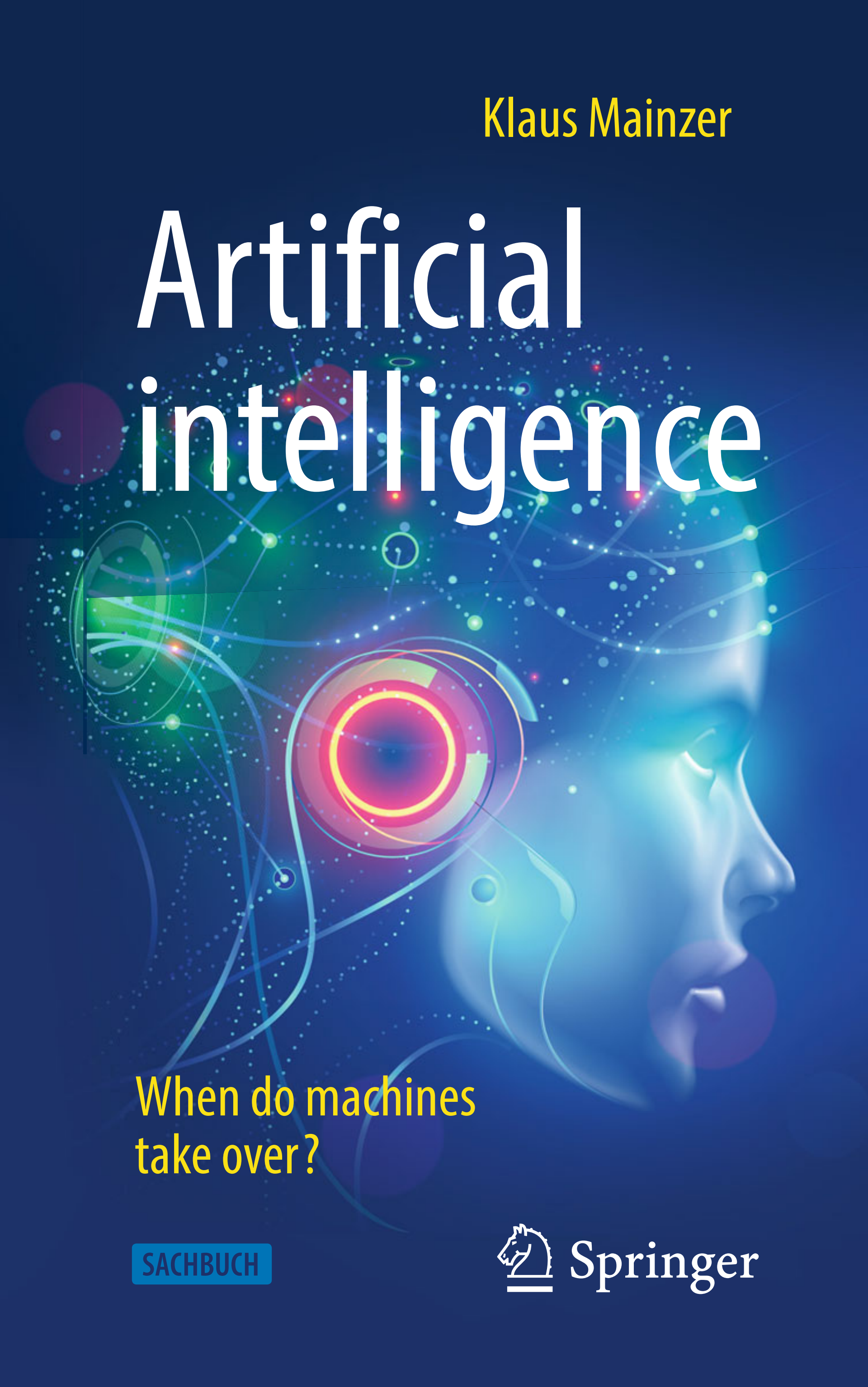
Klaus Mainzer

Artificial intelligence

When do machines take over?



SACHBUCH

Springer

Technik im Fokus

Die Buchreihe Technik im Fokus bringt kompakte, gut verständliche Einführungen in ein aktuelles Technik-Thema.

Jedes Buch konzentriert sich auf die wesentlichen Grundlagen, die Anwendungen der Technologien anhand ausgewählter Beispiele und die absehbaren Trends.

Es bietet klare Übersichten, Daten und Fakten sowie gezielte Literaturhinweise für die weitergehende Lektüre.

More information about this series at http://www.springer.com/series/8887

Klaus Mainzer

Artificial intelligence – When do machines take over?

