Neural Networks*[25] which – if thoroughly trained – are able to (rightly) respond to questions, e.g. about the properties of animals. The system relies on various nodes (so-called *neurons*), which can basically be compared to electrical circuits or coded mathematical formulas, that are able to process the incoming signals.[26] After weighing each node with specific coefficients, the signals are processed from one layer to the next, eventually yielding an outcome. By weighing the nodes, the models can be fitted to the available data. This process is called *backpropagation.* Errors are fed back from the output[27] by propagating the error (i.e. the difference between actual and desired output) back through the output layer to the input layer.[28] In doing so, the weights of the neural connections are adapted depending on their influence on the error. When a new input is made, an approximation to the desired output can be attempted and the neural network «learns». This process can be automated by using a *backpropagation algorithm.*[29] As this approach proved to be flexible and versatile it has great success in various fields of application.[30] At the same time, however, these models run on large amounts of data. Providing this data is expensive and time-consuming.

---

20 *Baum*, No. 14.

21 For a comprehensive overview see *Saker/Zhou/Eberhart/Hitzler*, passim.

22 *Saker/Zhou/Eberhart/Hitzler*, p. 1.

23 *Drexl/Hilty/Beneke* et al., p. 6.

24 *Yann LeCun, Geoff Hinton* and *Yoshua Bengio* are attributed to this school of thought; *Domingos*, p. 93 ff. refers to them as connectionists; see also *Misselhorn*, p. 22; *Zech*, Risiken, p. 32.

25 Defined by *Virdee*, p. 41 as «an interconnected assembly of simple processing nodes or units whose functionality is loosely based on the animal neuron. The interunit connection strength captures the ability of the network to learn. Such interunit strengths can be obtained through adaptation or running an iterative set of training programs.»

26 *Misselhorn*, p. 23.

27 Defined as «The computer's confidence level or prediction based on statistical calculations.», *Sheridan*, p. 113, table 7.2.

28 *Alpaydın*, Introduction, p. 288.

29 Such a backpropagation algorithm is a supervised algorithm. After calculating the error for the neurons in the output layer, it uses the weight-update rule to update the weights. It then shares the error calculated at a neuron with the correspondent neurons in the precedent layer. See *Kelleher/Tierney*, p. 129 ff. Regarding supervised learning, see 2.2.2. infra.

30 *Baum*, No. 14.

Combined Approaches

So far, this distinction between symbolic AI and sub-symbolic AI appears intuitive. However, computer scientists have tried to further combine different models, in order to create rich cognitive[31] models.[32] Most of these combined models cannot be exclusively attributed to either symbolic AI or sub-symbolic AI. In general, these combinations of different models are also known as *ensembles*.

Gradient Boosted Decision Trees (GBDT) are among the most popular ensembles.[33] In GBDTs, decisions trees are used in order to sum the predictions out of a series of different trees. In order to approximate the prediction and the ground truth, a new decision tree is trained at every step.[34] This means, that GBDTs describe an approach aiming at minimizing the loss function, thus the difference in between the predicted and the actual values.[35] This is done by successively adjusting the values, e.g. of the weights. Thanks to their high accuracy and the ability to make fast predictions, GBDTs are frequently used in industry. At the same time, they do not perform well in training sessions with large datasets.[36] Furthermore, GBDT models cannot be incrementally updated with new data.[37]

Genetic programming[38], builds on evolutionary algorithms. These algorithms[39] are essentially based on evolutionary theories and natural selection.[40] In their core, algorithms which are based on this approach, solve learning processes within their structure. Hence, and unlike in a typical connectionist setup, the aim is not to adjust parameters by back-propagation, but to create a «brain» that can refine these adjustments using genetic programming.[41] It is a domain-independent method, which can be used to transform several computer programs into «new generations» by applying operations known from nature.[42] At the same time, however, humans still need to interpret and evaluate the solutions that the algorithm selected.[43] Whilst in a first step, potential solutions are created, in a second step these possible solutions are tested with regard to their ability to solve a task. The best solutions are then changed (mutated) or mixed (mated), before testing

---

31 Cf. *Zech*, Risiken, p. 8, who underlines the importance of information processing in both digital and natural information processing systems, which led to the further development of the cognitive sciences.

32 *Garcez / Besold / De Raedt* et al., p. 18 ff.; for an approach aiming at combining multiple models see *Alpaydın*, ML, p. 71; *Alpaydın*, Introduction, p. 533 ff.

33 *Müller / Guido*, p. 80.

34 *Zhang / Jung*, p. 1.

35 https://c3.ai/glossary/data-science/gradient-boosted-decision-trees-gbdt/.

36 *Saberian / Delgado / Raimond*, p. 1.

37 *Saberian / Delgado / Raimond*, p. 1.

38 *John Koza, John Holland* and *Hod Lipson* are attributed to this school of thought; *Domingos*, p. 51 ff. refers to them as Evolutionaries.

39 On a general level, algorithms can be defined as «a sequence of instructions that should be carried out to transform the input to an output», *Alpaydın*, Introduction, p. 1; see also the definition as «a well-defined sequence of computational steps for solving a problem. Specifically, it takes zero or more values as inputs and applies the sequence of steps to transform them into one or more outputs», *Man-Cho So*, p. 11; *Zech*, Risiken, p. 14. It is to note, that algorithms are commonly written in flowcharts or diagrams before the code is written, *Bainbridge*, p. 100.

40 For a review of nature-inspired algorithms see *Wunnava / Naik / Jena / Panda*, p. 3 ff.

41 For an introduction into the topic see *Goyal / Srivastava / Bisht*, p. 27 ff.

42 *Koza / Poli*, Genetic Programming, p. 127 ff.

43 *Hilty / Hoffmann / Scheuerer*, p. 54

them again.[44] This process is repeated until a specific degree of fitness is reached, although reaching 100 % seems rather unlikely.[45]

Bayesian statistics[46], another common approach, are based on a large number of hypotheses, which are uncertain in every case. Parameters are thus considered as random variables with a distribution that allows to model uncertainty.[47] Hence, probabilities are needed in order to quantify the validity of each hypothesis. The parameters of a probability distribution of a particular model can be inferred by using *Bayes' theorem* – a way of representing probabilities by taking into consideration prior conditions.[48] In practice, this approach to learning is frequently applied in spam classifiers or face recognition software. In general, the approach is best applied in small data sets.[49]

The so-called *analogy-based approach*[50] is based on the hypothesis that a useful analogy provides the information needed to solve a particular task. The effectiveness of this model has been confirmed by experimental data collected using a program that embodies this analogy theory. To this end, kernel functions – «smooth weight function(s)»[51] – are used. These functions determine the shape of the influences.[52] The fields of application of this approach are diverse and reach from medical diagnosis to recommendation engines.

2.1.2 Machine Learning (ML)

With the exception of the most basic expert systems, most of the above-mentioned approaches in AI also pertain to the field of «*Machine Learning (ML)*»[53]. ML is a discipline that started to flourish in the 1990 as a separate field within AI[54] and denotes a computer's ability to solve certain tasks with some degree of autonomy[55].[56] Generally, ML is based on heuristics.[57] This means that it is mainly correlation-based and uses a set of methods for determining probabilities and developing solutions

44 *Drexl / Hilty / Beneke* et al., p. 11.

45 *Slowinski*, p. 344

46 *David Heckerman, Judea Pearl* and *Michael Jordan* are attributed to this school of thought; *Domingos*, p. 167 f. refers to them as bayesians.

47 *Alpaydın*, Introduction, p. 491.

48 https://plato.stanford.edu/archives/spr2019/entries/bayes-theorem/.

49 *Alpaydın*, ML, p. 83.

50 *Peter Hart, Vladimir Vapnik* and *Douglas Hofstadter* are attributed to this school of thought; *Domingos*, p. 51 ff. refers to them as analogizers.

