

ARTIFICIAL INTELLIGENCE AND CREATIVITY

ANGELO MONTANARI

SUMMARY: 1. *Introduction*. 2. *Model-Based AI*. 3. *Data-Driven AI*. 4. *AI and Creativity*.

1. INTRODUCTION

SINCE its inception, artificial intelligence (AI for short) developed along two significantly different directions: symbolic AI, aka model-based AI, and sub-symbolic AI, aka data-driven AI. The focus of symbolic AI is on knowledge representation and automated reasoning methods and techniques, while sub-symbolic AI relies on neural networks, genetic algorithms, and machine/deep learning. The former adopts mathematical logic as the most suitable formalism to represent human knowledge and logical methods as the most appropriate tools to reason about it; the latter aims at mimicking the behavior of the human brain and of its complex network of interconnected neurons. In the last years, data-driven AI is becoming dominant, thanks to its undeniable successes in a large number of application domains. However, there is a significant trade-off between the efficacy/efficiency of sub-symbolic AI and its interpretability, that is, the possibility for a human being to readily understand the computation processes that yield the outcomes of the systems (decisions, classification, and predictions). Such a lack of interpretability turns out to be a major issue, especially in safety-critical domains, and led to the emergence of the research area of explainable/trustworthy AI.

In this paper, we first explore model-based (symbolic) AI, pointing out its strength and limitations. Then, we take into consideration data-driven AI, giving a short account of machine and deep learning and discussing some practical and theoretical limitations of it. Finally, we focus on the specific issue of creativity in AI.

2. MODEL-BASED AI

The beginning of symbolic (model-based) AI is commonly traced back to the paper by Turing on the relationship between computer systems and intelligence.¹ In fact, in his paper Turing wonders about the possibility of making

angelo.montanari@uniud.it, Dipartimento di Scienze Matematiche, Informatiche e Fisiche, Università degli Studi di Udine, Via delle Scienze 206, 33100 Udine, Italia.

¹ A. M. TURING, *Computing Machinery and Intelligence*, «Mind», 59 (1950), pp. 433-460.

an artificial system intelligent, but he does not explicitly outline a way to reach it, even though the close relationship it establishes between intelligence and language actually assigns a fundamental role to the symbolic human dimension. A few years later, Newell and Simon formulated the so-called 'physical symbol system hypothesis', which states that the ability of processing structures of symbols (expressions) is enough to make a computer system intelligent; moreover, they claimed that the same kind of symbolic manipulations are at the basis of human intelligence.² Since then, a lot of work has been done to provide artificial systems with the ability of performing advanced tasks involving symbolic representation and reasoning methods. Among them, we would like to mention automated planning, scheduling of activities, diagnosis capabilities, and configuration systems. Some important achievements of this AI research areas are expert systems and multiagent systems, including advanced autonomous systems, in particular robotic systems. In the following, we first give a short account of symbolic AI methods and techniques, and then we make a critical analysis of its potentialities and weaknesses.

2. 1. *A Short Account of Symbolic AI*

In his fundamental work on computing machinery and intelligence, Turing proposed a thought experiment (*Gedankenexperiment*), called the imitation game and, later, *Turing test*, where he tried to figure out the conditions under which one can speak of intelligence in reference to a machine.³ His statement can be summarized as follows: we can say that a machine is intelligent if a human being is not able to distinguish between its behaviour and the behaviour of an ordinary human being. As a matter of fact, in trying to characterize artificial intelligence, the experiment focuses on the 'intellectual' behaviour of a human being, implicitly assuming that it can be completely abstracted away from the other dimensions of human behaviour. This disembodied view of human intelligence is a main weakness of Turing test.⁴ Turing test takes place in three separate rooms. The human examiner is in the first room. In the other two rooms, there are a second person and the computer undergoing the test, respectively. The examiner, say A, knows the names of

² A. NEWELL, J. C. SHAW, H. A. SIMON, *Elements of a Theory of Human Problem Solving*, «Psychological Review», 65/3 (1958), pp. 151-166, and A. NEWELL, H. A. SIMON, *GPS: A Program that Simulates Human Thought*, in E. A. FEIGENBAUM, J. FELDMAN (eds.), *Computers and Thought*, McGraw-Hill, New York 1963.

