Definition of AI

Steckelmacher, Denis; Bogaerts, Bart

Published in:
The European Artificial Intelligence Act

DOI:
10.1007/978-3-031-98406-8

Publication date:
2025

Definition of AI

Denis Steckelmacher[1] and
Bart Bogaerts[2]

Abstract

Artificial Intelligence is present in human imagination since Antiquity, but is considered to have been founded as a research field only in 1956 during the Darthmouth Summer Research Project on Artificial Intelligence[01]. While this workshop identified a set of tasks and approaches that are Artificial Intelligence, a strong definition, that maintains its accuracy over decades, has always remained elusive. In this chapter, we will explore the area of Artificial Intelligence Research, how AI is often defined, and then discuss possible other definitions.

1 Computer Science

Artificial Intelligence is part of the broader field of Computer Science, the science of designing and working with computers. We provide a general introduction to Computer Science in this section, to better position Artificial Intelligence in the world, and its relation with technology in general. In the next section, we explore Artificial Intelligence Research, before, in Sections 3 and 4, discussing how to precisely define AI.

The computers, phones, smart appliances and machines that we use every day, or that make the world run, are not harvested from nature but are human-made machines. These machines have two essences that closely interact with each other:

1. The hardware, the physical realization and existence of the machine. The hardware encompasses the silicon used to manufacture microprocessors, plastics used for the keyboard, the chemistry of screens, the metal of cables, etc. Improving hardware requires research and development in electronics, chemistry, physics and engineering in general. Computer Science usually does not try to make hardware better.

2. The software, that runs on the hardware. Software is made of algorithms and data, and we discuss in the next section what an algorithm is. The same hardware can run an infinite variety of software, which explains why we can update the software on a computer, or install a new program, without having to buy a new computer. Computer Science is the science of software, and mainly revolves around making better software, or using software to tackle the problems of the world.

1.1 Algorithms

Algorithms are not limited to computers. The word algorithm itself is more than a millennium old, and originates from the name and work of *al-Khwārizmī*, a well-respected Persian mathematician from the IX[th] century[02].

---

1 Denis Steckelmacher
Denis Steckelmacher, Pleinlaan 9, 1050 Ixelles, Belgium, denis.steckelmacher@vub.be
2 Bart Bogaerts
Bart Bogaerts, Pleinlaan 9, 1050 Ixelles, Belgium, bart.bogaerts@vub.be

An algorithm is the description of a step-by-step process that allows a person, a machine or a computer to perform some task, by combining smaller and easier tasks that the person, machine or computer is already able to perform.

Example of an algorithm

The best-known algorithm, that almost everyone learns in high school, is the algorithm that allows to find the roots of a second-degree equation. We briefly review this algorithm in this section, because it is a clear example of an algorithm that can be performed both by a person, and by a computer.

We consider the equation below:

$$ax^2 + bx + c = 0$$

with a, b and c given (we assume that we know these numbers by the time we want to perform the algorithm). The purpose of the algorithm is, given a, b and c, to find the x values for which the equation above holds, so for which $ax^2 + bx + c$ is equal to zero.

The algorithm that we learn in high-school assumes a person (or a computer) who is able to add numbers, multiply numbers, divide numbers, and compute the square root of a number. This last requirement is in practice challenging to obtain in the real world, and people usually defer to a calculator for computing the square root, while computers implement an additional algorithm, that allows them to compute a square root by only performing additions, multiplications and divisions. The algorithm is as follows:

1. We compute a number, which we call “delta”, that is equal to $b^2 - 4ac$. Because we know a, b and c, it is possible to compute delta exactly and to write it on a piece of paper.
2. If delta is negative, it is impossible to find a value of x that satisfies the equation, so we simply write “no root” on the piece of paper.
3. If delta is positive, it allows us to compute the two roots of the equation: $\frac{-b+\sqrt{delta}}{2a}$ and $\frac{-b-\sqrt{delta}}{2a}$.
4. If the two roots are the same (which happens when delta is exactly zero), we usually consider that there is only a single root for the equation.

Algorithms and Computer Science

The above algorithm is expressed in a human readable form, in English. Computer Science mainly revolves around designing and expressing algorithms:

- How can a given problem, that matters in the real world, be expressed as an algorithm that uses simple-enough constructs to be performed by a computer?
- How can an algorithm be expressed in a computer language in a way that is correct and fast?

Computers do not understand English, but, over the decades, a variety of tools have been designed to allow humans (programmers) to express algorithms in a language that is understandable both by trained experts (the programmers) and a machine. These languages are called “programming languages”, and more than a thousand of them exist, with varied properties, ease of use, acceptance by the field, and speed considerations.

The algorithm is the mathematical concept of what has to be done (and can be expressed in English like the example above). A program is the expression of an algorithm in some programming language, also called the *implementation* of the algorithm. That expression is usually stored in a computer file, comparable to a Word document but much simpler. Programmers can open the file to modify it and improve it. The computer itself can also open the program to *run* it, to perform the algorithm it describes. Programs, both readable and modifiable by a person and runnable by a computer, are at the core of Computer Science. They are the tangible expression of an algorithm, and are being referred to when some regulation mentions something like “the implementation of an AI system”.

1.2 *Research areas in Computer Science*

Artificial Intelligence is one of the research areas in Computer Science, but not the only one. Other research areas either consider how to improve the computers themselves, or how to use computers for the improvement of the world. Focusing on the areas that aim at improving how computers work, we start with:

- Cryptography and Security in Computing in general: algorithms that allow to transfer or store information securely between two people or pieces of software, even if they are separated by a potentially-hostile network or other equipment. Cryptography also considers the problem of authentication (asserting that some is who they pretend to be), password management and 2-factor authentication.
- Compiler Technology and High-Performance Computing: improved algorithms that make specific computing devices (small or big computers, computers in cars, calculators, or big computers in datacenters) faster and more energy-efficient at running programs.
- Software Engineering and Programming Languages: this area lies at the border between Human Sciences (programmers are people) and Computer Science, and considers methods and processes that make writing good programs, that work and do what the end-users want, easier and cheaper. This area encompasses the design of new programming languages, but also testing methodologies, managing the history of ever-changing pieces of software, delivering updates to the users, efficiently collecting and representing the users’ requirements, and analyzing programs to identify and locate problems.