51 See *Alpaydın*, Introduction, p. 192 ff.

52 *Alpaydın*, Introduction, p. 193.

53 Cf. *FDFA*, p. 4. This includes genetic programming, GBDT and bayesian statistics, but also the approaches described in the following, namely ANN, DNN, CNN, random forests, and SVM. For a schematic overview of machine learning see *Virdee*, p. 41.

54 There is a debate whether ML is a subfield of AI or merely overlaps with AI, with some parts lacking the required «intelligence», see e.g. https://towardsai.net/p/machine-learning/differences-between-ai-and-machine-learning-1255b182fc6.

55 For a definition of autonomy see the European Parliament, Civil liability regime for AI, p. 22 – 23: Art. 3 (b): «means an AI-system that operates by interpreting certain input and by using a set of pre-determined instructions, without being limited to such instructions, despite the system's behaviour being constrained by, and targeted at, fulfilling the goal it was given and other relevant design choices made by its developer.»

56 Cf. *Surden*, 35 Ga. St. U. L. Rev. 2019, p. 1311 ff.

57 *Surden*, 35 Ga. St. U. L. Rev. 2019, p. 1308.

based on incomplete information as well as within a limited timeframe.[58]

ML has been described as an AI System that learns from experience with respect to some class of tasks and performance measure, its main feature being that the performance measure improves with increasing experience.[59] In ML, each model is chosen for a specific task and purpose.[60] However, whenever the area of expertise or the task changes, a different model has to be chosen and trained.[61] One exception to this general rule is *transfer learning*. If two networks are both trained on the same data – or different data but for a similar task – one can for example copy early convolutional layers and only train the later layers with the data of the specific task. The early layers that are copied act as preprocessor, which allows to map the data to a representation that might work out well.[62]

2.2 Elements of an AI System

An AI System consists of three defining features: First, a distinct architecture of the AI System; second, an algorithm that is responsible for «learning» by optimizing the performance measure; and third, the data input that serves as the basis for these learning processes. In order to attain a certain goal, all three components must be aligned or complement each other.

2.2.1 Architecture

With regard to the architecture of these systems, it is important to understand these building blocks in a non-physical sense. Instead, the use of this term relates to the fact that even non-physical AI Systems rely on specific structures. Accordingly, the most common AI architectures treated in the literature – namely *Artificial Neural Networks (ANN)*, *Deep Neural Networks (DNN)*, *Convolutional Neural Networks (CNN)*, *Recurrent Neural Networks (RNN)*, *Random Tree Forests* and *Support Vector Machines (SVM)* – can be characterized by their structure: They are either set up as a network (ANN, DNN, CNN, RNN), in the form of trees (or forests) or can be characterized by the space they occupy or delineate when performing regression tasks (SVM).

Artificial Neural Networks (ANN)

Artificial Neural Networks (ANN) are based on a regression – hence a supervised learning method[63] – whose origins date back to the 1940s.[64] Generally, ANNs are inspired by biological or natural neural networks (NNN), where the information is contained in the network itself and not in the format of symbols.[65] Above all, they try to imitate both the functions and activities of the human brain. It is to note, however, that this imitation is highly idealized.[66] One major difference is that biological

58 *Drexl/Hilty/Beneke* et al., p. 7; see also *FDFA*, p. 4, stating that it «facilitates predictions and classifications of (as yet unseen) data, which can assist with decision-making. Machine learning, which involves inductive reasoning, is the most important subset of AI. ML therefore deploys a 'data first' approach that uses an inductive process to learn from data.»

59 *Mitchell*, p. 2; *Zech*, Risiken, p. 28.

60 In that way, models are used to map inputs to predictions. That is why they are also known as «predictors», *Molnar*, GRUR Int. 2021, p. 10; see also *Alpaydın*, ML, p. 24–25. For a discussion, whether from a copyright perspective ML models can be defined as algorithms, computer programs or databases see *Otero*, GRUR Int. 2021, p. 1052 ff.

61 *Slowinski*, p. 343.

62 *Alpaydın*, Introduction, p. 338.

63 *Deng*, Antitrust Magazine 2018, p. 83.

64 *Mehlig*, p. 2 ff.

65 *Zech,* Risiken, p. 15.

66 *Mehlig*, p. 1.

systems rely on chemical reactions, whereas an ANN works based on electrical circuits.[67]

The system itself is built up in *layers*. It starts out with a data reception layer, passes data on to the input layer, on to one or several middle layers and, finally, to an output layer.[68] This latter is also known as classification layer.[69] Input is collected from outside the network, e.g. image pixels from a camera. Higher layers receive information from the lower layers. In the end, the highest layer will provide the output of the network for example, the statement that a specific object can be seen in a particular image. In that way, each layer plays a specific role in transforming the data into information.[70] In ANNs, the *neurons* – basically multi-input linear-regression functions[71] – , are interconnected and transmit data, consisting of 0s and 1s. These nodes are structured hierarchically.[72] Therein, synapses connect the different neurons. As each neuron has a specific weight, its role in the system varies accordingly.[73] By continuously changing the connections in between these different neurons, the ANN learns.[74] One of the strengths of ANNs is their ability to carry out several different tasks that involve information processing.[75] They are thus able to recognize specific structures in datasets.[76] ANNs can also generalize these results, in order to apply them to different sets of data.[77]

Deep Neural Networks (DNN)

Deep neural networks (DNN), which have been described as early as 2004,[78] are a sub-form of ANNs. In comparison to other ANNs, DNNs contain multiple layers between input and output layer.[79] There is no clear delineation of how many layers are needed in order for a ANN to classify as DNN.[80] However, the more hidden layers there are between the input and output layers, the more data can be processed.[81]

As with an ANN, the raw, visible information is recorded in an input layer. This information is then processed in subsequent layers[82], which – unlike in ANNs – are called hidden layers.[83] With every

67 For a comparison between the functioning of the brain and computers see *Alpaydın*, Introduction, p. 271 ff.

68 This basic model (input–processing–output) can be found almost all systems, see *Zech*, Risiken, p. 11, p. 42 ff.

69 *Misselhorn*, p. 23.

70 *Muhammad / Algehyne / Usman* et al., p. 7.

71 *Kelleher / Tierney*, p. 121.

72 *Misselhorn*, p. 23.

73 *Misselhorn*, p. 23.

74 *Mehlig*, p. 1. Basically, there are two types of learning which can be distinguished: learning by generalizing experiences made or by transforming a representation of a problem domain, *Zech*, Risiken p. 30 f.

75 *Mehlig*, p. 1.

76 *Mehlig*, p. 1.

77 *Mehlig*, p. 1.

78 *Lei* et al., p. 1 with reference to *Dalvi / Domingos / Sanghai / Verma*, p. 99 ff.; however, according to *Wiyatno* et al., p. 5, the current DNNs date back to 2011; cf. also *Huang* et al., p. 43 ff.

79 *Kelleher / Tierney*, p. 131.

80 *Drexl / Hilty / Beneke* et al., p. 6; for the question of how many hidden layers are needed see also *Mehlig*, p. 108 ff.; see *Sheridan*, p. 113, table 7.2, who refers to «more than one».

81 *Misselhorn*, p. 23.

82 *Söbbing*, MMR 2021, p. 112.