Springer

Klaus Mainzer
Emeritus of Excellence
Technical University of Munich
Munich, Germany

Carl Friedrich von Weizsäcker
Center, Eberhard Karls University
of Tübingen, Tübingen, Germany

ISSN 2194-0770 ISSN 2194-0789 (electronic)
Technik im Fokus
ISBN 978-3-662-59716-3 ISBN 978-3-662-59717-0 (eBook)
https://doi.org/10.1007/978-3-662-59717-0

Translation from the German Language edition: Mainzer: Künstliche Intelligenz – Wann übernehmen die Maschinen?, © Springer-Verlag GmbH Deutschland 2018

© Springer-Verlag GmbH Germany, part of Springer Nature 2020
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer-Verlag GmbH, DE part of Springer Nature.
The registered company address is: Heidelberger Platz 3, 14197 Berlin, Germany

Foreword

Artificial intelligence has long dominated our lives without many people being aware of it. Smartphones that talk to us, wristwatches that record our health data, workflows that organize themselves automatically, cars, airplanes and drones that control themselves, traffic and energy systems with autonomous logistics or robots that explore distant planets are technical examples of a networked world of intelligent systems. They show us how our everyday life is determined by AI functions.

Biological organisms are also examples of intelligent systems which, like humans, have evolved during evolution and can solve problems more or less independently and efficiently. Occasionally nature is a model for technical developments (for example, neural networks as simplified models of the human brain). However, sometimes, computer science and engineering find solutions that are different, even better and more efficient than in nature. Therefore, there is not "the" artificial intelligence, but degrees of efficient and automated problem solving in different domains.

Behind this is the world of machine learning with learning algorithms, which become more and more powerful with exponentially growing computing capacity. Scientific research and medicine are already using neural networks and learning algorithms to discover correlations and patterns in a growing flood of measurement data. Machine learning algorithms are already applied in business strategies and the industrial Internet. They control the processes of a networked world in the Internet of

Things. Without them, the flood of data generated by billions of sensors and networked devices would not be manageable.

But, the state of the art in machine learning is based on statistical learning and reasoning with an exploding number of parameters. In general, statistical correlations cannot be replaced by causal explanations. They deliver risky black boxes trained by Big Data. Therefore, statistical learning must be overcome by causal learning. Causal learning does not only enable better explanation of causes and effects, but also better accountability to decide legal and ethical questions of responsibility (e.g. in autonomous driving or in medicine). Obviously, in addition to the innovation of artificial intelligence, the challenges of security and responsibility come to the fore. This book is a plea for certification and verification of AI-programs. We analyze empirical test procedures as well as automated formal proofs. In the end, the demand for certification is no killer of innovation, but the chance for better and sustainable AI-programs.

Since its inception, AI research has been associated with great visions of the future of mankind. Is “artificial intelligence” replacing humans? Some already speak of a coming “super intelligence” that triggers fears and hopes. This book is also a plea for technology design: AI must prove itself as a service in society. As service system, AI-technology with its immense need of energy must not be at the expense of ecology. Therefore, we should integrate the advantages of biological brains with their low rates of energy in new neuromorphic computer architectures. Quantum computing will also offer new computational technologies for AI.

Artificial intelligence is already a key technology that will determine the global competition of societal systems. The wealth of nations will depend decisively on their power of AI-innovation. But, their way of life will depend on their evaluation of AI-Technology. Will our political systems change under the influence of a dominating AI-technology? How are we supposed to assert our individual freedoms in the AI world? Europe will have to position itself not only as a technical AI location, but also with its moral value system.

Since my early studies as student, I was fascinated by the algorithms that make artificial intelligence possible. We need to know their foundations in order to assess their performance and limitations. Surprisingly, this is an essential insight of this book, no matter how fast supercomputers are, they do not change the logical-mathematical foundations proven by human intelligence. Only on the basis of this knowledge, societal impacts can be assessed. For this purpose, we had already founded the Institute for Interdisciplinary Computer Science at the University of Augsburg at the end of the 1990s. At the Technical University of Munich, I was also head of the Carl von Linde Academy and, as part of the Excellence Initiative 2012, founded the Munich Center for Technology in Society (MCTS). In 2019, I was inspired by a research project of the Volkswagen-Stiftung on the topic “Can software be responsible?” As a member of a High Level Group (HLG) of the German Ministery of Economy and the DIN Commission, we work on an roadmap of AI-certification. In the thematic network of the German Academy of Science and Technology (acatech), there is also “Technology in Focus - Data Facts Background”, as this new book series is called by Springer. As a long-time author at Springer publisher, I thank the publisher for the proven support of the English translation of the German 2nd edition 2019.

Munich Klaus Mainzer
in June 2019

Contents

1

Introduction: What Is AI?