Computer Science also considers the use of computers for running algorithms that produce knowledge in other domains, such as:

- Computational Biology applies computers to biology, for instance to simulate how medicines interact with molecules in our body, or to detect variants of viruses, establish phylogenetic trees between species, etc.
- Computer-Aided Design, for managing, designing and simulating various physical and engineering processes, such as heat transfer in structures, the load-bearing capacity of bridges, or how electricity may flow in a particular grid.

- Weather forecasts, something that almost everyone uses on a daily basis, and requires some of the most advanced and multidisciplinary knowledge in human history. Good weather forecasts mixes accurate simulation of large-scale physical processes, among with advanced data engineering and various forms of Artificial Intelligence (Machine Learning, Optimization and Constraint Solving).
- Graphics and Multimedia consider the use of computers for the production of human-consumed text, images, videos and audio. This field encompasses algorithms for drawing text, curves and 3D shapes, but also research on audio and video compression, critical in our current age of video streaming and video-conferencing, where every little bit of data is paid for in transfer time and energy expenditure.

Artificial Intelligence is a special area of Computer Science, in the sense that it relates and touches many of the other areas, and is not clearly categorized as "Computer Science for Computers" or "Computer Science for the World". We now focus on Artificial Intelligence and present its subfields, each accompanied with a brief review of the current scientific literature on the topic.

2 Subfields of Artificial Intelligence research

Later in this chapter, we discuss how to succinctly define Artificial Intelligence, aiming for a definition that considers what an AI system does, as opposed to the technical means it uses to do it. Before doing so, this section reviews the different research areas of Artificial Intelligence. AI researchers all over the world want to solve as many problems requiring AI as possible, while avoiding losing their time on problems that cannot or should not be solved with AI. As such, we consider that a review of scientific research in AI closely captures the set of real-world problems that require AI, and thus gives the reader an overview of what is "being artificially intelligent".

2.1 Search & Function Optimization

Search and Function Optimization are two different families of problems, but are solved by similar (or even identical) algorithms. We therefore present these two families together.

Search consists of finding a specific solution in a large problem space, with as little compute time as possible. The best-known example of a Search problem is the Sudoku game: given a grid of numbers (and missing numbers), and the set of rules of the game, the player has to find what numbers to put in every empty cell. Mazes are another example of search problems, where only one specific path leads to the exit.

Function Optimization can be seen as a variant of Search, in which not only *any* solution is required, but *the best one*. Function optimization considers a mathematical function, something that computes an output value given some input, and tries to produce the input that lead to the highest-possible output value.

Two kinds of functions are generally considered: black-box functions, for which the actual computations are not known to the optimization algorithm, and white-box functions, for which the computations are known. An example of a black-box function is an industrial processes (inputs: how machines are configured, output: how efficient the whole manufacturing plant is). An example of a white-box function is some quadratic function, $y=-8x^2+2x+3$, for which it is possible to analytically find the value of x that maximizes y.

Both Search and Function Optimization are highly general and abstract. In practice, the "problem space" and "input" can be anything that can be represented on a computer, such as graphs (road networks, machines on the Internet), descriptions of manufacturing machines, text, photos, etc. We provide more details below.

Real-world applications

We first note that "Search" does not have the same definition in AI research than in the general society. While most people consider "searching" looking for some piece of text, file or video, Search in AI means "finding a solution that meets constraints".

The Search application most people use on a daily basis is route planning[03], used primarily for GPS navigation. Given a map of localities connected by roads, a start point, and an end point, the algorithm has to find a path between the start and end points that is feasible (does not jump nor teleports) and maximizes some objective, such as speed or economy. A more general approach does not consider roads and localities, but the mathematical objects of edges and nodes in a graph. Many real-world objects can be mapped to edges and nodes in a graph, which allows the same algorithms to compute short routes for delivery trucks, but also efficient routing of data on the Internet[04], or manufacturing objects with as little machine movement as possible[05].

A difficult Search problem is the Boolean Satisfiability Problem (SAT). Given a large set of statements such as "A or B", "not A or B or C" or just "D", a true/false value has to be assigned to every variable (A to D in our example) such that *every* statement is true. Finding solutions to SAT problems, in reasonable time, is so difficult that there is an ongoing world-wide competition of SAT algorithms[3]. SAT is used to find how to update a Linux system[4], prove that two logic circuits perform the same operation (used in silicon manufacturing), or detect the impact of DNA mutations in the occurrence of diseases[06].

Finally, Search is often used in games to produce artificial opponents, such as in chess[07], make virtual opponents move in an intelligent way[08], or play the game itself[09]. This last point has more of an academic purpose than a real-world use.

Algorithms

The best-known algorithm for Search is A* [10] . If we introduce a concept of quality of the solution, Optimization algorithms are numerous and include Iterated Local Search[11], the nature-inspired Ant-Colony Optimization algorithm[12] - that considers virtual ants that depose virtual pheromones that evaporate over time, so that the shortest path automatically contains the most pheromones - genetic algorithms[13] and the cross-entropy method[14].

The above algorithms are compatible with black-box function, for which no knowledge is available. White-box functions are known to the algorithm, that is able to perform operations on it. With white-box functions, approaches such as Gradient Descent are available[15]. Gradient descent needs to compute the gradient (derivative) of the function, hence its need to know the equations of the function. Gradient Descent is presented in more details later in this chapter, as it is the core of how Neural Networks (part of Machine Learning) are trained.

3 http://www.satcompetition.org/
4 https://en.opensuse.org/Portal:Libzypp

2.2 Natural Language Processing

Natural language processing considers every operation performed by a computer on human-understandable or human-produced language, in all its forms. This broad research field includes, for instance, translation, voice recognition, sentiment analysis (from a piece of text, inferring the mood of the writer), text to speech (having a machine pronounce words), or text understanding (having a machine follow natural language instructions, for instance).

The two main aims of Natural Language Processing are:

- Proposing a user interface that is intuitive for as many people as possible. NLP algorithms are used here to produce natural language text or speech, and/or to understand natural language text or speech. Chatbots and automatic voicemails are examples of user-facing NLP systems.
  For this point, the user that interacts with the NLP system does it in real-time. The user is in front of the system.
- Extracting knowledge from human discourse produced some significant time in the past (not a user in front of a machine). Text summarization, voice recognition on archive data, and optical character recognition (turning photos of old documents to real text, searchable by a computer) are examples of NLP used to turn human-produced text or voice to digital data on which further processing can be done.