83 *Misselhorn*, p. 23; the term «hidden» simply describes that the neurons are neither in the input, nor in the output layer, *Kelleher / Tierney*, p. 124.

layer, the processed information is getting more and more abstract.[84] This abstraction of data provides better insights and hence allows for better solutions to problems which are challenging to deterministic algorithms.[85] As every layer of the DNN consists of multiple neuron layers, there are more possibilities to adapt the way the input information is handled throughout the system.[86] In the early days of DNNs, each layer was trained separately, which had the advantage that unlabeled data could be used. Then, fine-tuning was carried out with labeled data. Nowadays, the deep network is generally trained as a whole with an unsupervised method[87] to initialize the weights.[88] So far, DNN have been successfully applied in complex image recognition applications such as cancer detection and the identification of markers in genomes.[89] Furthermore, DNNs are used in inventive processes, especially in the field of drug discovery[90] and in representation learning where a machine is provided with unlabeled data and learns the representation by itself. This type of learning is characterized by the fact that the networks are able to learn a new representation of the input data, which is stronger in predicting the target output attribute than the original raw input.[91] DNNs have also become important in speech recognition systems because they can model large vocabularies and can robustly recognize speech independent from the actual speaker. This allows them to be used in voice interfaces. Accordingly, DNN-based systems are used more frequently in smartphones. Thanks to their features, DNNs are able to learn even the most complex structures and functions.[92] At the same time, as they contain more weights and processing units, one major challenge is that they require more computation. Because of the many hidden layers, training sessions are more challenging.[93] Moreover, DNNs are more difficult to interpret on the output level, which makes validation impossible.[94] As the network itself cannot explain the outcomes, humans will most often not be able to understand how the system reached them. AI Systems built on this architecture have therefore been called *black boxes.*[95] The fact that small perturbations of input data can be used to deceive ML models (by fabricating what is called an adversarial example) supports this observation. Such adversarial examples are «an indicator that the way such networks generalize from training examples is not completely understood, and it may just be that some of these deep networks work simply as gigantic lookup tables.»[96]

Convolutional Neural Networks (CNN)

In the 1980s, scientists developed so-called *Convolutional Neural Networks (CNN)*[97] that also

84 *Alpaydın*, Introduction, p. 314.

85 *Rani/Shanmugavadivu*, p. 198.

86 *Slowinski*, p. 344.

87 See 2.2.2. infra.

88 *Alpaydın*, Introduction, p. 315.

89 *Rani/Shanmugavadivu*, p. 198.

90 *Hilty/Hoffmann/Scheuerer*, p. 69.

91 *Kelleher/Tierney*, p. 134.

92 *LeCun/Bengio/Hinton*, Nature 521 (2015), p. 436.

93 *Alpaydın*, Introduction, p. 314; *Söbbing*, MMR 2021, p. 112.

94 *Alpaydın*, Introduction, p. 354.

95 For the concept and current status of explainable AI see *Gohel/Singh/Mohanty*, passim; also *FDFA*, 7; *Zech*, Risiken, p. 34.

96 *Alpaydın*, Introduction, p. 354.

97 *Mehlig*, p. 136.

belong to the deep neural networks. Different from common neural networks, these networks consist of fewer parameters and a smaller number of connections in between the neurons. They are hence easier to handle in training sessions and cheaper to train.[98] At the same time, however, the performance of CNN can be inferior to other systems.[99] In order to learn a greater number of abstract features, CNNs can be based on several convolution layers, sometimes even different in their type.[100] Furthermore, less connections in between the neurons also reduce the risk of *overfitting*.[101] This phenomenon occurs, when the model is perfectly customized to a training data set, but may not be applicable to unknown data.[102] As a result, the model is too complex and too specific for the amount of available information.[103]

CNNs are commonly applied in the field of image and pixel recognition.[104] In these systems the «units are fed with localized two-dimensional patches in the image»[105] and not a mere list of attributes.[106] This knowledge of primitives such as pixels and digits on different levels is used in order to understand and define the architecture, as well as the connectivity within that framework.[107] That is why CNNs are best used in situations where large learning capacities are needed, e.g. when it comes to facial recognition or image classification. As their depth and breadth can vary, they are able to make assumptions about the specific nature, e.g. of images.

Recurrent Neural Networks (RNN)

In *Recurrent Neural Networks (RNN)*, the neurons are interconnected in between the same or previous layers.[108] As the name suggests, these networks are based on «a feed-forward layout with feedbacks».[109] On a technical level this means that the output of a neuron is fed back into the neuron before proceeding to the next input. A main differentiation can be made in between direct, indirect and lateral feedback.[110] This feedback can be in between the neuron of the same or of a preceding layer. Thanks to this loop structure, a RNN is able to transmit information back into itself.[111] The network is thus able to take into consideration the inputs it has processed before, not only the current input.[112]

This particular feature of RNNs allows an application in cases, where patterns within data sequences – which can inter alia be genomes – need to be detected.[113] RNNs are also particularly useful in the

98 *Mehlig*, p. 136.

99 *Krizhevsky/Sutskever/Hinton*, p. 1097.

100 *Mehlig*, p. 137.

101 *Mehlig*, p. 136.

102 *Hurwitz/Kirsch*, p. 15.

103 *Müller/Guido,* p. 28.

104 For a closer look on convolutional neural networks see *Goodfellow/Bengio/Courville*, p. 326 ff.; see also *Alpaydın*, Introduction, p. 331 ff.

105 *Alpaydın*, Introduction, p. 314.

106 *Mehlig*, p. 136.

107 *Alpaydın*, Introduction, p. 314.

108 *Alpaydın*, Introduction, p. 345.

109 *Mehlig*, p. 150.

110 For other ways in which the feedback can act see *Mehlig*, p. 150.

111 *Schmidt*, p. 1.

112 *Kelleher/Tierney*, p. 133.

113 *Schmidt*, p. 1.

field of natural language processing and speech recognition.[114] At the same time, however, a key challenge in RNNs is to avoid vanishing *gradients.*[115] In this context, gradients – basically vectors – are used to explore how the outcome of a function will differ if inputs are changed. This means that in some cases, the gradient will be very small, which effectively prevents the weights from changing their value which may completely stop the training process. In order to solve this problem, scientists have tried to develop long short term memory units (LSTMs).[116] The aim of these LSTMs is to apply backpropagation-through-time in order to transform the networks into a feedforward network. In that way, the complexity of the learning behavior will increase.

Random Forests (RF)

Random forests (RF) can be described as a combination of several *decision trees*. In computer science, trees are used in order to decrease complexity by subdividing tasks in simpler subtasks. This is done by the structure which is composed of decision nodes and leaves, where at each decision a splitting test is carried out.[117] In random forests, each tree slightly differs from the others. Each of the single trees is dependent «on the values of a random vector sampled independently and with the same distribution for all trees in the forest.»[118] This means, that overall strength of this approach depends on the correlation between the single trees and their respective strength. Every decision tree is trained on subsamples of the training data and is able to make relatively precise decisions. Since the tree tends to yield decisions that are too specific, calculating an average for all the trees in the forest will diminish the issue of overfitting.[119] Based on mathematical methods, the prediction returned by the model for a specific query is the majority prediction across all the trees in the forest.[120] The decision trees are referred to as random, as the difference between the trees is based on an accidental decision.[121] Thus, decision trees consist of *if/else* rules,[122] where the goal is to «find a set of classification rules that divide the training data set into sets of instances that have the same value for the target attribute.»[123] The trees itself are based on «a hierarchical data structure implementing the divide-and-conquer strategy.»[124]

Whilst they are best applied for classification and regression,[125] they do not work well with numeric data and attributes,[126] as the number of values is infinite. This means that also the number of branches of the trees would need to be infinite.[127] In general, decision trees are nonparametric, as the structure is not set beforehand but depends on «the complexity

---

114 For some basic reading on recurrent neural networks see *Hochreiter/Schmidhuber*, 9 Neural Computation 1997, p. 1735 ff.

115 *Schmidt*, p. 3

116 *Schmidt*, p. 3 ff.

117 *Alpaydın*, ML, p. 77 – 79.

118 *Breiman*, p. 2.

119 See supra.

120 *Kelleher/Tierney*, p. 142.

121 *Bernard/Heutte/Adam*, p. 536; *Müller/Guido*, p. 80 ff.

122 *Müller/Guido,* p. 1 – 2.

123 *Kelleher/Tierney*, p. 136 – 137.

124 *Alpaydın*, Introduction, p. 217.

125 *Alpaydın*, Introduction, p. 217.