After the ringing of my alarm clock has shocked me a little rough, the familiar and friendly female voice of Susanne wishes me a good morning and asks me how I slept. Somewhat sleepy I inquire after my appointments this morning. Susanne reminds me of an appointment in Frankfurt at our branch. Friendly, but certainly reminds me of the exercise training prescribed by a doctor. I look at my wristwatch, which shows my current blood pressure and blood values. Susanne's right. I'd have to do something. Susanne and the alarm clock are in my smartphone, which I put in my pocket after showering and dressing and hurry to the car. Turned to the cockpit of my car, I briefly explain my destination. Now I have time for a coffee and read the newspaper relaxed. My car's heading for the freeway on its own. On the way, the car evades a construction vehicle. It complies with the traffic regulations in an exemplary manner and nevertheless makes better progress than some human drivers who want to be faster stressed with excessive speed, flashing lights and too short distances. People are just chaotic systems, I still think. Then I ask Susanne to give me market profiles of our products, which she filters out with Big Data algorithms at lightning speed. Arriving at the Frankfurt branch I have the car parked independently. Semiconductor production in our plant is largely automatic. Special customer requests can also be entered online in the sales department. The production then adapts itself independently to these special wishes. Next week I want to go to Tokyo and meet our Japanese business partner. I still have to ask him not to put me in one of the new robot hotels. The last time I checked in, everything was automatic, just like checking in at the airport. Even in the reception a friendly robot lady sat. With human service, it'll be a little more expensive. But here I am European "old-fashioned" and at least in my private life I prefer human affection …

K. Mainzer, *Artificial intelligence – When do machines take over?*, Technik im Fokus,
https://doi.org/10.1007/978-3-662-59717-0_1

That wasn't science fiction scenario. These were AI technologies which are technically feasible today and which, as part of the field of computer science and engineering, can be developed. Traditionally, AI (Artificial Intelligence) as a simulation of intelligent human thinking and acting. This definition suffers from the fact that "intelligent human thinking" and "acting" are not defined. Furthermore, man is made the yardstick of intelligence, although evolution has produced many organisms with varying degrees of "intelligence". In addition, we have long been surrounded in technology by "intelligent" systems which, although they are independent and efficient but often different from humans in controlling our civilization.

Einstein has answered the question "What is time?" independent of human imagination: "Time is what a clock measures." Therefore, we propose a working definition that is independent of human beings and only depends on measurable quantities of systems [1]. To this end, we look at systems that can solve problems more or less independently. Examples of such systems could be organisms, brains, robots, automobiles, smartphones or accessories that we wear on our bodies (wearables). Systems with varying degrees of intelligence are also available at factory facilities (industry 4.0), transport systems or energy systems (smart grids) which control themselves more or less independently and solve central supply problems. The degree of intelligence of such systems depends on the degree of self-reliance, the complexity of the problem to be solved and the efficiency of the problem-solving procedure.

So there is not "the" intelligence, but degrees of intelligence. Complexity and efficiency are measurable variables in computer science and engineering. An autonomous vehicle then has a degree of intelligence that depends on its ability to reach a specified destination independently and efficiently. There are already more or less autonomous vehicles. The degree of their independence is technically precisely defined. The ability of our smartphones to communicate with us is also changing. In any case, our working definition of intelligent systems covers the research that has been working successfully for many years in computer science and technology under the title "Artificial Intelligence" and is developing intelligent systems [2].

▶ **Working definition** A system is called intelligent when it can solve problems independently and efficiently. The degree of intelligence depends on the degree of autonomy of the system, the degree of complexity of the problem and the degree of efficiency of the problem-solving procedure.

It is true that intelligent technical systems, even if they have a high degree of independent and efficient problem solving, were ultimately initiated by people. But even human intelligence has not fallen from the sky and depends on specifications and limitations. The human organism is a product of evolution that is full of molecularly and neuronally encoded algorithms. They have developed over millions of years and are only more or less efficient. Randomness often played along. This has resulted in a hybrid system of abilities that by no means represents "the" intelligence at all. AI and technology have long since overtaken natural skills or solved them differently. Think of the speed of data processing or storage capacities. There was no such thing as "consciousness" as necessary for humans. Evolutionary organisms such as stick insects, wolves or humans solve their problems in different ways. In addition, intelligence in nature is by no means dependent on individual organisms. The swarm intelligence of an animal population is created by the interaction of many organisms, similar to the intelligent infrastructures that already surround us in technology and society.

Neuroinformatics attempts to understand the functioning of nervous systems and brains in mathematical and technical models. In this case, AI researchers work like natural scientists who want to test models of nature. This can be interesting for the technology, but does not have to be. AI researchers often work as engineers who find effective solutions to problems independently of the natural model. This also applies to cognitive skills such as seeing, hearing, feeling and thinking, such as modern software engineering shows. Even in the case of flying, the technology was only successful when it had understood the laws of aerodynamics and, for example, had developed other solutions with jet aircraft than in evolution.