³ A. M. TURING, *Computing Machinery and Intelligence*, cit.

⁴ One of the most significant achievements of AI research in the last decades is the awareness of the crucial role played by sense and actuation organs in the interaction of human beings with the external world. Artificial systems that aim at reproducing human behavior must necessarily be provided with sense and actuation capabilities connected to symbolic representation and reasoning ones.

the other two subjects, say B and C, but ignores who is the person and who is the computer. Both B and C interact with A in an independent way by means of a suitable communication channel. A may ask question to both B and C and read their answers. The task of A is to discover the identities of B and C within a fixed amount of time. A may ask any kind of question to B and C (apart from those that must be obviously excluded). Given the goal of the game, the computer undergoing the test will try to answer A questions in a way that conceals its identity, thus preventing A from identifying the nature of the two interlocutors. The machine passes the test successfully if A fails to identify it in the given amount of time. The test must be repeated several times, involving different examiners, to reduce the margins of subjectivity as much as possible.⁵

The influence of behaviorism, in vogue in the first half of the last century, on Turing test is quite evident.⁶ According to methodological behaviorism, introspection is not a suitable tool to get any scientifically reliable data; the only viable path is the analysis of the measurements of the stimuli (perceptions) provided to a man/animal and of their responses (actions) to the former.

Now, Turing test defines the conditions under which the behavior of an artificial system may be considered intelligent, but it does not explicitly suggest how such a behavior can be obtained. One important and influential contribution in this direction was given by *cybernetics*, that proposed the adoption of the animal/human being as a model for the study of control and communication mechanisms in artificial (intelligent) systems.⁷

Symbolic AI aims at reproducing human capabilities of representing and reasoning about knowledge in a symbolic way. To this end, in analogy to human beings, an artificial system must be able to build an internal symbolic representation of the external world, including both commonsense knowledge and knowledge about the specific domain of interest. Moreover, it must be able to reason about knowledge by means of rules that allow it to suitably manipulate the symbols. Different formalisms have been proposed over the years for knowledge representation, like, for instance, semantic networks⁸

⁵ The formulation of the game by Turing was slightly different: B and C are a man and a woman, and the examiner has to establish who is the man and who is the woman. Turing asked himself what would happen if one replaced the man with a computer, that is: would the number of successful decisions by the examiner be the same or not?

⁶ J. B. WATSON, *Psychology as a Behaviorist View*, «Psychological Review», 20/2 (1913), pp. 158-177.

⁷ N. WIENER, *Cybernetics or Control and Communication in the Animal and the Machine*, MIT Press, Cambridge MA-London 1962.

⁸ M. MINSKY, *A Framework for Representing Knowledge*, MIT-AI Laboratory Memo 306, Cambridge MA-London 1974.

and conceptual graphs,⁹ but all of them can be directly or indirectly brought back to logic. Similarly, reasoning about knowledge has been implemented by means of suitable logic mechanisms, including deductive reasoning, e.g., in automated planning, but also abductive reasoning, e.g., in diagnosis, and inductive reasoning, e.g., in machine learning. In general, each artificial reasoning systems is characterized by a set of logical rules that allow it to derive new pieces of knowledge from given ones.

Starting from the 1970s, rule-based systems, aka *expert systems*, have been developed and used in many contexts. The idea was to encode the knowledge of the experts of a specific application domain by means of a set of deductive condition-action rules that operate on a given knowledge base. By exploiting these rules, expert systems were able to answer queries about their domain of competence. Expert systems have been successfully applied in a variety of domains, including industrial and medical ones. Over the years, their original simple structure was enriched by introducing heuristics, probabilities, fuzziness, observations, and preferences. Natural descendants of expert systems are decision support systems, that assist decision makers by providing opinions and suggestions that take into account available data (observations) as well as preferences and expectations of the final users.