126 See 2.2.3. infra.

127 *Kelleher/Tierney*, p. 142.

of the problem underlying the data.»[128] In contrast, decision trees with gradient boosting are built up one after the other, which allows to work on the mistakes made earlier in the process. This method can both be applied to regression and to classification.[129] Unlike random forests, the decision trees in gradient boosting are relatively simple. Therefore, a high number of decision trees is needed in order to make good prediction.[130]

Support Vector Machines (SVM)

Support Vector Machines (SVM) are based on the implementation of principles from statistical learning theory, where the learning process occurs when a function is estimated out of training sets. This is done by choosing the function which is closest to the yet unknown function. SVM can be described as large margin classifiers, as they aim at keeping an area around the boundaries of a class free of objects. Within this approach, the dataset points are subdivided into groups with similar structures.[131] At the beginning, a set of training objects is needed, each of which assigned to a specific class. Each object is represented by a vector in a vector space. The distance of those vectors that are closest to the hyperplane – which separates the two classes – is maximized. This empty margin ensures that objects which do not correspond exactly to the training objects can be classified as reliable as possible. Thus, the main goal of SVMs is to separate different classes of data through mathematical optimization.[132]

When it comes to regression and (binary)[133] classification tasks,[134] support vector machines (SVM) are a common method in the field of ML. The SVM model is also known under the more generalized name *kernel machine*, which has become popular in the last few years.[135] On a general level, SVMs belong to the supervised learning models[136] and, before the development of deep learning, even outperformed ANNs in different occasions.[137] What makes SVM relevant for different fields of application is their high accuracy, whilst depending on less computational power and smaller datasets.[138]

2.2.2 Learning and Training Methods (Algorithms)

A second feature of an AI System applying ML is its selected learning and training method. In general, in order to train an AI System, a programmer has to set up an architecture which can then be used with training data and a training algorithm. The aim of training is to reduce the error rate.[139] The methods for training can be subdivided into online and offline based methods. Whereas the first approach describes a dynamic model where the optimization is continuously adapted, in an offline setting the models are

128 *Alpaydın*, Introduction, p. 218.

129 *Müller/Guido*, p. 85.

130 *Müller/Guido*, p. 85.

131 *Muhammad/Algehyne/Usman* et al., p. 7.

132 *Yeh*, p. 2.

133 *Yeh*, p. 14.

134 *Shmilovici*, p. 257.

135 *Alpaydın*, Introduction, p. 395.

136 See 2.2.2. infra.

137 *Yeh*, p. 2.

138 *Yeh*, p. 2.

139 See *Vargas*, p. 156

static.[140] In the latter, the optimization process and the actual application are separated and in the application phase, the weighting does not change.[141]

A more telling distinction, however, is made on the basis of the autonomy the AI Systems demonstrates during training. The learning and training methods can be separated in supervised machine learning (1), semi-supervised machine learning (2), unsupervised machine learning (3) and reinforcement machine learning (4). All of these approaches use algorithms to extract information from data.[142] We therefore refer to these learning and training methods as «*Algorithms*». However, to a smaller or larger extent, all of these Algorithms rely on human input.[143]

Supervised Machine Learning

Supervised learning, as the most common form of ML,[144] is widely used in different fields of practice.[145] As the term «supervised» suggests, the learning process is subject to human oversight.[146] Basically, it relies on the following steps:[147] the starting point typically consists of an established training data[148] set. This data, which is labeled by humans,[149] is used to predict the information that is missing in the test data.[150] During this classification process, the training samples are labelled within the specific category. In this way, it is clear how many groups there are and how similarities and differences can be defined.[151] In the following second step, the goal is to find a rule, which can be used to predict the labels of the test data samples.[152] This search of the learning algorithm for the best function is guided by checking how every single function matches with the data set.[153] During this process, so-called *loss functions* help to evaluate the performance. The accuracy of the prediction rule can be evaluated by comparing the predicted labels and the actual labels of the data samples.[154]

The aim of supervised learning is to maximize the accuracy of these predictions.[155] As a smaller loss function goes along with a better model, the aim of the training algorithm is to minimize the loss function. This is done by searching for the best combination of trainable parameters.[156] Hence,

---

140 *Drexl/Hilty/Beneke* et al., p. 6.

141 *Otero*, GRUR Int. 2021 p. 1051.

142 Cf. *Surden*, 35 Ga. St. U. L. Rev. 2019, p. 1315, who underlines the dependence of ML upon the availability of data.

143 *Man-Cho So*, p. 26; for the role of human beings in this context see *Zech*, Risiken, p. 12.

144 *LeCun/Bengio/Hinton*, Nature 521 (2015), p. 436.

145 *Müller/Guido*, p. 27; *Murphy*, p. 3; for a quick overview on different types of ML from a practitioners' perspective see *Sheridan*, p. 112, table 7.1.

146 *Hilty/Hoffmann/Scheuerer*, p. 54.

147 For the following see *Man-Cho So*, p. 13 ff.; *Zech*, p. 36.

148 Defined in Art. 3 (29) of the EC Proposal as «data used for training an AI system through fitting its learnable parameters, including the weights of a neural network», EU Commission, AI, p. 41.

149 *Hilty/Hoffmann/Scheuerer*, p. 54; *Deutscher Bundestag*, p. 52; this «human factor» is associated with some of the challenges in supervised ML. See *Man-Cho So*, p. 19.

150 *Man-Cho So*, p. 12.

151 *Man-Cho So*, p. 20.

152 *Man-Cho So*, p. 19.

153 *Kelleher/Tierney*, p. 100.

154 *Man-Cho So*, p. 19.

155 *Welsch/Eitle/Buxmann, 55 HMD Praxis der Wirtschaftsinformatik 2018,* p. 371.

156 *Drexl/Hilty/Beneke* et al., p. 7.

supervised ML is used to learn «a model of the relationship between a set of descriptive features and a target feature.»[157] In other words, the model is aligned by moving the predictions closer to the answers contained in the training data.[158]

One common example for the application of supervised ML is weather forecasting. By using a regression analysis, it takes into account known historical weather patterns and current conditions to make a prediction about the weather. Using the two methods – *classification* and *regression* – [159] a variety of learning tasks can be accomplished.[160] In *regression analysis*, the expected average value for a numeric target attribute is estimated, given that all input attributes are fixed. Hence, after a hypothesis has been made about the structure of relationship between the input attributes and the target, a parametrized model is defined. This is known as regression function.[161] Another example for regression is the prediction of prices for used cars. Whilst the input consist of the attributes such as the brand or capacity, the output is the expected price.[162] *Classification* instead, «involves estimating the value of a continuous attribute.»[163] The aim is to make a selection from a given list. A further example for classification is credit scoring, meaning the calculation of the risk when money is loaned to a customer. Here, based on past data – such as the financial history or the age of a person – the aim is to find a rule that can be applied to new fields.[164]

Besides its many advantages, supervised ML is confronted with several challenges. First of all, the creation of the data set, i.e. the collection of input-output pairs, can be very costly. Secondly, it is difficult to guarantee a balanced training set. This means, that it can be challenging to ensure that all output categories are represented equally. Thirdly, supervised learning is confronted with so-called *overfitting*.[165] And finally, working with labelled data may not only be impractical at times, but does often also not allow to uncover hidden structures.[166]

Semi-supervised Machine Learning (SSL)

During the training of *semi-supervised ML*, both labeled and non-labeled datasets are used and combined.[167] This can save some costs and time necessary for labelling the data.[168] Moreover, (partly) integrating unlabeled data into the machine learning process may enhance the learning performance.[169] However, good predictions purely based on unlabeled data remain challenging or almost impossible. The space of potential inference rules must thus be restricted from the very beginning.[170]

---

157 *Kelleher/Namee/D'Arcy*, p. 3.

158 *Hacker*, GRUR 2020, p. 1026.