In Chap. 2, we begin with a “Brief History of AI,” linked to the great computer pioneers of the 20th century. The computer was first taught to reason logically. The computer languages developed for this purpose are still used in AI today. Logical-mathematical reasoning leads to automatic proofs that help save computer programs. On the other hand, their analysis is connected with deep-seated epistemological questions of AI (chap. 3). However, general methods are not sufficient to solve specific problems in different specialist areas. Knowledge-based expert systems simulated diagnoses by doctors and analyses by chemists for the first time. Today, expert systems are part of everyday life in research and work, without still being called “artificial intelligence” (chap. 4). One of the most spectacular breakthroughs of AI are speech processing systems, since language is traditionally considered the domain of man. The tools used show how different technology and evolution can solve problems (chap. 5).

Natural intelligence originated in evolution. It therefore makes sense to simulate evolution sing algorithms. Genetic and evolutionary algorithms are now also being used in technology. (chap. 6). Biological brains not only enable amazing cognitive performance such as seeing, speaking, hearing, feeling and thinking. They also work much more efficiently than energy-guzzling supercomputers. Neural Networks and learning algorithms are intended to decipher these abilities (chap. 7). The next step is humanoid robots in a human-like form that works together with people at work and in everyday life. In a stationary industrial robot, the work steps are defined in a computer program. Social and cognitive robots, on the other hand, must learn to perceive their environment, to decide independently and to act. This requires intelligent software with sensor technology to realize this kind of social intelligence (chap. 8).

Automobiles are already referred to as computers on four wheels. As autonomous vehicles, they generate intelligent behavior that is intended to more or less completely replace the human driver. Which application scenarios are associated with this in traffic systems? Like the swarm intelligence in nature, intelligence is not limited to individual organisms. In the Internet of Things, objects and devices can be equipped with intelligent software interfaces and sensors to solve problems collectively. A current example is the industrial Internet, in which production and sales are largely organized independently. A factory then becomes intelligent according to our working definition. In general, one speaks meanwhile of cyberphysical systems, smart cities, and smart grids (chap. 9).

Since its inception, AI research has been associated with great visions of the future of mankind. Will there be neuromorphic computers that can fully simulate the human brain? How do analogue processes of nature and digital technology differ? Will the technologies of artificial life converge with artificial intelligence? The book discusses new research findings on logical-mathematical fundamentals and technical applications of analog and digital techniques.

Despite the sobriety of everyday AI research, hopes and fears motivate and influence the development of high-tech societies. Especially in the strongholds of American information and computer technology such as Silicon Valley, one believes in a singularity when AI will replace humans. We are already talking about a collective superintelligence.

On the one hand, superintelligence, as shown in this book, would also be subject to the laws of logic, mathematics, and physics. We therefore need interdisciplinary basic research so that the algorithms do not get out of hand. On the other hand, we demand technical design: After the experiences of the past, we should recognize the chances, but also consider exactly for which purpose and use we should develop AI in the future. AI must prove itself as a service in society [2]. That is their ethical yardstick (chap. 10).

References

1. Mainzer K (2003) AI – Artificial Intelligence. Foundations of Intelligent Systems. Wissenschaftliche Buchgesellschaft, Darmstadt
2. DFKI (German Research Center for Artificial Intelligence). http://www.dfki.de/web. Accessed 8 Jan 2016

A Short History of the AI

2

2.1 An Old Dream of Mankind

An automaton is in ancient usage an apparatus that can act independently (autonomously). According to ancient understanding, self-activity characterizes living organisms. Reports on hydraulic and mechanical automats are already mentioned in ancient literature against the background of the technology of the time. In Jewish tradition, at the end of the Middle Ages, the Golem was described as a human-like machine. The Golem can be programmed with combinations of letters from the “Book of Creation” (Hebrew: Sefer Jezira)—to protect the Jewish people in times of persecution.

At the beginning of modern times, automation was approached from a technical and scientific point of view. From the Renaissance, Leonardo da Vinci’s construction plans for vending machines are known. In the Baroque era, slot machines were built on the basis of watchmaking technology. P. Jaquet-Droz designs a complicated clockwork that was built into a human doll. His “androids” play the piano, draw pictures, and write sentences. The French physician and philosopher J. O. de Lamettrie sums up the concept of life and automata in the age of mechanics: “The human body is a machine that tensions its (drive) spring itself” [1].

The baroque universal scholar A. Kircher (1602–1680) already promotes the concept of a universal language in which

K. Mainzer, *Artificial intelligence – When do machines take over?*, Technik im Fokus,
https://doi.org/10.1007/978-3-662-59717-0_2

all knowledge is to be represented. Here, the philosopher and mathematician G. W. Leibniz directly follows and designs the momentous program of a “Universal Mathematics” (mathesis universalis). Leibniz (1646–1716) wants to trace thinking and knowledge back to arithmetic, in order to be able to solve all scientific problems by arithmetic calculations. In his age of mechanics, nature is imagined as a perfect clockwork in which every condition is determined as if by interlocking gears. Accordingly, a mechanical calculating machine executes each calculation step of a calculation sequence one after the other. Leibnizen’s decimal machine for the four basic arithmetic operations is the hardware of his arithmetic calculations. Fundamental is the idea of a universal symbolic language (lingua universalis) in which our knowledge can be represented according to the model of arithmetic and algebra. What is meant is a procedure by which “truths of reason, as in arithmetic and algebra, can also be achieved, so to speak, by a calculus in any other area in which it is concluded” [2].