Some tasks, such as Machine Translation (translating between two natural languages) can be performed either in real-time or on historical data.

Real-world applications

Natural Language processing has been used to propose user-friendly human-machine interfaces, such as voice-based automatic answering machines[16] and chatbots that people interact with using natural language. Producing speech with a machine also helps accessibility, with tools such as Read Aloud[5] that transform text on webpages and documents to speech, to help blind people navigate the Internet.

Going in the other direction, understanding human speech leads to easy-to-use dictation devices[6] and automatic video subtitling, used, for instance, by the Czech Parliament to offer subtitled meetings[17]. That last application is particularly interesting when combined with Machine Translation, which allows someone to understand a foreigner in real time[18].

Reading human script is also often considered Natural Language Processing (some people consider that it is part of Computer Vision), and is used in note-taking applications[19] and to automatically read addresses on paper mail[20].

Finally, sentiment analysis allows to give an overall score or quality to a piece of text (or speech). It is used to flag customers that need special attention, complains that are worth looking at, mis-information on the Internet[21], detect suicidal people[22], detect possible child abuse[23], etc.

5 https://addons.mozilla.org/en-US/firefox/addon/read-aloud/
6 https://www.nuance.com/dragon.html

Algorithms

Understanding written human text requires methods such as classical parsing[24] (that looks at the words, and knows the grammar of the language, to analyze sentences) or Machine Learning approaches such as Sequence-to-Sequence neural networks[25]. Machine Learning is explored in more details later in this chapter.

Understanding speech requires signal processing such as Fourier Transforms, that map an audio signal to a description of the frequencies (high, medium, low pitches) in it[26]. Then, phoneme detection identifies sounds like vowels and consonants, before grouping them in words[27]. As is often the case in Natural Language Processing, where the tasks to be performed (understanding speech, understanding text, translation, etc) are complicated, but data abundant, Machine Learning approaches often compete or outperform hand-written algorithms based on expect linguistic knowledge. For instance, speech recognition can also be performed from sound directly to text by a neural network[28].

Optical Character Recognition can be perform by matching the shape of letters to images of characters, or by training a neural network to do the task[29]. Sentiment analysis can be performed in various ways, from the simplest (looking for specific key-words, or counting occurrences of some words) to Machine Learning approaches[30], some of them able to infer that sentences such as “This movie is as pleasant as an ice-cream by a cold winter day” is a negative review.

2.3 Knowledge Representation and Reasoning

One of the most valuable resources of mankind is knowledge. We collect vast amounts of it in different places (libraries, or digital archives), we build schools with the sole purpose of transferring this knowledge, and most of our current welfare is obtained thanks to applications of this knowledge. In order to transfer this knowledge, we typically use natural language. However, to transfer this knowledge to a computer, or some other artificially intelligent agent, for it to use this knowledge intelligently, natural language no longer suffices. Natural language is often ambiguous, and computers are (still) far from close to humans in resolving these ambiguities. For this reason *logics* are developed. They are formal languages (and hence, without ambiguities, and perfectly interpretable by a machine), often designed to not just be readable by machines, but also by humans. For instance, the sentence “All men are human” can be expressed as $\forall m : Man(m) \rightarrow Human(m)$ in typed first-order logic. With the knowledge that “ $\forall$ ” is read “*for all*”, and “→” is “implies”, we can recognize the original natural language sentence and interpret the formula as “for any object or idea ‘m’, if m is a man, then m is a human”.

Once knowledge has been represented in a formal way, it can be used in several to perform several tasks. For instance, deducing conclusions from a set of statements, proving whether something is true or false, or explaining something by producing the smallest amount possible of statements that will cause some recipient of these statements to deduce what we want it to deduce.

Real-world applications

Databases constitute the most-used application of logic in computer science. All communication with databases happens through logic. Somewhat more recently, the *semantic web* has surged in popularity. The idea underlying the semantic web is that any agent (human or artificial) on the internet can publish not just textual webpages, but also logical elements, such as information about themselves or others online[31]. Publishing this knowledge openly makes it reusable for other agents. Moreover, each agent can publish this knowledge using their own ontology, for instance while one might use the symbol “marriedTo” to represent the relation between two people, someone else might use “spouse”. How these different ontologies relate can then again be represented in a suitable logic. Other applications of logic and knowledge include: *certifying correctness of hardware design* (see the description of Boolean satisfiability above), *configuration tools*, where a complex system needs to be configured, either automatically, or in interaction with a human, *verifying correctness of other systems*, or even *explaining decisions made by other AI methods*.

2.4 Machine Learning

Machine Learning is the last big subfield of Artificial Intelligence that we consider in this chapter, on purpose, to emphasize that despite the hype for Machine Learning, Artificial Intelligence is much broader than that. Regulators must therefore pay extra attention to whether they want to regulate Machine Learning (and its fundamental aspect, the fact that a machine learns) or something bigger in AI (for its close interaction with people for instance).

Machine Learning considers problems in which an algorithm has to extract knowledge from inputs presented to it. The extracted knowledge is what the algorithm “learns”, and is a summary of the inputs that have been presented to the algorithm.

Three families of Machine Learning algorithms exist:

- Supervised Learning, the best known by the general public. The algorithm is presented a large set of input-output pairs, called the training set (sometimes also the dataset, but training set is more precise). An example of such set is a large list of photos along with the name of the person in the photo. The goal of the algorithm is to learn a mapping from input to output, that correctly maps the inputs seen during training to their respective outputs, and also, hopefully, correctly maps new inputs (not seen during training) to the outputs they should have.
  The knowledge extracted by the algorithm is the mapping, some computer data that can have many different forms depending on the exact algorithm, such as the description of a decision tree, or the weights of a neural network. The important aspect is that the mapping is somewhat tangible, as it can be represented as a computer file, and is most of the times orders of magnitude smaller than the training set used to produce it.

- Unsupervised Learning has no output. The algorithm is presented a large amount of inputs, and learns the structure of it. The knowledge it extracts can then be used for a computer to answer questions such as “have I already seen something similar?” or “is this significantly different from what I’ve seen when training?”.