159 For an introduction into these types of supervised learning see *Alpaydın*, ML, p. 38 ff.; *Virdee*, p. 37 ff., p. 40.

160 *Welsch/Eitle/Buxmann, 55 HMD Praxis der Wirtschaftsinformatik 2018,* p. 372.

161 *Kelleher/Tierney*, p. 114.

162 *Alpaydın,* Introduction, p. 9.

163 *Kelleher/Tierney*, p. 179; see also *Alpaydın*, Introduction, p. 4 ff. for practical fields of application.

164 *Alpaydın*, Introduction, p. 5.

165 *Alpaydın*, Introduction, p. 41; *Kelleher/Namee/D'Arcy*, p. 11; *Mehlig*, p. 101; for overfitting see also 2.2.1. supra; for a training method to overcome overfitting see *Zhang/Li*, p. 11.

166 *Man-Cho So*, p. 24.

167 *Muhammad/Algehyne/Usman* et al., p. 2; *Virdee*, p. 40.

168 *Zhou/Belkin*, p. 1239.

169 *Hady/Schwenker*, p. 215 ff.

170 *Zhou/Belkin*, p. 1239.

Unsupervised Machine Learning

In contrast to supervised ML, *unsupervised ML* develops a model based on non-classified data.[171] This means, that only input data exists, without any predefined output.[172] Different from supervised ML, in unsupervised learning neither labelling nor prediction rules are given. Instead, the aim is to first find hidden structures within the data.[173] Learning thus occurs without specifically preparing the data for the learning process. In order for this task to be successful, different methods are used to analyze data on as large a scale as possible. Algorithms are used in order to recognize hidden patterns or groupings of data that serve as a reference for the model.[174] Unsupervised machine learning thus performs an iterative process of data analysis virtually without human intervention and is able to recognize differences and similarities in given sets of information. The clusters can then help to inform the decision of human users.[175]

The learning algorithms used in unsupervised ML segment data into groups of examples (clusters) or groups of features. This process is also known as *clustering,*[176] which can either be used for data exploration or for mapping data to a new space in which it is easier to proceed with supervised learning.[177] During clustering, different types of clustering algorithms – such as exclusive, overlapping, hierarchical, and probabilistic algorithms – are applied. Whilst in exclusive clustering data points can only assigned to one specific cluster, in the overlapping approach these data points can belong to various clusters. Different from that, in hierarchical clustering, the data points are first clustered in different groups, for then merging them together, e.g. based on similarities. In a probabilistic model, instead, clustering is done by comparing the data points and deciding on the likelihood that they belong to a particular distribution.

A great advantage of clustering is that large amounts of unlabeled data can be used in a first step. Clustering may thus save the costs for labelling the data. In a second step, it is then possible to use a small amount of «labeled data to learn the second stage of classification or regression.»[178] At the same time and different from classification in supervised machine learning, in clustering it is unclear how many data sample groups there will be and how similarities and differences will be defined.[179] Finding ways to measure this similarity is one of the major challenges in unsupervised ML.[180]

One practical field of application of clustering is the analysis of a companies' dataset, in order to find out more about the distribution of customer profiles. With the help of the model, customers can be allocated to groupings which share similar

171 *Deutscher Bundestag*, p. 52.

172 *Alpaydın*, ML, p. 111.

173 *Alpaydın*, ML, p. 111; *Man-Cho So*, p. 19; *Lee/Hilty/Liu*, p. 3; *Zech*, Risiken, p. 37.

174 *Alpaydın*, Introduction, p. 11.

175 *Man-Cho So*, p. 20.

176 Known as mixture models in statistics, see *Alpaydın*, ML, p. 112 ff.; a further definition is the one provided by *Sheridan*, p. 113, table 7.2, who refers to clustering as a «powerful machine learning technique that is able to analyse large data sets and spot clusters from data that may seem to have no discernible structures within it.»

177 *Alpaydın*, Introduction, p. 177.

178 *Alpaydın*, Introduction, p. 178.

179 *Man-Cho So*, p. 20.

180 *Kelleher/Tierney*, p. 102–103.

attributes.[181] Once these groupings are found, the attributes can be further defined.[182]

Reinforcement Machine Learning

Lastly, there is *reinforcement ML*.[183] This approach is constructed as a behavior-based learning model in which interaction between a decision-making entity and its environment induces a learning process.[184] Accordingly, data from the learning environment plays a crucial role.[185] In contrast to the previously described methods in which the system observes and learns from a set of elements, reinforcement ML takes a different approach. It puts a decision-making agent[186] in its center.[187] This agent is placed in an *environment* which is in a certain *state* and is able to take specific *actions* changing the state.[188] As each action receives a reward – which defines the aim of the task –,[189] the overall goal of the system is to maximize these rewards.[190] The reward function has to strike the right balance between exploration and exploitation.[191] After a set of trial-and-error runs, the algorithm receives feedback from the analysis of the data. Once it has received feedback, knowledge grows.[192] This means, that a sequence of successful decisions results in the process being «reinforced», as it best solves the problem at hand.[193] The best policy will be the course of actions with the highest total reward.[194] As reinforcement learning is aligned to actions in the past, it is also defined as «learning with a critic».[195] Once the reward estimates of actions have reached a sufficient level, the exploitation can begin.[196]

Reinforcement machine learning is most commonly applied in the field of games or robotics.[197] With good reason: In order to train a robot to navigate a staircase, the robot has to change its approach to navigate the terrain, depending on the outcome of its actions. Based on similar reasons, reinforcement ML is also the main algorithm used for self-driving cars.[198] In game playing, several steps have to be taken in order to come to a good result.[199] This

181 *Alpaydın*, Introduction, p. 11.

182 *Alpaydın*, Introduction, p. 177.

183 For an introduction see *Sutton/Barto*, passim; *Zech*, p. 36–37.

184 *Man-Cho So*, p. 23; *Lee/Hilty/Liu,* p. 3.

185 *Hacker*, GRUR 2020, p. 1026.

186 The term refers to a decision-making entity such as a computer program, see *Man-Cho So*, p. 23.

187 *Deutscher Bundestag*, p. 53.

188 *Alpaydın*, Introduction, p. 563 f.

189 *Alpaydın*, ML, p. 127.

190 *Man-Cho So*, p. 24.

191 *Man-Cho So*, p. 24.

192 *Alpaydın*, ML, p. 128.

193 Thus, it loosely mimics a trial and error approach, cf. *Deng*, Antitrust Magazine 2018, p. 85; see also *Virdee*, p. 40.

194 *Alpaydın*, Introduction, p. 563.

195 *Alpaydın*, ML, p. 127.

196 *Alpaydın*, ML, p. 132.

197 For an overview on the impact of robots – basically IT-controlled machines – on our everyday lives as well as on the legal discussions see *Zech*, Risiken, p. 21 ff.

198 *Döbel et al.*, p. 33; *Hurwitz/Kirsch*, Machine Learning for Dummies, p. 17; *Welsch/Eitle/Buxmann*, 55 HMD Praxis der Wirtschaftsinformatik 2018, p. 366 ff.

199 For an application of deep learning in game theory see *Silver/Huang/Maddison* et al., p. 484 ff.

means, that not necessarily the single step counts, but the overall sum of steps.[200]

However, even within this field of application, many challenges remain. Complex surroundings make it difficult to find ways of rewarding the agent, although constructing the reward function is central. Besides, learning the optimal strategy may be costly; if it takes millions of trials to find a certain solution, reinforcement learning will not be the first choice.

2.2.3 Input Data

As a third and last component, an AI System applying ML depends on data.[201] While data used for building the ML learning model is called «*training data*», the data used to assess the prediction quality of the model is referred to as «*test data*».[202] Data in general is of paramount economic importance[203] and this is also true for training data.[204] The goal of the training process is to improve the predictions for the classification of unclassified data. As with a higher quality of the training data, the performance of the AI System's output will improve,[205] choosing the right training data is crucial.[206]

Depending on the field of application, training data have different formats. A first distinction can be made between *unclassified training data* and training data that has already been *classified*. As data are classified with labels (attached by humans[207]), unclassified data is also called unlabeled data. Classified data is also called labeled data.