The further technical development from decimal calculators for the four basic arithmetic operations to program-controlled calculators did not take place in the scholars’ room, but in the manufactories of the 18th century. There, the weaving of fabric samples is first controlled by rollers based on baroque slot machines, then by wooden punch cards. This idea of program control applies the British mathematician and engineer C. Babbage (1792–1871) on calculating machines. His Analytical Engine provided, in addition to a fully automatic calculation unit consisting of gears for the four basic arithmetic operations and a number memory, a punched card control unit, a data input device for numbers and calculation instructions, and a data output device with printing unit [3]. Although the technical functionality was limited, the scientific and economic significance of sequential program control in the age of industrialization is correctly recognized.

Babbage also philosophizes about analogies and differences between his calculating machines and living organisms and humans. His comrade-in-arms and partner Lady Ada Lovelace, daughter of the romantic poet Lord Byron, already prophesied:

"The Analytical Engine will process things other than numbers. When one transfers pitches and harmonies to rotating cylinders, this machine could compose extensive and scientifically produced pieces of music of any complexity and length. However, it can only do what we know to command it to do" [4]. In the history of AI, this argument of Lady Lovelace is mentioned again and again when it comes to the creativity of computers.

Electrodynamics and the electro-technical industry in the second half of the 19th century laid new technical foundations for the construction of computers. While Hollerith's tabulation and counting machine was being used, the Spanish engineer Torres y Quevedo thought about control problems for torpedoes and boats and constructed the first chess machine in 1911 for a final chess position with tower king vs. king.

Light and electricity also inspire writers, science fiction authors, and the beginning film industry. In 1923, the Czech writer Capek invented a family of robots. to free humanity from hard labor. After all, at least in the novel, the robots were provided with emotions. As machine men, they could no longer endure their slavery and rehearse the rebellion against their human masters. In the cinemas, movies like "Homunculus" (1916), "Alraune" (1918) and "Metropolis" (1926) were shown.

In industry and military research, the first special computers for limited computing tasks were built in the 1930s. However, the development of universal program-controlled computers, which can be programmed for different applications, will be fundamental for AI research. In April 11, 1936, the German engineer K. Zuse (1910–1995) applied for a patent for his "Methods for the automatic execution of calculations with the aid of calculating machines" [5]. In 1938, the Z1 was the first mechanical version to be completed, which was replaced in 1941 by the Z3 with electromechanical relay switches.

In 1936, for the first time, the British logician and mathematician A. M. Turing (1912–1954) defined the logical-mathematical concept of a computer: What is an automatic computational method, independent of its technical implementation? Turing's ideal computing machine requires an unlimited memory and only the smallest and simplest program commands, to which in principle any computer program, no matter how complicated, can be traced [6].

2.2 Turing Test

AI research in the narrower sense was born in 1950, when Turing published his famous essay “Computing Machinery and Intelligence” [7]. Here you will find the so-called “Turing Test”. A machine is called “intelligent” if and only if an observer is unable to tell whether he is dealing with a human being or a computer. Observer and test system (human or computer) communicate via a terminal (today, e.g., with keyboard and screen). In his work, Turing presents sample questions and sample answers from various fields of application such as:

Example

Q Please write me a poem about the Firth of Forth bridge.
A I have to pass on this point. I could never write a poem.
Q Add 34,957 to 70,764.
A (waits about 30 s and then gives the answer) 105.721.
Q Do you play chess?
A Yes.
Q My king stands on e8; otherwise I have no more figures. All they have left is their king on e6 and a tower on h1. It’s your move. How do you draw?
A (after a pause of 15 s) Th1-h8, matt.

Turing is convinced in 1950: “I believe that at the end of this century the general views of scholars will have changed to such an extent that one will be able to speak of thinking machines without contradiction”. The fact that computers today calculate faster and more accurately and play chess better can hardly be denied. But people also err, deceive, are inaccurate and give approximate answers. This is not only a shortcoming, but sometimes even distinguishes them in order to find their way in unclear situations. In any case, these reactions should be able to be realized by a machine. The fact that Turing's test system did not want to write a poem, i.e. did not pass Lady Lovelace's creativity test, could hardly shake Turing. Which person is already creative and can write poems?