- Reinforcement Learning sits somewhat in-between. It considers a setting in which the learning algorithm (here called the “agent”) has to learn how to perform a sequential task, such as assembling something, guiding a user on a website, managing the heating and cooling of a building, bidding on an energy market, etc. The agent is provided inputs (called “states”), such as images acquired through a camera, and executes actions, such as moving a robot arm or turning on a machine in an industrial plant. After every action, the agent is provided with a reward, a single number that can be anything, positive or negative. The objective of the agent is, by trying actions and observing rewards, to progressively learn

what action leads to the best-possible sum of future rewards in any particular state. Reinforcement Learning is used when the designer of a system knows what has to be done (and hence can produce rewards) but not how to do it (in which case Supervised Learning would have been more efficient).
Reinforcement Learning is not Supervised Learning because the agent is not told what to do in every state, it only observes rewards and has to guess, try and learn what to do.
Reinforcement Learning is not Unsupervised either, because the rewards are there, the agent is fully alone doing its task.

Real-world applications

Machine Learning can be applied to so many real-world domains that it sometimes feels like anything can be done with Machine Learning. Because listing these applications is not possible in a book chapter, we instead choose to only succinctly present some applications of Machine Learning, and dedicate more pages later to explaining how these algorithms work and how they can potentially impact society.

Supervised Learning is used whenever some annotated data is available, and an automated way of generalizing that annotated data is desirable. Supervised learning can be used in biology to count cells, detect cancerous cells and predict the structure or function of proteins[32]. It also excels in Computer Vision, where it can be used to recognize handwriting[33], producing a textual description of the contents of images[34], detecting corrosion on public infrastructure[35], or analyzing satellite images for signs of human activity and needs[36]. Supervised Learning also achieved great success in Machine Translation[37], a task on which human knowledge (knowing vocabulary lists, grammar rules, etc) was thought to be necessary.

Unsupervised Learning is mainly used for fraud detection[38], and generative modeling[39], a task in which “things” are shown to the algorithm, that learns to produce new (original) “things” that look like the algorithm saw during training, such as new images in the pointillist style. Obstacle detection can also be performed with Unsupervised Learning[40]: if the algorithm has seen obstacle-free routes during training, and it does not recognize what it sees, then there is most probably an obstacle somewhere.

Reinforcement Learning first attracted wide attention in the AI field when it allowed a computer program to beat the human champion at Go[41], a board game previously thought to be 50 years ahead of what AI can do. Outside of game playing, Reinforcement Learning is used whenever we know what has to be done, but not how to do it, such as optimizing the cooling of large datacenters[42], designing efficient prevention strategies for infection diseases[43].

Biases in Data and in Algorithms

Machine Learning considers mathematical approaches that extract knowledge from the data presented to them, using formulas and algorithms. No part of them has a consciousness or knows the rules of society. This means that Machine Learning algorithms are inherently unbiased, but are also completely unable to correct the bias in the data they receive, to detect that they are tricked, or that their output is improper or impolite.

Several events have sparked interest towards biases in Machine Learning, and how to avoid them. Microsoft Corporation once design at chatbot that could learn from the interactions it had with people, and that was quickly turned to a racist and nazi propaganda machine by Twitter users[7]. Microsoft disabled the chatbot shortly after deploying it. Google developed an image recognition engine that worked very well, but never saw dark-skinned people in its training set (probably because nobody designing it thought of that). It was however presented with numerous images of gorillas, annotated as “gorilla”, and over-generalized in a way that some Black people were classified as gorillas[8]. The system was taken offline and Google apologized profusely.

There are two takeaways in the examples listed above: first, even big tech companies had biases in their datasets, and damaged their public image due to it. Second, the problem always comes from the dataset. A very simple example of a Machine Learning algorithm could be the simple computation of an average, for which an exact mathematical formula exists. There is no bias in the formula. But if only the height of tall people are given to the average function, then it will predict that the average height of humans is much larger than what it actually is.

Correcting biases in Machine Learning systems is therefore a human problem, not an algorithmic one. People gathering data for Machine Learning should be well aware of their own possible biases, and on how to collect balanced datasets[44].

Explainable Machine Learning

Explainable Machine Learning, also sometimes referred to as Explainable AI (Machine Learning is sometimes confused with the wider field of AI), focuses on the interaction between an algorithm, and the people who use it. Contrary to biases discussed above, that consider the data presented to the algorithm, Explainability is purely an algorithmic issue. Most modern Machine Learning methods, such as neural networks, are perfectly grounded, use exact algorithms for which every formula is known, but the formulas are so far from how the human brain works that it is impossible, given a prediction or decision output by the algorithm, to know *why* it produced that prediction.

An illustration of that problem is a neural-network based method for predicting whether someone could pay back a loan. The description of a person comes in (age, salary, household size, etc), and a yes or no answer is produced by the algorithm. Neural networks are built on matrix multiplications and simple operations such as sinuses. However, nobody, given the description of a person and the numerical value of every cell of a dozen of matrices, could explain why the loan was rejected.

Explainable Machine Learning considers technical approaches that allow to link inputs to outputs of algorithms in a way that makes more sense for humans. It is closely related to psychology and human-computer interface design. The field is still in its early days, but some results are already available, such as knowledge that people well-versed in a domain (banking, medicine, etc) are usually able to make sense of decision rules, decision trees, and simple regressions[45]. Several approaches therefore exist that first train an algorithm using an easily-trainable but not-explainable method, such as neural networks, then *distills* the resulting knowledge to a set of rules or a decision tree, much harder to train but somewhat explainable.

Algorithms

Supervised Learning methods include decision trees, learned with algorithms such as ID3 and C4.5[46], random forests[47], support vector machines[48], Gaussian mixture models[49] and neural

7 https://spectrum.ieee.org/in-2016-microsofts-racist-chatbot-revealed-the-dangers-of-online-conversation
8 https://www.wsj.com/articles/BL-DGB-42522

networks. Because, for most people, AI is synonym with Machine Learning, that in turn is synonym with Neural Networks, we take the opportunity to briefly present them and explain how they work in the next section.