Moreover, data can exist in an *unstructured, semi-structured* or *structured* form. Structured data is frequently categorized as quantitative data with a predefined format, often in a table with rows and columns,[208] whereas unstructured data is commonly organized as qualitative data, with no identifiable data structure.[209] A further differentiation can be made between the set of attributes within the data. The values of these attributes can be continuous, categorical or binary.[210] Categorical (or nominal) attributes receive the values from a finite set, where the values are names for categories. In a binary attribute this set of values of limited to the two values true and false.[211] In numeric attributes the quantities are measurable.[212] With regard to the quality of the data, the data set needs to be balanced. Any bias in the data set will be reproduced in the application of the trained AI System.

---

200 *Alpaydın*, Introduction, p. 12.

201 On a general level, data can be defined as symbolically represented, machine-readable information, Zech, Risiken, p. 14; for an overview of which data (directly identifiable, indirectly identifiable, de-identified and/or non-personal) is used in which phase (acquisition, analysis and application) see *Oostveen*, 6 International Data Privacy Law 2016, p. 306 ff.; see also *Zech*, GRUR 2015, p. 1152.

202 *Müller/Guido,* Einführung in Machine Learning mit Python, p. 17–18. For a definition of «testing data», see Commission (EU), Art. 3 (31).

203 *Zech*, GRUR 2015, p. 1151; for an overview into the status quo and challenges within data law, with a focus on Switzerland see *Thouvenin/Weber/Früh*, passim.

204 *Deutscher Bundestag*, p. 33.

205 *Man-Cho So*, p. 12.

206 *Man-Cho So*, p. 12.

207 For risks associated with this «human factor» see *Man-Cho So*, p. 19.

208 *Winter/Battis/Halvani*, ZD 2019, p. 490; *Sarre/Pruß*, § 2 Daten, Datenbanken und Datensicherheit, No. 30–32.

209 *Sarre/Pruß*, § 2 Daten, Datenbanken und Datensicherheit, No. 44–45.

210 *Welsch/Eitle/Buxmann, 55 HMD Praxis der Wirtschaftsinformatik 2018,* p. 366 ff.

211 *Kelleher/Tierney*, p. 43.

212 *Kelleher/Tierney*, p. 42.

A further common distinction is made between *personal* and *non-personal data*. Often, the data used for the training are «banal» or simple «by-products of technology usage»,[213] thus mostly abstract content, other data can be linked to individuals.[214] If this data is used for training models, it must be ensured that non-personal, anonymized data is used.[215] Whether anonymized data can be used depends on the process.[216] If this is the case, most data protection regulations do not apply.[217] However, if the data must first be anonymized,[218] the procedure of anonymization itself may subject to under data protection law.[219]

Data can further be subdivided into *raw data, real-time data, secondary data, synthetic data* and *metadata.*[220] Raw data is data which is directly derived from observations or statistical surveys. It is important to point out that not even raw data is 'objective'. Even collecting raw data means to make decisions about what to abstract from and how to measure. As a consequence, most data is partial and biased.[221] Real-time data, as a specific form of raw data, is collected in a specific timeframe. In the industry, this is used to draw conclusions about a certain time. These two types of raw data are said to be of particular importance for AI Systems. They seem to allow AI Systems to draw conclusions about correlations that may elude human perception.[222] Secondary data instead, is data that was collected in a first step and then processed it in a second. As a subtype of secondary data, synthetic data is data that was artificially created through data processing. In that way, large amounts of data can be generated, in order to test and train AI Systems.[223]

2.3 Trained AI-System

Once a AI System is trained and ready for application, we refer to it as «*ML Model*». The ML Model is unique and a result of the final configuration of all elements of the AI System: the Architecture, Algorithms and the Input Data that was used to train the system. This means that the number of layers of a neural network is set and the nodes of each layer within the network are weighed. Similarly, the random tree forest, i.e. the number of decision trees and their respective properties, is set up or the SVM is defined with all its margin classifiers.

All ML Models have at their core one or several mathematical functions that assign an output value or output category to any given new input data. This mathematical function is called «*Classifier*». The literature sometimes also refers to this mathematical function as a (classification) algorithm.[224] However, to avoid confusion with the

213 *Sobel*, p. 224.

214 For the consequences thereof see *Deutscher Bundestag*, p. 67 ff.

215 However, see *Deutscher Bundestag*, p. 57 highlighting the possibility that even from non-personal data inferences about the individuals can be made in the future.

216 See e.g. for the calibration of algorithmic models *Hacker*, ZfPW 2019, p. 152.

217 *Deutscher Bundestag*, p. 57.

218 For some common anonymization techniques see *Valkanova*, No. 32 ff.

219 *Valkanova*, No. 12 – 14; see also *Deutscher Bundestag*, p. 33 calling for the implementation of «Trust-Centers» in order to enable interdisciplinary and trustworthy exchange of data, see also p. 57.

220 For the following see *Deutscher Bundestag*, p. 55 – 56.

221 *Kelleher / Tierney*, p. 46.

222 *Deutscher Bundestag*, p. 55.

223 *Deutscher Bundestag*, p. 55.

224 *Kotsiantis / Zaharakis / Pintelas*, Artif. Intell. Rev. 2006, p. 159 ff.

Algorithms used for learning and training, we propose to use the term Classifier. The Classifier is the essence of the ML Model. It allows a trained AI system to classify new data or to predict outcomes for new data.

3 Summary and Outlook

This contribution tries to sharpen the contours of several basic terms in the realm of AI and ML. In order to do so, we have defined AI Systems as computer environment applying AI techniques.[225] These AI techniques are best described by distinguishing the three core features of every AI System: its Architecture, its Learning and Training Method (which we call Algorithm) and its Input Data used for training.[226] The trained AI System can be called ML Model. It has, at its core, a Classifier, a mathematical function that assigns an output to any given input.[227]

A closer look at the different elements of AI Systems shows that every Architecture and every Algorithm has unique properties and therefore also its strengths and weaknesses: Some Architectures perform better with more data, some with less. Some setups may work with unlabeled data, some do not. Moreover, certain Architectures require a certain type of Input Data. And finally, some approaches may be combined as ensembles, whereas others may not.

The technical development in AI and ML is fast-paced and the state of the art is constantly evolving. Nevertheless, obtaining a clear picture of the current state of the art is crucial. Both building on basic terms and outlining the specifics of this technology in more detail allow legal academia to make reliable legal statements and to advance the research in this field.

225 See supra 2.1.

226 See supra 2.2.

227 See supra 2.3.

References

Alpaydın, Ethem, Machine Learning: The New AI, Cambridge, MA: MIT Press 2016 (cited: ML)

Alpaydın, Ethem, Introduction to Machine Learning, Fourth Edition, Cambridge, MA: MIT Press 2020 (cited: Introduction)

Bainbridge, David I., Information Technology and Intellectual Property Law, Seventh Edition, London: Bloomsbury Professional, 2019

Baum, *Lothar*, Teil 9.1 Technische Grundlagen, in: Andres Leupold / Andreas Wiebe / Silke Glossner (eds.), Münchener Anwaltshandbuch IT-Recht, 4. Auflage 2021

Bernard, Simon / Heutte, Laurent / Adam, Sébastien, Towards a Better Understanding of Random Forests through the Study of Strength and Correlation, in: De-Shuang Huang / Kang-Hyun Jo / Hong-Hee Lee / Hee-Jun Kang / Vitoantonio Bevilacqua (eds.), Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence. ICIC 2009. Lecture Notes in Computer Science, Vol. 5755, Berlin / Heidelberg: Springer, https://doi.org/10.1007/978-3-642-04020-7_57, *p. 536 – 545*