2.3 From "General Problem Solver" to Expert System

When in 1956 leading researchers like J. McCarthy, A. Newell, H. Simon et al. met at the Dartmouth conference on machine intelligence, they were inspired by Turing's question "Can machines think?" Characteristic was the interdisciplinary composition of this conference of computer scientists, mathematicians, psychologists, linguists and philosophers. Thus the group around the universally educated H. Simon, the later Nobel Prize winner for economics, advocated a psychological research program to investigate cognitive processes of human problem and decision making on the computer.

The first phase of AI research (around the mid-1950s to mid-1960s) is still dominated by euphoric expectations [8, 9]. Similar to Leibnizen's Mathesis Universalis, general problem-solving procedures are to be used for computers. After Newell, Shaw and Simon had developed the LOGICAL THEORIST in 1957, a proof program for the first 38 propositions from Russell's and Whitehead's logic book "Principia Mathematica", the GPS (General Problem Solver) program was to determine the heuristic basis for human problem solving at all in 1962. The disappointment was great given the practical results. The first specialized programs such as STUDENT for solving algebra tasks or ANALOGY for pattern recognition of analog objects proved more successful. It was found that successful AI programs depend on appropriate knowledge bases ("databases") and fast retrieval procedures.

In the second phase of the AI (around the mid-1960s to mid-1970s), an increased trend towards practical and specialized programming can be observed. Typical are the construction of specialized systems, methods for knowledge representation and an interest in natural languages. At MIT J. Moser developed the program MACSYMAL, which was actually a collection of special programs for solving mathematical problems in the usual mathematical symbolism. Further programs of this kind (e.g. for integration and differentiation) are still in practical use today.

In 1972, Winograd presented a robotics program to manipulate differently shaped and colored building blocks with a magnetic arm. For this purpose, the building blocks with their properties and locations were represented in data structures. Programming of the location information was carried out with the magnetic arm by changing the building blocks.

In the third phase of AI (around the mid-1970s to mid-1980s), knowledge-based expert systems that promised the first practical applications move to the fore. The delimited and manageable specialist knowledge of human experts such as engineers and doctors should be made available for daily use. Knowledge-based expert systems are AI programs that store knowledge about a specific field and automatically draw conclusions from that knowledge, in order to find concrete solutions or provide diagnoses of situations.

In contrast to the human expert, the knowledge of an expert system is limited. It has no general background knowledge, no memories, no feelings and no motivations, which can be different from person to person despite common special knowledge: An elderly family doctor who has known a family for generations will use different background knowledge in the diagnosis of a family member than the young specialist who has just left university.

Knowledge is a key factor in the representation of an expert system. We distinguish between two types of knowledge. One kind of knowledge concerns the facts of the field of application, which are recorded in textbooks and journals. Equally important is the practice in the respective area of application as knowledge of the second kind. It is heuristic knowledge on which judgement and any successful problem-solving practice in the field of application are based. It is experiential knowledge, the art of successful presumption, which a human expert only acquires in many years of professional practice.

E. A. Feigenbaum, one of the pioneers of this development, compared the development of knowledge-based expert systems in the mid-1980s with the history of the automotive industry. In the world of AI, it would be 1890, so to speak, when the first automobiles appeared. They were manually operated horseless cars, but already automobiles, i.e. self-driven vehicles. Just as Henry Ford had the first prototypes for mass production in his day, Feigenbaum also said that knowledge-based systems would go into mass production. Knowledge-based systems were thus understood as "automobiles of knowledge" [10].

References

1. de La Mettrie JO (2009) L'homme machine/The machine man. Meiner, Hamburg (translated and ed by C. Becker)
2. Leibniz GW (1996) Philos. Schr. VII. Olms, Hildesheim, p 32 (ed C.I. Gerhardt, repr.)
3. Babbage C (1975) On the mathematical powers of the calculating engine. In: Randell B (ed) The origins of digital computers—selected papers. Springer, Berlin, pp 17–52 (unpublished manuscript, Dec. 1837)
4. Lovelace C (1842) Translator's notes to an article on Babbage's analytical engine. Sci Mem 3:691–731 (ed V. R. Taylor)
5. Zuse K (1936) Procedure for the automatic execution of calculations with the aid of calculating machines. German patent application Z 23624 (April 11, 1936)
6. Turing AM (1936–1937) On computable numbers, with an application to the decision problem. Proc London Math Soc Ser 2(42):230–265
7. Turing AM (1987) Computing machinery and intelligence (1950). In: Turing AM (ed) Intelligence service. Fonts. Brinkmann and Bose, Berlin, pp 147–182
8. Feigenbaum EA, Feldman J (1963) Computers and thought. McGraw-Hill, New York
9. Mainzer K (1985) The concept of intelligence from the point of view of epistemology and science theory. In: Strombach W, Tauber MJ, Reusch B (eds) The concept of intelligence in the various sciences. Series of the Austrian computer society, vol 28. Oldenbourg, Vienna, pp 41–56
10. Further historical notes of AI in K. Mainzer (1994), Computer—new wings of the mind? The evolution of computer-aided technology, science, culture and philosophy. De Gruyter, Berlin, p 103 ff. The development of AI in the 1990s is documented in the AI journal "Artificial Intelligence" of the German Society of Informatics e. V