Unsupervised Learning methods include k-Nearest Neighbors[50], algorithms that aim at quickly finding which data points look like some other data point, Gaussian mixture models referenced above, and special kinds of neural networks used for generative modeling. Generative modeling consists of observing a dataset, learning its structure and inherent properties, and be able to generate new data points that look like what was in the dataset. The first main generative modeling approaches is Generative Adversarial Neural networks[51], in which two neural networks, the generator and the discriminator, are trained together. The generator is rewarded for producing outputs that the discriminator is unable to differentiate from the original dataset. The discriminator is rewarded for being able to tell apart generated and original data points. The second Generative Modelling approach is Variational Autoencoders[52], in which a single neural network learns to map an original datapoint to something that looks as close as possible to it, with some randomness in the network that allows to "vary" the output, and produce new data points that look like the ones in the dataset.

Reinforcement Learning methods are divided in two families, depending on whether they learn how good doing something in a particular situation is (value-based methods), or how often they should perform an action in a particular situation (policy-based methods). Both approaches have strengths and weaknesses on different domains, and none dominates the other. Value-based methods include DQN[53], known to play Atari games at human level, its modern variants such as Rainbow[54], and the MuZero algorithm[55] that combines value-based Reinforcement Learning with a Search algorithm (Monte-Carlo Tree Search) to achieve super-human level on the game of Go. Policy-based methods usually perform better than value-based methods on robotic tasks, with the two dominant approaches being PPO[56] and, the current state-of-the-art, the Soft Actor-Critic[57].

Neural Networks

The term "neural network" originates from the work of Rosenblatt et al. in 1958 on the Perceptron, a proposed computer model of how the brain stores data[58]. Since that visionary paper, the same architecture (simple components connected in layers, like the neurons in the brain) has been used for progressively more advanced methods, such as the multi-layer perceptron[59], backpropagation[60], convolutional neural networks[61] and recurrent neural networks[62]. These method continued to be called "neural networks", even though they are now very far from how neurons and the brain work.

In the modern literature, a neural network is any kind of differentiable parametric function. A function "f" computes an output "y" given some input "x", which is usually denoted "$y = f(x)$". The computation done by f can be anything, from a simple addition to a large sequence of mathematical operations. A parametric function adds another input, θ (theta), the parameter, to form "$y = f(x, \theta)$".

Using a parametric function for supervised learning is straightforward: given examples of inputs "x" and outputs "y", the learning algorithm has to find θ such that the outputs predicted by "f" are as close as possible to the true outputs presented to the algorithms. Gradient descent[63] does this by assuming that "f" is differentiable, hence has a gradient, a mathematical concept that answers the question "how should I change θ for the output of $f(x, \theta)$ to be a bit closer to the true output y". Explaining how it is done is outside the scope of this chapter, but the Gradient Descent review referenced above can easily be followed by an interested reader with (good) high-school mathematical literacy.

The most important take-away of this section is that a neural network is therefore not magical. It is a piece of computer code that computes an output given some input and a parameter θ (usually by performing a number of matrix multiplications and some additions), and the Gradient Descent algorithm, that allows to find θ such that the outputs produced by the function are close to what can be found in some dataset. Everything in these algorithms has exact implementations, readable by any programmer, and devoid of incantations, dark magic or randomness.

3 Defining Artificial Intelligence

The first part of this chapter focused on exploring and detailing several areas of Artificial Intelligence system. By listing what is, at the time of writing this chapter, done on a regular basis by Artificial Intelligence researchers, we aim at helping the reader grasping the scope of current AI research, but also how big it is, how varied it is, and how difficult to succinctly define it is.

Artificially-Intelligent Nature or Behavior

Artificial Intelligence may refer to the research field of AI, intelligent systems (AI systems), and Artificial Intelligence tasks such as text translation, speech recognition, or image captioning:

1. The research field of Computer Science that focuses on defining Intelligence, studying it in natural systems, and replicating it in computer programs and algorithms. Artificial Intelligence research also encompasses mathematics concepts such as proofs, formal logic and numerical analysis. Artificial Intelligence research is usually self-defined: researchers claim that they do, or do not do, Artificial Intelligence research.

2. Artificial Intelligence systems, such as robots, computer software, or web platforms, that have a particular hardware or software realization (e.g. it uses Machine Learning and this particular Planning algorithm). This definition focuses on *how* a system performs a task, its *technical means*. It is possible to use Machine Learning to propose holiday trips to people on a website (which would fall under this definition of an Artificial Intelligence system), but it is also possible to propose holiday trips at random, which would then not be considered being Artificially Intelligent by most people.

3. An Artificial Intelligence task or problem. This is a specific outcome that has to be reached, such as “finding the shortest path from A to B”. This definition focuses on *what* task has to be performed, the *overall objective* (as opposed to the technical means above). As in mathematics, the set of Artificial Intelligence tasks is assumed to be timeless and perpetual, part of the Universe. We, humans, discover the existence of these problems as society evolves.

Schuett et. al study in more detail the difference between defining Artificial Intelligence according to what a system does, or how it does it[64].

Interestingly, while non-AI experts focus on the Point 2 above approach to recognizing and defining AI (focusing on the tools and means used to achieve an objective), AI researchers follow Point 3, and systematically wonder whether a particular task is Artificial Intelligence by nature, or could be solved without Artificial Intelligence.

In the remaining of this chapter, we argue that we should focus on Point 3: is a particular task an Artificial Intelligence task? A system is then an AI system if it solves an AI task, regardless of the means used. We believe that a definition built on *what it does* is longer-term, and more robust to future technological developments.

We also hope that defining AI according to what a system does, not how it does it, will avoid some of the early criticism that was received by the AI Act[9] early in its inception[10]: an AI system is regulated or forbidden based on what it does, its particular use or deployment in the real world. A malevolent entity thus cannot pretend not to be using AI. It is also clearer that the use of the tool is forbidden, not the making of the tool itself.

Focusing on the what, the task performed by a system, instead of how it does it, also nicely reflects the general rule of Law that someone should be judged on their actions and not their nature.