Bisht, Dinesh C. S. / Ram, Mangey (eds.), Computational Intelligence: Theoretical Advances and Advanced Applications, Berlin / Boston: De Gruyter 2020

Braun Binder, Nadja / Burri, Thomas / Lohmann, Melinda Florina / Simmler, Monika / Thouvenin, Florent / Vokinger, Kerstin Noëlle, Künstliche Intelligenz: Handlungsbedarf im Schweizer Recht, Jusletter v. 28. Juni 2021

Breiman, Leo, Random Forests, 2001, https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf

Burgess, Andrew, The Executive Guide to Artificial Intelligence: How to Identify and Implement Applications for AI in Your Organization, Cham: Palgrave Macmillan 2018

Commission (EU), Proposal for a Regulation of the European Parliament and of the Council Laying Down Harmonized Rules on Artificial Intelligence (Artificial Intelligence Act) and Amending Certain Union Legislative Acts, 21.04.2021, Brussels (cited: EU Commission, AI)

Dalvi, Nilesh / Domingos, Pedro / Sanghai, Sumit / Verma, Deepak, Adversarial classification, in: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM 2004, pp. 99–108, https://doi.org/10.1145/1014052.1014066

Deng, Ai, An Antitrust Lawyer's Guide to Machine Learning, Antitrust Magazine 2018, p. 82 – 87

Deutscher Bundestag, 19. Wahlperiode, Drucksache 19 / 23700, Unterrichtung und Bericht der der Enquete-Kommission Künstliche Intelligenz – Gesellschaftliche Verantwortung und wirtschaftliche, soziale und ökologische Potenziale, 28.10.2020, https://www.aaai.org/ocs/index.php/SSS/SSS15/paper/view/10281/10029

Döbel, Inga / Leis, Miriam / Vogelsang, Manuel Molina / Neustroev, Dmitry / Petzka, Henning / Rüping, Stefan / Voss, Angelika / Wegele, Martin / Welz, Juliane, Maschinelles Lernen – Kompetenzen, Anwendungen und Forschungsbedarf, Fraunhofer Institut, 2018

Domingos, Pedro, The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World, London: Penguin Books 2015

Drexl, Josef / Hilty, Reto M. Hilty / Beneke, Francisco / Desaunettes, Luc / Finck, Michèle Globocnik, Jure / Otero, Begoña Gonzalez / Hoffmann, Jörg / Hollander, Leonard / Kim, Daria / Richter, Heiko / Scheuerer, Stefan / Slowinski, Peter R. / Thonemann, Jannick, Technical Aspects of Artificial Intelligence: An Understanding from an Intellectual Property Law Perspective, Max Planck Institute for Innovation and Competition Research Paper No. 19-13, October 2019, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3465577

Federal Department of Foreign Affairs (FDFA, Switzerland), Artificial intelligence and international rules: Report for the Federal Council, April 13th, 2022, https://www.newsd.admin.ch/newsd/message/attachments/71099.pdf

Garcez, Artur S. d'Avila / Besold, Tarek R. / De Raedt, Luc / Földiák, Peter / Hitzler, Pascal / Icard, Thomas / Kühnberger, Kai-Uwe / Lamb, Luís C. / Miikkulainen, Risto / Silver, Daniel L., Neural-Symbolic Learning and Reasoning: Contributions and Challenges. AAAI Spring Symposia 2015, Stanford, AAAI Press, p. 18 ff., https://www.aaai.org/ocs/index.php/SSS/SSS15/paper/view/10281/10029

Gohel, Prashant / Singh, Priyanka / Mohanty, Manoranjan, Explainable AI: current status and future directions, arXiv:2107.07045, 2021

Goodfellow, Ian / Bengio, Yoshua / Courville, Aaron, Deep Learning. Adaptive Computation and Machine Learning, MIT Press 2016, www.deeplearningbook.org

Goodfellow, Ian J. / Shlens, Jonathon / Szegedy, Christian, Explaining and harnessing adversarial examples, arXiv preprint arXiv:1412.6572, 2014

Goyal, Gunjan / Srivastava, Pankaj Kumar / Bisht, Dinesh C.S., Genetic algorithm: a metaheuristic approach of optimization, in: Bisht / Ram (eds.) (cf there), 27 – 4

Hacker, Philipp, Daten als Gegenleistung: Rechtsgeschäfte im Spannungsfeld von DS-GVO und allgemeinem Vertragsrecht, ZfPW 2019, p. 148 – 197

Hacker, Philipp, Immaterialgüterrechtlicher Schutz von KI-Trainingsdaten, GRUR 2020, p. 1025 – 1033

Hady, Mohamed Farouk A. / Schwenker, Friedhelm, Semi-supervised Learning, in: Monica Bianchini / Marco Maggini / Lakhmi L. Jain (eds.), Handbook on Neural Information Processing. Intelligent Systems Reference Library, Vol 49, Berlin / Heidelberg: Springer 2013, https://doi.org/10.1007/978-3-642-36657-4_7, p. 215 – 239

Hilty, Reto M / Hoffmann, Jörg / Scheuerer, Stefan, Intellectual Property Justification for Artifical Intelligence, in: *Lee / Hilty / Liu* (eds.) (cf there), p. 50 – 72

Hochreiter, Sepp / Schmidhuber, Jürgen, Long short-term memory, in: 9 Neural Computation 1997, p. 1735 – 1780

Huang, Ling / Joseph, Anthony D. / Nelson, Blaine / Rubinstein, Benjamin I. P. / Tygar, J. D., Adversarial Machine Learning, in: Proceedings of 4th ACM Workshop on Artificial Intelligence and Security, October 2011, p. 43 – 58, https://dl.acm.org/doi/pdf/10.1145/2046684.2046692

Hurwitz, Judith / Kirsch, Daniel, Machine Learning for Dummies, Hoboken, NJ: John Wiley & Sons, 2018

Kelleher, John D. / Namee, Brian Mac / D'Arcy, Aoife, Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies, Cambridge, MA / London: MIT Press 2015

Kelleher, John D. / Tierney, Brendan, Data Science, Cambridge, MA: MIT Press 2018

Kotsiantis, Sotiris B. / Zaharakis, Ioannis D. / Pintelas, Panagiotis E., Machine learning: a review of classification and combining techniques, Artif. Intell. Rev. 2006, p. 159 – 190

Koza, John R. / Poli, Riccardo, Genetic Programming, in: Edmund K. Burke / Graham Kendall (eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, New York: Springer 2005, p 127 – 164

Krizhevsky, Alex / Sutskever, Ilya / Hinton, Geoffrey E., ImageNet classification with Deep Convolutional Neural Networks, Advances in Neural Information Processing Systems 25, p. 1097 – 1105

LeCun, Yann / Bengio, Yoshua / Hinton, Geoffrey, Deep learning, Nature 521, Nature 2015, p. 436 – 444, p. 436 – 444

Lee, Jyh-An / Hilty, Reto M / Liu, Kung-Chung (eds.), Artifical Intelligence and Intellectual Property, Oxford: Oxford University Press 2021

Lei, Qi / Wu, Lingfei / Chen, Pin-Yu / Dimakis, Alexandros G. / Dhillon, Inderjit S. / Witbrock, Michael, Discrete Adversarial Attacks and Submodular Optimization with Applications to Text Classification, arXiv:1812.00151, 2019

Lee, Jyh-An / Hilty, Reto M / Liu, Kung-Chung, Roadmap to Artificial Intelligence and Intellectual Property: An Introduction, in: *Lee / Hilty / Liu*, (eds.) (cf. there), p. 1 – 7

Man-Cho So, Anthony, Technical Elements of Machine Learning for Intellectual Property Law, in: *Lee / Hilty / Liu*, (eds.) (cf there), p. 11 – 27