Logical Thinking Becomes Automatic

3

3.1 What Does Logical Reasoning Mean?

In the first phase of AI research, the search for general problem-solving methods was successful at least in formal logic. A mechanical procedure was specified to determine the logical truth of formulas. The procedure could also be executed by a computer program and introduced automatic proving in computer science.

The basic idea is easy to understand. In algebra, letters x, y, $z\ldots$ are used by arithmetic operations such as add (+) or subtract (−). The letters serve as spaces (variables) to insert numbers. In formal logic, propositions are represented by variables A, B, $C\ldots$, which are connected by logical connectives such as “and” ($\land$), “or” ($\lor$)′ “if-then” ($\rightarrow$), “not” ($\neg$). The propositional variables serve as blanks to use statements that are either true or false. For example, the logical formula $A \land B$, by using the true statements $1+3=4$ for A and $4=2+2$ for B, is transformed into the true statement $1+3=4 \land 4=2+2$. In arithmetic, this leads to the true conclusion $1+3=4 \land 4=2+2 \rightarrow 1+3=2+2$. But, in general, the conclusion $A \land B \rightarrow C$ is not true. On the other hand, the conclusion is $A \land B \rightarrow A$ is logically generally valid, since for the insertion of any true or false statements for A and B there is always a true overall statement.

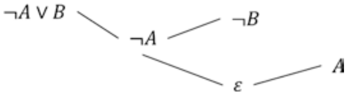
The proof of the general validity of a logical conclusion can be very complicated in practice. Therefore, in 1965, J.A.

K. Mainzer, *Artificial intelligence – When do machines take over?*, Technik im Fokus,
https://doi.org/10.1007/978-3-662-59717-0_3

Robinson proposed the so-called resolution method, according to which proofs can be found by logical refutation procedures [1–3]. One thus starts with the assumption of the opposite (negation), i.e. the logical conclusion is not generally valid. In the next step it is shown that all possible application examples of this assumption lead to a contradiction. Therefore, the opposite of negation is true and the logical conclusion is generally valid. Robinson's resolution method uses logical simplifications, according to which any logical formula can be converted into a so-called conjunctive normal form. In propositional logic, a conjunctive normal form consists of negated and non-negated propositional variables (literals), which are connected by conjunction ($\wedge$) and disjunction ($\vee$).

Example

For the conjunctive normal form $(\neg A \vee B) \wedge \neg B \wedge A$ the formula consists of the clauses $A \vee B$, $\neg B$ and A, which are connected by the conjunction $\wedge$. In this example, the literal $\neg A$ follows logically from the conjugation elements $\neg A \vee B$ and $\neg B$. (The reason is simple: The conjunction $B \wedge \neg B$ is always wrong for each application example for B and $\neg A$ follows logically from $\neg A \wedge \neg B$). From $\neg A$ and the remaining clause A, in the next step, the always wrong formula $\neg A \wedge A$ follows, and thus the contradiction ε ("empty word"):



Mechanically, therefore, the procedure consists of deleting contradictory partial propositions from conjunctive elements of a logical formula ("resolution") and repeating this procedure with the resulting "resolvent" and another corresponding conjunctive element of the formula until a contradiction (the "empty" word) can be derived.

In a corresponding computer program, this procedure terminates for the propositional logic. Thus, it shows in finite time whether the presented logical formula is generally valid. However, the calculation time increases exponentially with the number of literals of a formula according to the previously known methods. Concerning “Artificial Intelligence”, computer programs with the resolution method can automatically decide about the general validity of logical conclusions at least in the propositional logic in principle. People would have great difficulty keeping track of complicated and long conclusions. In addition, people are much slower. With increasing computing capacity, machines can therefore much more efficiently perform this task of logical concluding.

In predicate logic, statements are broken down into properties (predicates), which objects are assigned to or denied. Thus, in the statement $P\ (a)$ (e.g. “Anne is a student”), the predicate “student” (P) were assigned to an individual named “Anna” (a). This statement is either true or false again. In a predicative form of statement $P(x)$, blank spaces (individual variables) x, y, z… are used for the individuals a, b, c… of an assumed application domain (e.g., the students of a university). Beside logical connectives of propositional logic, now also quantifiers $\forall x$ (“For all x”) and $\exists x$ (“There is a x”) may be used. For example, $\forall x\, P(x) \rightarrow \exists x\, P(x)$ is a generally valid conclusion of predicate logic.