3.2 Definitions used in existing literature

Many works have tried to define Artificial Intelligence in the past decades. No formal and widely-accepted definition emerged mainly due to different authors relating “Intelligence” to different concepts:

1. Comparable to humans: an Artificial Intelligence is anything that is able to perform tasks that humans are good at. This definition is used in 1967 by Dreyfus to temper the original optimism about AI, at first used for tasks difficult for humans but easy for machines[65], and by Dobrev to build a formal model of a world in which a machine performs comparably to humans[66].
2. Interacting with humans: an Artificial Intelligence performs a task that interacts with humans, such as understanding speech or images, making decisions that impact humans, or help humans in their daily life. This definition does not appear in formal literature, but is often used in classes on AI, or in the early versions of the AI Act.
3. Independent from humans: definitions of Intelligence that do not compare it against humans. For instance, the AI Act refers to a system’s adaptiveness or its ability to infer outputs from inputs it may receive[11] (Article 3). Lehman-Wilzig mentions a definition originating from cybernetics, with most of the definition (being able to find and fulfill purposes, for instance) being independent from humans, and only one small point being human-centric[67] (being as creative as humans). Nielsson builds his definition of AI as being able to act with foresight[68].

Finally, a broad overview of definitions of AI, along with a discussion of what is a definition, is exposed by Wang[69].

3.3 Definitions used in the Artificial Intelligence Act

The Artificial Intelligence Act refers to AI systems and tasks in various places, and tries to define AI in a succinct yet general enough way. A complete review of the Act is outside the scope of this chapter, but we provide some snippets of the Act with comments on the definitions they contain.

9 EUR-Lex document 52021PC0206, see footnotes below for more user-friendly links and references
10 https://www.brookings.edu/blog/techtank/2021/05/04/machines-learn-that-brussels-writes-the-rules-the-eus-new-ai-regulation/
11 https://www.europarl.europa.eu/meetdocs/2014_2019/plmrep/COMMITTEES/CJ40/AG/2024/02-13/1296003EN.pdf

In this section, we refer to Articles and Annexes from a pre-approval version of the AI Act[12] of April 2024.

Article 3 contains a very brief definition of AI:

'AI system' is a machine-based system designed to operate with varying levels of autonomy and that may exhibit adaptiveness after deployment and that, for explicit or implicit objectives, infers, from the input it receives, how to generate outputs such as predictions, content, recommendations, or decisions that can influence physical or virtual environments;

This definition focuses on what the system does, with a mention that it is machine-based. This raises the question of what is a machine, and *human-made* would have been clearer even though not perfect yet (is an AI system designed by another AI system an AI system?). We note that a succinct list of AI tasks, such as prediction and recommendation, are listed. These tasks appear to focus mostly on Machine Learning, and tasks such as logic reasoning, planning, function optimization, or translation, are not mentioned. The "may" in the definition ensures that it is not a closed list, and allows for any machine-based system to be considered an AI system. This links to Article 5 that we discuss now:

(a) the placing on the market, putting into service or use of an AI system that deploys subliminal techniques beyond a person's consciousness or purposefully manipulative or deceptive techniques, with the objective to or the effect of materially distorting a person's or a group of persons' behaviour by appreciably impairing the person's ability to make an informed decision, thereby causing the person to take a decision that that person would not have otherwise taken in a manner that causes or is likely to cause that person, another person or group of persons significant harm;

Article 5 provides a description of actions a system cannot perform, which completely focuses on the *what* a system does. Combined with Article 3, that has a very open definition of an AI system, it seems that the regulators tried to rephrase the AI Act to actually not care about whether a system is an AI system or not.

Earlier versions of the draft, consulted by the author in 2020 and 2021, had a definition of "AI System" that was less open and explicitly listed AI methods (focusing on the how). Criticism about allowing forbidden tasks if they are done by a non-AI system seems to have been addressed by making it impossible to say that a system is not AI.

The analysis of the definitions of AI in the AI Act already stop here, because the Act does not try to provide further definitions of AI. The rest of the Act focuses on forbidden or regulated activities (what, not how), mentioning the term "AI" often, but always ensuring that the "AI" in "AI System" is wide enough that any machine, algorithm or device falls into the definition. The authors of this chapter welcome this phrasing in the AI Act and draw the parallel with how the legal system aims for, and should, judge persons according to their actions and not their nature.

3.4 Tentative definition focusing on the What

While Section 2 provides a detailed list of fields in Artificial Intelligence research, which gives an intuition of what AI is through examples, we propose in this section a short discussion on how to concisely define AI, in a way that is broad enough to cover current and futures applications of AI,

12 https://www.europarl.europa.eu/meetdocs/2014_2019/plmrep/COMMITTEES/CJ40/AG/2024/02-13/1296003EN.pdf

yet simple enough to be understandable by the general public. We don't aim at providing a legal definition of AI, and strive to focus on what a system does, not how it does it.

Artificial System

For most people, the Artificial part of Artificial Intelligence is important, and a person is not an AI system. It is however an open question of whether some Artificial Intelligence forbidden activities (such as tricking a person or causing harm to them) remain forbidden if they are performed by something that is not an AI system, such as a person or an animal. In case it is important to distinguish a person from a "system" in "AI system", we propose the following definition of *system*:

A human-made artifact that is not a legal person, such as, but not limited to, a robot, a website, a computer program or a physical object.

This definition aims at considering everything possible, except for natural elements (animals, plants, the Earth) and people. We consider that a trained dog is not an AI System. We also consider that, if, in the future, some advanced computer programs or robots act such that that society recognizes them as people and makes them legal persons, these "persons" are not systems anymore and should be judged on their own.

The definition above provides a brief list of systems, that includes websites and physical objects. We believe that it is important to note that AI may be deployed on many physical devices, not only computers or robots. A website is usually not considered a robot, but, because it presents information to a person and is dynamic (it can react to clicks), a website may still have significant psychological impact on the human being visiting it and is not just a nice-looking sheet of paper.

Finally, *human-made* requires a definition too. Proposing a detailed description is outside the scope of this chapter, as we have no expertise in such an important definition, but we point out that anything that comes to existence thanks to the intervention of a human or another human-made artifact should be considered human-made too. A device produced by a 3D printer is human-made, a robot assembled by another robot is human-made, and a component designed though Generative Design[13] is human-made too.

Artificial Intelligence System

We aim at defining an Artificial Intelligence System as:

A system that performs an Artificial Intelligence task

We therefore purely focus on *what* the system does and do not mention any possible technical mean of achieving the task. This simple definition builds on the definition of what an AI task is.