Expert systems have been progressively replaced by *multi-agent systems*,¹⁰ which can be seen as the ‘applicative’ counterpart of Minsky’s society of mind.¹¹ They got started from the notion of an intelligent artificial agent, that operates within a community of agents (agency), and established themselves as a unifying concept in symbolic AI.¹² An intelligent artificial agent can be viewed as a computational systems, that is, a software program with a hardware support, which (i) interacts with the surrounding environment (it is reactive), (ii) is able to make decisions and to act autonomously to achieve its own goals or goals negotiated with the other agents (it is proactive), and (iii) is able to communicate with the other agents, and/or with human beings, to execute coordination, cooperation, or negotiation activities (it makes social interactions). These capabilities are common to quite different types of agents that have been developed in AI, including robots, software agents, and virtual agents. Specific classes of agents may feature more advanced properties such as mobility, either in the physical

⁹ J. F. SOWA, *Conceptual Graph for a Database Interface*, «IBM Journal of Research and Development», 20/4 (1976), pp. 336-357.

¹⁰ The following account of multi-agent systems is a shortened version of the one given in A. MONTANARI, *Intenzioni e fini nei sistemi artificiali intelligenti*, in S. RONDINARA (ed.), *Scelte razionali, intenzionalità, fini*, Città Nuova, Roma 2014, pp. 156-181.

¹¹ M. MINSKY, *The Society of Mind*, Simon and Schuster, New York 1988.

¹² S. RUSSEL, P. NORVIG, *Artificial Intelligence: A Modern Approach*, 2 vols., Prentice Hall, Hoboken NJ 2003.

world (robotic systems) or in the network (software or virtual agents), and learning.

Until the 1980s, the vast majority of approaches to multi-agent systems were of symbolic nature. They are based on the 'reasoning and planning' paradigm, which requires that the agent be given a formal description of (i) the surrounding environment (set of states in which it may be), (ii) a goal (a state) that captures agent's intention, and (iii) the actions that can be carried out in the given environment to achieve the goal, each provided with its effects and possible preconditions. Effects and preconditions are defined in terms of the states of the environment. The explicit symbolic representation of agent's knowledge makes use of logical formalisms, such as, for instance, epistemic logics and descriptive logics, and it allows for the exploitation of formal reasoning techniques. Multi-agent systems consist of a set of agents that interact with each other and with the surrounding environment. The behavior of agents is characterized by a certain degree of independence, self-awareness, and autonomy. As a rule, no agent has complete knowledge of the whole system; it has only information on its duties and on the other agents with whom it has to interact. In most advanced systems, there is no specific agent in charge of centralized control of the whole system. Each agent interacts with the environment in an independent way, on the basis of its preferences about the states of the world. A crucial issue in multi-agent systems is how to achieve cooperation among agents. Here, the decisive element is the way in which they communicate. Communicative actions are usually included in the plans of the individual agents, which model the actions they can perform. The goal of multi-agent systems is to generate advanced processes of system organization and control even in the presence of extremely simple individual strategies of individual agents. Communication can be done via message passing, in accordance with specific communication protocols, or by using shared memories. Another distinctive feature of multi-agent systems is their flexibility, which encompasses both the ability to incorporate new agents and to modify existing ones, and the ability to reconfigure the system in the presence of failures and malfunctions.

2. 2. Practical and Theoretical Limitations of Symbolic AI

The main criticism addressed to a purely symbolic approach to multi-agent systems was that it works well only in the case of simple/simplified application domains, because it pays little attention to the surrounding environment. In complex scenarios, the intelligent behavior is inextricably linked to physical interaction with the environment and such a behavior emerges from the composition/interaction of simpler reactive behaviors that cannot be reduced to the manipulation of symbolic representations. Reactive agents choose their

actions of the basis of their current perceptions. These actions can either be defined by means of explicit (deductive) rules or by exploiting mathematical or probabilistic functions that make use of sub-symbolic knowledge. One of the most significant proposals is Brooks' subsumption architecture, where behaviors are organized in a hierarchy of levels (the agencies).¹³ Each behavior is a function that receives in input some perceptions and returns some corresponding actions as output. Each function is specialized in the execution of a particular elementary task. Agents are organized in tiered architectures. On the basis of the type of interaction among the levels, one can distinguish horizontal architectures, where the levels operate in parallel and contribute to the generation of the output, and vertical architectures, where each level processes the input and then transfers the control to the next level, up to the level that generates the output. This is the case of robotic systems, for instance, where available knowledge is organized and managed in a highly hierarchical way. The reasoning process moves from the level of perception to the level of symbolic representation and reasoning, and then back from it to the level of action. The relationships between the levels of perception and action and the level of symbolic representation and reasoning are managed by a 'magical' intermediate level. Horizontal architectures are much simpler to implement, but they often require a coordinator who ensures the consistency of processing.