For the formulae of predicate logic, a generalized resolution procedure can also be indicated in order to derive a contradiction from the assumption of the general invalidity of a formula. For this purpose, a formula of predicate logic must be transformed into a normal form from whose clauses a contradiction can be derived mechanically. Since, however, in predicate logic (in contrast to propositional logic) the general validity of a formula cannot be decided in general, it can happen that the resolution procedure does not come to an end. The computer program then continues to run infinitely. It is then important to find subclasses in which the procedure not only terminates efficiently, but also at all. Machine intelligence can indeed increase the efficiency of decision-making processes and accelerate them. However, it is also (like human intelligence) restricted by the principle limits of logical decidability.

3.2 AI Programming Language PROLOG

To solve a problem with a computer, the problem must be translated into a programming language. One of the first programming languages was FORTRAN where a program consists of a sequence of commands to the computer like "jump to the position z in the program", "write the value a into the variable x". The focus is on variables, i.e. register or memory cells in which input values are stored and processed. Because of the commands entered, one also speaks of an imperative programming language.

In a predicative programming language, on the other hand, programming is understood as proving in a system of facts. This knowledge representation is well familiar from logic. A corresponding programming language is called "Programming in Logic" (PROLOG), which has been in use since the early 1970s [4–6]. The basis is the predicate logic, which we have already got to know in Sect. 3.1. Knowledge is represented in predicate logic as a set of true statements. Knowledge processing is what AI research is all about. Therefore PROLOG became a central programming language of AI.

Here we want to introduce some modules of PROLOG in order to clarify the connection with knowledge processing. The logical statement "The objects O_1, …, O_n stand in the relation R" corresponds to a fact, which in predicate logic is given the general form R(O_1, …, O_n). In PROLOG you write:

NAME(O_1, …, O_n),

where "NAME" is any name of a relation. Strings that are represented in the syntactic form of facts are called literals.

Example

An example of a fact or literal is:

married (socrates, xantippe),
married (abélard, eloise),
is a teacher (socrates, plato),
is a teacher (abélard, eloise).

Statements and proofs about given facts now can be represented into question-answer-systems. Questions are marked with a question mark and outputs of the program with an asterisk:

? married (socrates, xantippe),
* yes,
? is a teacher (socrates, xantippe),
* no.

Questions can also refer specifically to objects for which variables are used in this case. In programming languages, descriptive names are used for this, such as "man" for any man or "teacher" for any teacher:

? married (man, xantippe),
* man = socrates,
? is a teacher (teacher, plato),
* teacher = socrates.

In general, a question in PROLOG is "are L_1 and L_2 and… and L_n" or in short:

? $L_1, L_2, \ldots, L_n$

where $L_1, L_2, \ldots, L_n$ are literals. Logical concluding rules such as the direct conclusion (modus ponens) are "if L_1 and L_2 and… and L_n is true, then L is also true" or in short:

$$L: - L_1,\ L_2,\ \ldots,\ L_n$$

Example

This is how the rule can be introduced:
is a pupil (pupil, teacher):- is a teacher (teacher, pupil)
Then, it follows from the given facts:

? is pupil (pupil, socrates),
* student = plato

Based on a given knowledge base in the form of literals, PROLOG can find solutions to a question or problem using the resolution method.

3.3 AI Programming Language LISP

As an alternative to statements and relations, knowledge can also be represented by functions and classifications such as those used in mathematics. Functional programming languages therefore do not regard programs as systems of facts and conclusions (such as PROLOG), but as functions of sets of input values in sets of output values. While predicative programming languages are involved in predicate logic, functional programming languages are based on the λ-calculus which A. Church defined in 1932/1933 for the formalization of functions with calculation rules [7]. One example is the functional programming language LISP, which was developed by J. McCarthy as early as the end of the 1950s during the first AI-phase [8, 9]. Therefore it is one of the oldest programming languages and was connected from the beginning with the goal of artificial intelligence to bring human knowledge processing to the machine. Knowledge is represented by data structures, knowledge processing by algorithms as effective functions.

Linear lists of symbols are used as data structures in LISP ("List Processing Language"). The smallest (indivisible) building blocks of LISP are called atoms. It can be numbers, sequences of numbers or names. In arithmetic, the natural numbers are generated by counting, starting with the "atom" of one (1) and then step by step the successor $n+1$ by adding the one from the predecessor number n. Therefore, arithmetic properties are defined inductively for all natural numbers: You first define a property for the one. In the inductive step, under the condition that the property is defined for an arbitrary number n, it is also defined for the successor $n+1$ defined. Inductive definitions can be generalized for finite symbol sequences. Thus, s-expressions ("s" for "symbolic") are formed from the atoms inductively as objects of LISP:

▶ 1. An atom is an s-expression.
2. If x and y are s-expressions, then also (x.y).