Artificial Intelligence Task

Defining what is an Artificial Intelligence Task is the most difficult aspect of the definition. We observe in the literature several directions being followed to define an AI task:

- Approaches that focus on *intelligence*: an AI task is a task that requires intelligence. This then allows to move to philosophy and thousands of years of thinking about what

13 An example of such technology used in production is Autodesk's Generative Design: https://www.autodesk.com/solutions/generative-design

intelligence is. In his book The Quest for Artificial Intelligence[70], Niels J. Nielsson defines intelligence as: “intelligence is that quality that enables an entity to function appropriately and with foresight in its environment”.

- Approaches that focus on the effect of the task: an AI task is a task that requires interacting directly or indirectly with people, or computing something based on data generated by people, or producing data or decisions that will be used by people or will influence people. The aim of this approach is to avoid falling into “everything is AI”, because, following the definition of intelligence by Nielsson, any useful computer program or algorithm is intelligent (and therefore AI). This approach tries to make the distinction between algorithms that interact with people (websites, loan decisions, chatbots) and algorithms that don’t (motion planning in a factory, packet routing on the Internet).
- Approaches based on lists: the approach followed by the AI Act, that sometimes refers to an AI task as any task *listed in some Annex*. This approach is the most precise, but also the most anchored in current times. Creative people come up with new tasks to artificially solve every day, and identifying which ones are AI, to update a legal document, would require significant work.

If the aim of a definition of AI is to select what tools or products fall under a regulation, we propose to adopt the widest-possible definition, to avoid missing a tool or product. We therefore suggest to follow the *intelligence* approach, even though it encompasses most if not all computer programs and algorithms. Over-simplifying things a bit, we therefore propose that everything artificial and that performs something (so, every robot even the simplest one, every dynamic website, every computer program) is an Artificial Intelligence System.

4 Conclusion

In this chapter, we aimed at providing a definition of Artificial Intelligence. The first section started with an overview of Computer Science, and gave an intuition of what an algorithm is, and how tangible it is. The second section explored the area of Artificial Intelligence research, and described a selection of tasks tackled by AI. Finally, the third section took a step back and discussed how Artificial Intelligence can be defined based on what an AI system does, to finally propose a succinct definition.

[01]McCarthy, John; Minsky, Marvin; Rochester, Nathan; Shannon, Claude (1955). "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence".
[02]Merriam-Webster. (n.d.). Algorithm. In Merriam-Webster.com dictionary. Retrieved May 14, 2024, from https://www.merriam-webster.com/dictionary/algorithm
[03]Dantzig, G. B. (1960). On the shortest route through a network. Management Science, 6(2), 187-190.
[04]Cantor, D. G., & Gerla, M. (1974). Optimal routing in a packet-switched computer network. IEEE Transactions on computers, 23(10), 1062-1069.
[05]Chentsov, P. A., & Petunin, A. A. (2016). Tool routing problem for CNC plate cutting machines. IFAC-PapersOnLine, 49(12), 645-650.
[06]Marques-Silva, J. (2008). Practical applications of boolean satisfiability.
[07]Haworth, G., & Hernandez, N. (2021). The 20th Top Chess Engine Championship, TCEC20. J. Int. Comput. Games Assoc., 43(1), 62-73.
[08]Cui, X., & Shi, H. (2011). A*-based pathfinding in modern computer games. International Journal of Computer Science and Network Security, 11(1), 125-130.
[09]Chaslot, G., Bakkes, S., Szita, I., & Spronck, P. (2008). Monte-carlo tree search: A new framework for game ai. In Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (Vol. 4, No. 1, pp. 216-217).
[10]Hart, P. E.; Nilsson, N. J.; Raphael, B. (1968). "A Formal Basis for the Heuristic Determination of Minimum Cost Paths". IEEE Transactions on Systems Science and Cybernetics. 4 (2): 100–107.
[11]Lourenço, H. R., Martin, O. C., & Stützle, T. (2003). Iterated local search. In Handbook of metaheuristics (pp. 320-353). Springer, Boston, MA.
[12]Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. IEEE computational intelligence magazine, 1(4), 28-39.
[13]Sivanandam, S. N., & Deepa, S. N. (2008). Genetic algorithms. In Introduction to genetic algorithms (pp. 15-37). Springer, Berlin, Heidelberg.
[14]De Boer, P. T., Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A tutorial on the cross-entropy method. Annals of operations research, 134(1), 19-67.
[15]Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747.
[16]McCarthy, I. O., Wojno, A. E., Joseph, H. A., & Teesdale, S. (2017). Check and Report Ebola (CARE) Hotline: the user perspective of an innovative tool for postarrival monitoring of Ebola in the United States. JMIR Public Health and Surveillance, 3(4), e7817.
[17]Pražák, A., Psutka, J. V., Hoidekr, J., Kanis, J., Müller, L., & Psutka, J. (2006, September). Automatic online subtitling of the Czech parliament meetings. In International Conference on Text, Speech and Dialogue (pp. 501-508). Springer, Berlin, Heidelberg.
[18]Zhang, G., & Huang, J. (2016). U.S. Patent No. 9,477,657. Washington, DC: U.S. Patent and Trademark Office.
[19]Grahame, J. A. (2016). Digital note-taking: Discussion of evidence and best practices. The Journal of Physician Assistant Education, 27(1), 47-50.
[20]Rosenbaum, W. S., Carris, B. T., & Ankerstjerne, A. (1993). U.S. Patent No. 5,249,687. Washington, DC: U.S. Patent and Trademark Office.
[21]Bhutani, B., Rastogi, N., Sehgal, P., & Purwar, A. (2019, August). Fake news detection using sentiment analysis. In 2019 twelfth international conference on contemporary computing (IC3) (pp. 1-5). IEEE.
[22]Christensen, H., Batterham, P. J., & O'Dea, B. (2014). E-health interventions for suicide prevention. International journal of environmental research and public health, 11(8), 8193-8212.
[23]Lyu, Y., Chow, J. C. C., & Hwang, J. J. (2020). Exploring public attitudes of child abuse in mainland China: A sentiment analysis of China's social media Weibo. Children and Youth Services Review, 116, 105250.
[24]Collins, M. (2003). Head-driven statistical models for natural language parsing. Computational linguistics, 29(4), 589-637.
[25]Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461.
[26]Liu, W., Liao, Q., Qiao, F., Xia, W., Wang, C., & Lombardi, F. (2019). Approximate designs for fast Fourier transform (FFT) with application to speech recognition. IEEE Transactions on Circuits and Systems I: Regular Papers, 66(12), 4727-4739.
[27]Ostendorf, M., & Roukos, S. (1989). A stochastic segment model for phoneme-based continuous speech recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, 37(12), 1857-1869.
[28]Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K., & Bengio, Y. (2015). Attention-based models for speech recognition. Advances in neural information processing systems, 28.
[29]Charles, P. K., Harish, V., Swathi, M., & Deepthi, C. H. (2012). A review on the various techniques used for optical character recognition. International Journal of Engineering Research and Applications, 2(1), 659-662.
[30]Yadollahi, A., Shahraki, A. G., & Zaiane, O. R. (2017). Current state of text sentiment analysis from opinion to emotion mining. ACM Computing Surveys (CSUR), 50(2), 1-33.
[31]Hitzler, P. (2021). A review of the semantic web field. *Communications of the ACM*, *64*(2), 76-83.