A different and more fundamental criticism to symbolic AI comes from the thought experiment known as Searle's Chinese room.¹⁴ The target of Searle's attack was what he called 'strong AI'; however, his work was also a criticism of the behaviourist perspective of Turing test. As a matter of fact, Searle did not dispute the effectiveness of symbolic AI, but the possibility of reproducing human cognitive abilities through it (simulation as duplication). To this end, he introduced and distinguished two alternative approaches to AI, the strong and the weak/cautious one. According to the supporters of the strong approach, a properly programmed AI system is comparable to a mind, in the sense that the states of the system are comparable to cognitive/mental states and the abilities of the system are comparable to cognitive skills, such as, for instance, the abilities to understand, reason, and learn. According to this point of view, the structure of the artificial system, that is, the program in which it is articulated, can be used to explain the structure of the mind, that is, the cognitive abilities, of a human being. His experiment intends to disprove the perspective of strong AI. More precisely, his goal was to show that mental

¹³ R. A. BROOKS, *A Robust Layered Control System for a Mobile Robot*, «IEEE Journal on Robotics and Automation», 2/1 (1986), pp. 14-23.

¹⁴ J. R. SEARLE, *Minds, Brains, and Programs*, «Behavioral and Brain Sciences», 3/3 (1980), pp. 417-457.

processes cannot be reduced to computational processes that operate on formally defined elements. The experiment concerns the ability of human beings to demonstrate their understanding of a narrative text by suitably answering a number of questions about it thanks to both common knowledge and domain-specific one.¹⁵ According to Searle, his experiment proves that the ability to manipulate input information according to some well-defined formal rules is not sufficient to explain the process of understanding, as it lacks the fundamental dimension of intentionality. In fact, such a claim about the inadequacy of AI systems for the manipulation of formal symbols should be actually framed in the general issue concerning the incompleteness and, subordinately, the undecidability of powerful enough formal systems of computation.¹⁶ In such a perspective, the fact that a system for the manipulation of formal symbols does not achieve any form of understanding could be explained by the impossibility of defining a correct and complete computing system capable of capturing the process of human understanding (the same holds for the other cognitive abilities). It goes without saying that if such a system cannot exist, then it cannot be through such a system that human beings realize understanding and their other cognitive abilities.

As acknowledged by Searle himself, his experiment leaves open the problem of determining what differentiates the process of human understanding from a machine's ability of manipulating symbolic representations of information to execute a given task according to well-defined formal rules, and from the issue of establishing whether, and possibly how, what substantiates such a difference can be transferred to a machine. Searle simply states that there is no fundamental reason that prevents giving an (artificial) machine the ability of understanding, because, in a certain sense, an example of such a machine already exists: human beings with their body and mind. He brings back intentionality to the human biological structure, which, under suitable conditions, is able of causally producing perception, action, understanding, learning, and other forms of intentional behavior. Only something that has such 'causal powers' is able, according to Searle, of possessing forms of intentionality. He does not exclude the possibility that these causal powers, and, thus, some forms of intentionality and consciousness, can be obtained through physical-chemical principles/mechanisms other than those used by human beings (this

¹⁵ A detailed account of Searle's experiment can be found in A. MONTANARI, *Riduzionismo e non in Intelligenza Artificiale*, in L. GRION (ed.), *La differenza umana. Riduzionismo e antiumanesimo*, «Anthropologica» (2009), pp. 113-128.

¹⁶ For a short account of the relationships among formal systems, algorithms, and computable functions, we refer the reader to A. MONTANARI, *Per un vocabolario filosofico dell'informatica*, in G. CICHESSE, A. PETTOROSI, S. CRESPI REGHIZZI, V. SENNI (eds.), *Istanze epistemologiche e ontologiche delle scienze informatiche e biologiche*, Città Nuova, Roma 2011, pp. 82-106.

is an empirical question, he claims). What he radically excludes is the possibility that a formal model (system) can ever be sufficient by itself for intentionality, since formal properties are not in themselves constitutive of intentionality and have no causal power.