Mehlig, Bernhard, Machine Learning with neural networks, arXiv:1901.05639, 2021

Misselhorn, Catrin, Grundfragen der Maschinenethik, Ditzingen: Reclam 2018

Mitchell, Tom M., Machine Learning, New York: McGraw-Hill 1997

Moerland, Anke / Freitas,Conrado, Artificial Intelligence and Trademark Assessment, in: *Lee / Hilty / Liu*, (eds.) (cf. there), p. 266 – 291

Molnar, Christoph, Interpretable Machine Learning: A Guide for Making Black Box Models Interpretable, 2021, https://christophm.github.io/interpretable-ml-book/

Muhammad, L.J. / Algehyne, Ebrahem A. / Usman, Sani Sharif / Ahmad, Abdulkadir / Chakrabort, Chinmay / Mohammed, I. A., Supervised Machine Learning Models for Prediction of COVID-19 Infection using Epidemiology Dataset, SN COMPUT. SCI. 2, 11 «Sn Comput. Sci.2021», https://doi.org/10.1007/s42979-020-00394-7, p. 1 – 13

Müller, Andreas C. / Guido, Sarah, Einführung in Machine Learning mit Python, Praxiswissen Data Science, Heidelberg: dpunkt.verlag 2017

Murphy, Kevin P., Machine Learning: A Probabilistic Perspective, Cambridge, MA / London: MIT Press 2012

Oostveen, Manon, Identifiability and the applicability of data protection to big data, in: International Data Privacy Law, Vol. 6, Issue 4, 2016, p. 299 – 309

Otero, Gonzalez, Machine Learning Models under the copyright microscope: is EU Copyright fit for purpose?, GRUR Int., 2021, p. 1043 – 1055

Parliament (EU), Civil liability regime for artificial intelligence: European Parliament resolution of 20 October 2020 with recommendations to the Commission on a civil liability regime for artificial intelligence (2020 / 2014(INL)), P9_TA(2020)0276, 2020

Rani, M. Mary / Shanmugavadivu, P., Deep learning based food image classification, in: Bisht / Ram (eds.) (cf there), 197 – 207

Saberian, Mohammad / Delgado, Pablo / Raimond, Yves, Gradient Boosted Decision Tree Neural Network, arXiv:1910.09340, 2019

Saker, Md Kamruzzaman / Zhou, Lu / Eberhart, Aaron / Hitzler, Pascal, Neuro-Symbolic Artificial Intelligence, arXiv:2105.05330, 2021

Sarre, Frank / Pruß, Michael, § 2 Daten, Datenbanken und Datensicherheit, in: Astrid Auer-Reinsdorff / Isabell Conrad (eds.), Handbuch IT- und Datenschutzrecht, 3rd Edition, 2019, No. 44 – 45

Sheridan, Iain, Commercial Contracts, in: Kerrigan, Artificial Intelligence: Law and Regulation, Cheltenham: Edward Elgar 2022, p. 109 – 132.

Schmidt, Robin M., Recurrent Neural Networks (RNNs): A gentle Introduction and Overview, arXiv:1912.05911, 2019

Shmilovici, Armin, Support Vector Machines, in: Oded Maimon / Lior Rokach (eds.), Data Mining and Knowledge Discovery Handbook, Boston, MA: Springer 2005, https://doi.org/10.1007/0-387-25465-X_12, p. 257 – 276

Silver, David / Huang, Aja / Maddison, Chris J., et al., Mastering the game of Go with deep neural networks and tree search, Nature 2016, p. 484 – 489, https://doi.org/10.1038/nature16961

Slowinski, Peter R, Rethinking Software Protection, in: *Lee / Hilty / Liu* (eds.) (cf there), p. 341 – 361

Sobel, Benjamin, A Taxonomy of Training Data: Disentangling the Mismatched Rights, Remedies, and Rationales for Restricting Machine Learning, in: *Lee / Hilty / Liu*, (eds.) (cf there), p. 221 – 242

Söbbing, Thomas, Künstliche neuronale Netze: Rechtliche Betrachtung von Software- und KI-Lernstrukturen, MMR 2021, p. 111 – 116

Surden, Harry, Artificial Intelligence and Law: An Overview, 35 Ga. St. U. L. Rev. 2019, p. 1305 – 1337, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3411869

Surden, Harry, Machine Learning and Law, 89 Wash. L. Rev. 2014, p. 87 – 115, https://digitalcommons.law.uw.edu/cgi/viewcontent.cgi?article=4799&context=wlr

Sutton, Richard S. / Barto, Andrew G., Reinforcement Learning: An Introduction, Cambridge, MA / London: MIT Press, Second Edition, 2014 / 15

Thouvenin, Florent / Weber, Rolf H. / Früh, Alfred, Elemente einer Datenpolitik, Zürich: Schulthess / Nomos 2019

Valkanova, Monika, Kapitel 8.1: Trainieren von KI-Modellen, in: Markus Kaulartz / Tom Braegelmann, Rechtshandbuch Artificial Intelligence und Machine Learning, 1. Ed. 2020, No. 1 ff.

Vargas, Danilo Vasconcellos, Learning Systems Under Attack – Adversarial Attacks, Defenses and Beyond, in: Steven Van Uytsel / Danilo Vasconcellos Vargas (eds.), Autonomous Vehicles: Business, Technology and Law, Singapore: Springer 2021, p. 147 – 161

Virdee, Tirath, Understanding AI, in: Kerrigan, Artificial Intelligence: Law and Regulation, Cheltenham: Edward Elgar 2022, p. 37 – 55.

Welsch, Andreas / Eitle, Verena / Buxmann, Peter, Maschinelles Lernen: Grundlagen und betriebswirtschaftliche Anwendungspotenziale am Beispiel von Kundenbindungsprozessen, 55 HMD Praxis der Wirtschaftsinformatik 2018, p. 366 – 382

Winter, Christian / Battis, Verena / Halvani, Oren, Herausforderungen für die Anonymisierung von Daten, in: ZD 2019, p. 489 – 493

Wiyatno, Rey Reza / Xu, Anqi / Dia, Ousmane / de Berker, Archy Adversarial Examples in Modern Machine Learning: A Review, arXiv:1911.05268, 2019

Wunnava, Aneesh / Naik, Anee Manoj Kumar / Jena, Bibekananda / Panda, Rutuparna, Nature-inspired optimization algorithm and benchmark functions: a literature survey, in: Bish / Ram (eds.) (cf there), 3 – 26

Yeh, Wei-Chang, Convolutional Support Vector Machine, arXiv:2002.07221, 2020

Zanol, Jakob / Buchelt, Alexander / Tjoa, Simon / Kieseberg, Peter, What is «AI»?, Jusletter IT, 24.2.2022

Zech, Herbert, Risiken Digitaler Systeme: Robotik, Lernfähigkeit und Vernetzung als aktuelle Herausforderungen für das Recht. (Weizenbaum Series, 2), Berlin: Weizenbaum Institute for the Networked Society - The German Internet Institute (2020), https://doi.org/10.34669/wi.ws/2 (cited: Risiken)

Zech, Herbert, «Industrie 4.0» – Rechtsrahmen für eine Datenwirtschaft im digitalen Binnenmarkt, GRUR 2015, p. 1151 – 1160

Zech, Herbert, Information als Schutzgegenstand, Tübingen: Mohr Siebeck 2012, https://doi.org/10.1628/978-3-16-152162-1 (cited: Information)

Zhang, Zhendong / Jung, Cheolkon, GBDT-MO: Gradient Boosted Decision Trees for Multiple Outputs, arXiv:1909.04373v2, 2019

Zhang, Jiliang / Li, Chen, Adversarial examples: Opportunities and challenges, arXiv:1809.04790, 2019

Zhou, Xueyuan / Belkin, Mikhail, Semi-Supervised Learning, Academic Press Library in Signal Processing Vol. 1, 2014, https://doi.org/10.1016/B978-0-12-396502-8.00022-X, p. 1239 – 1269