[32]Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., ... & Robles, V. (2006). Machine learning in bioinformatics. Briefings in bioinformatics, 7(1), 86-112.
[33]Dutta, K., Krishnan, P., Mathew, M., & Jawahar, C. V. (2018, August). Improving CNN-RNN hybrid networks for handwriting recognition. In 2018 16th international conference on frontiers in handwriting recognition (ICFHR) (pp. 80-85). IEEE.
[34]Hossain, M. Z., Sohel, F., Shiratuddin, M. F., & Laga, H. (2019). A comprehensive survey of deep learning for image captioning. ACM Computing Surveys (CsUR), 51(6), 1-36.
[35]Atha, D. J., & Jahanshahi, M. R. (2018). Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection. Structural Health Monitoring, 17(5), 1110-1128.
[36]YEH, Christopher, PEREZ, Anthony, DRISCOLL, Anne, et al. Using publicly available satellite imagery and deep learning to understand economic well-being in Africa. Nature communications, 2020, vol. 11, no 1, p. 1-11.
[37]Popel, M., Tomkova, M., Tomek, J., Kaiser, Ł., Uszkoreit, J., Bojar, O., & Žabokrtský, Z. (2020). Transforming machine translation: a deep learning system reaches news translation quality comparable to human professionals. Nature communications, 11(1), 1-15.
[38]Hilas, C. S., & Mastorocostas, P. A. (2008). An application of supervised and unsupervised learning approaches to telecommunications fraud detection. Knowledge-Based Systems, 21(7), 721-726.
[39]Salakhutdinov, R. (2015). Learning deep generative models. Annual Review of Statistics and Its Application, 2, 361-385.
[40]Wellington, C., Courville, A., & Stentz, A. (2006). A generative model of terrain for autonomous navigation in vegetation. The International Journal of Robotics Research, 25(12), 1287-1304.
[41]Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Hassabis, D. (2017). Mastering the game of go without human knowledge. nature, 550(7676), 354-359.
[42]Li, Y., Wen, Y., Tao, D., & Guan, K. (2019). Transforming cooling optimization for green data center via deep reinforcement learning. IEEE transactions on cybernetics, 50(5), 2002-2013.
[43]Libin, P. J., Moonens, A., Verstraeten, T., Perez-Sanjines, F., Hens, N., Lemey, P., & Nowé, A. (2020, September). Deep reinforcement learning for large-scale epidemic control. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 155-170). Springer, Cham.
[44]Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. ACM Computing Surveys (CSUR), 54(6), 1-35.
[45]Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., ... & Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Information fusion, 58, 82-115.
[46]Singh, S., & Gupta, P. (2014). Comparative study ID3, cart and C4.5 decision tree algorithm: a survey. International Journal of Advanced Information Science and Technology (IJAIST), 27(27), 97-103.
[47]Biau, G., & Scornet, E. (2016). A random forest guided tour. Test, 25(2), 197-227.
[48]Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines. IEEE Intelligent Systems and their applications, 13(4), 18-28.
[49]Rasmussen, C. (1999). The infinite Gaussian mixture model. Advances in neural information processing systems, 12.
[50]Friedman, J. H., Baskett, F., & Shustek, L. J. (1975). An algorithm for finding nearest neighbors. IEEE Transactions on computers, 100(10), 1000-1006.
[51]Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2020). Generative adversarial networks. Communications of the ACM, 63(11), 139-144.
[52]Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.
[53]Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Hassabis, D. (2015). Human-level control through deep reinforcement learning. nature, 518(7540), 529-533.
[54]Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., ... & Silver, D. (2018, April). Rainbow: Combining improvements in deep reinforcement learning. In Thirty-second AAAI conference on artificial intelligence.
[55]Schrittwieser, J., Hubert, T., Mandhane, A., Barekatain, M., Antonoglou, I., & Silver, D. (2021). Online and offline reinforcement learning by planning with a learned model. Advances in Neural Information Processing Systems, 34, 27580-27591.
[56]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.
[57]Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., ... & Levine, S. (2018). Soft actor-critic algorithms and applications. arXiv preprint arXiv:1812.05905.
[58]Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review, 65(6), 386.
[59]Ruck, D. W., Rogers, S. K., & Kabrisky, M. (1990). Feature selection using a multilayer perceptron. Journal of Neural Network Computing, 2(2), 40-48.
[60]Almeida, L. B. (1989). Backpropagation in perceptrons with feedback. In Neural computers (pp. 199-208). Springer, Berlin, Heidelberg.

[61]LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 3361(10), 1995.
[62]Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.
[63]Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747.
[64]Schuett, J. (2019). A legal definition of AI. *arXiv preprint arXiv:1909.01095*.
[65]Dreyfus, H. L. (1967). Why computers must have bodies in order to be intelligent. The Review of Metaphysics, 13-32.
[66]Dobrev, D. (2005). Formal definition of artificial intelligence. International Journal of Information Theories & Applications, 12(3), 277-285.
[67]Lehman-Wilzig, S. N. (1981). Frankenstein unbound: Towards a legal definition of artificial intelligence. Futures, 13(6), 442-457.
[68]Nilsson, N. J. (2009). The quest for artificial intelligence. Cambridge University Press.
[69]Wang, P. (2019). On defining artificial intelligence. Journal of Artificial General Intelligence, 10(2), 1-37.
[70]Nilsson, N. J. (2009). The quest for artificial intelligence. Cambridge University Press.