3. DATA-DRIVEN AI

In the last years, data-driven AI is taking a dominant role in the AI field. As a matter of fact, machine learning was present since the beginning of AI. Its current explosion and pervasiveness are motivated by at least two crucial factors: the strong increase in computational power of computer systems and the availability of a huge amount of information (texts, images, sound recordings, videos) on the net. The ability of systematically analyse huge collections of data (big data) to discover meaningful similarities among them and to make use of these similarities to identify new data of the same kind as well as to make reliable predictions made it possible to take a radical step forward in research and, especially, in applications. In the following, we will first give a short account of machine and deep learning methods and techniques, and then we will discuss a number of practical and theoretical issues.

3. 1. *A Short Account of Machine and Deep Learning*

Machine learning can be defined as a set of methods, techniques, and tools allowing a system to improve its ability to carry out some tasks by learning from experience.¹⁷ Such an improvement can obviously take different forms, like, for instance, an increase in coverage, an improvement in performance, or a quality growth. From the point of view of its operational structure, any machine learning system has to pass through a training phase, followed by a validation one, before becoming operational. Different machine learning algorithms have been proposed in the literature. A fundamental distinction is the one among supervised, unsupervised, and reinforcement learning.

In *supervised learning*, in order to learn a function that is capable of mapping every input into the corresponding output, the system is initially fed with a number of input-output example pairs (labeled training data, training set for short). Building on such instances, a model is derived that allows one to predict an output value for any input value, as close as possible to the provided labels. To determine the generalization capabilities of the model, the latter must then be validated on a test set, that is, a set of input-output pairs disjoint from the training set. The inputs may in general be different from

¹⁷ A comprehensive book on machine learning is S. SHALEV-SHWARTZ, S. BEN-DAVID, *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press, Cambridge 2014.

the ones encountered during the training phase, although they should still come from a similar distribution. Commonly used tools for supervised learning are decision tree and linear regression. In *unsupervised learning*, the goal is not to define an input-output mapping on the basis of a given set of known solution instances, but to determine any possible correlation among the elements of a given dataset. In such a case, by exploiting a sufficiently large number of examples, the system learns how to classify any new input, looking, for instance, at its similarity to previous inputs. Notice that, unlike the case of supervised learning, there is no distinction between training data and test data. Compared to supervised learning, unsupervised one is more general and flexible, but also much more complex. Clustering algorithms and methods are commonly exploited in unsupervised learning. *Reinforcement learning* can be viewed as an intermediate form of learning. Training data are richer than testing data, but the learner must, however, predict more information for the latter. Reinforcement learning assumes that the system is provided with tools that allow it to interact with the environment (sensors, cameras, GPS) to better understand its characteristics and to refine its learning abilities accordingly. The environment is often modeled as a Markov decision process, and dynamic programming techniques are exploited to build reinforcement learning algorithms.

The most successful tools exploited in machine learning are neural networks. Their systematic application to machine learning led to the emergence of the so-called *deep learning*. Thanks to programmable neural networks, artificial systems can take rather sophisticated and accurate decisions without any intervention from human beings. *Neural networks* have been devised as artificial models of the behavior of the network of human neurons. They actually define hierarchies of concepts that allow a system to learn complex concepts by building them out of simpler ones. These hierarchies are then structured in graphs that may be many layers deep.

Neural networks have a long history behind them.¹⁸ The first mathematical models of human neurons date back to the 1940s, with the proposal by McCulloch and Pitts,¹⁹ later improved by exploiting the so-called Hebb's rule.²⁰

¹⁸ There are various textbooks on neural networks in the literature. The most famous one is probably the following: I. GOODFELLOW, Y. BENGIO, A. COURVILLE, *Deep Learning* (Adaptive Computation and Machine Learning Series), MIT Press, Cambridge MA-London 2016. In our exposition, we benefit from the following lecture notes (in Italian): M. BUDINICH, *Introduzione alla teoria delle reti neurali*, Dipartimento di Fisica, Università degli Studi di Trieste-INFN, January 2022.

¹⁹ W. S. MCCULLOCH, W. PITTS, *A Logical Calculus of the Ideas Immanent in Nervous Activity*, «The Bulletin of Mathematical Biophysics», 5 (1943), pp. 115-133.

²⁰ D. HEBB, *The Organization of Behavior: A Neuropsychological Theory*, Wiley, New York 1949.

The basic ingredients of these models are the following. A *neuron* is modeled as an element with n binary inputs and 1 binary output. The two values represent the two possible states of each neuron: at rest or firing. A weight is associated with each input to model the different intensities of the synaptic contact. The neuron computes the weighted sum of the inputs (activation), which mimics the behavior of neurons, whose global potential difference is the sum of those of the active single synapses. If the activation is equal to or greater than a certain threshold value, the neuron fires. Compared to actual neurons, such a model makes a number of simplifying assumptions, like, for instance, binary inputs and output, linear sum, fixed delay, synchronicity, no relationship between the intensity of the stimulus and the frequency of firing. However, the goal of computer scientists is not to faithfully reproduce the structure and the behavior of neurons, but to develop artificial systems with the ability of managing a huge amount of information in an effective and efficient way to perform meaningful tasks of human beings like, e.g., speech recognition and computer vision.

How can a learning system be built out of McCulloch and Pitts model of neurons? The idea is to put together in a network a huge number of neurons that operate in parallel. Even though their basic structure is the same, different behaviors of the net can be obtained by assigning different weights to the neurons. In principle, for any given problem, one should find a suitable set of neurons, with appropriate weights, whose combined behaviors solves it. The problem is that there exists no general method to determine these weights. Learning can be viewed as a process that attempts to compute the right weights, that is, a way of programming artificial neural nets. It is accomplished by providing a set of solution examples (training set) from which the (supervised) learning process produces appropriate weights for the involved neurons, that is, weights that allow the network to compute the right solutions or, at least, to minimize a suitable error function. It is worth noticing that the resulting algorithm heavily relies (and depends) on the training set, and the way in which it is 'synthesized' makes it difficult to understand (interpret) the solutions it computes. Such a dependency on solution examples raises various issues, in particular, that of overfitting, that we will discuss later, and explain why the validation of the generated algorithm on a representative testing set is of crucial importance.

Ten years later (1958-1962), Rosenblatt proposed the *Perceptron*, that, when it was formulated, appeared to be the right solution.²¹ As the name suggests, the goal was to give an artificial system the ability of recognizing patterns that characterizes human beings. The Perceptron basically consists of a neuron

²¹ F. ROSENBLATT, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan, Washington DC 1962.

paired with a simple *learning rule*, which drew inspiration from Hebb's rule. The main difference between the original solution outlined by McCulloch and Pitts and the one devised by Rosenblatt can be summarized as follows: while the problem addressed by the former is that of finding (if it exists) a set of weights that, on a given set of examples, return the expected outputs, Rosenblatt aims at devising a procedure that, given an initial set of weights (all of them can be assumed to be equal to 0), progressively modifies them in order to find the solution set. To this end, Rosenblatt introduced a learning rule and proved that, for all problems for which a solution exists, a set of weights that solves the problem can be computed by a finite, bounded number of applications of such a learning rule. The bound on the number of learning steps depends on the distance between the set of examples, with their initial weights, and the solution. A few years later, Minsky and Papert pointed out some insuperable limits of the Perceptron.²² They indeed proved that the apparently harmless condition on the existence of a solution, which has a fairly natural geometrical counterpart, is actually very restrictive, as there are very few problems that satisfy it.²³ Such a discovery greatly reduced the expectations raised by Rosenblatt's result.

Various possible ways out were proposed in the following years, ranging from the replacement of the search for exact solutions with that for approximate ones, that minimize the number of errors (although this latter problem turns out to be computationally hard), to the proposal of more powerful neural networks. The first direction led to the introduction of continuous (and differentiable) error functions, that can be more easily dealt with analytical tools. One of the simplest and most successful techniques, that has been exploited to minimize error functions, is that of gradient descent, which is an iterative optimization algorithm for finding a local minimum of a differentiable function. Such an algorithm actually provides a learning rule for the continuous Perceptron. The second direction led to the replacement of the single Perceptron by a network of Perceptrons. This is the case with the so-called *feed-forward neural networks*, where information flows in one direction only, but the original simple model of a neuron is replaced by a layered structure featuring one input layer, one output layer, and one or more intermediate, hidden layers. It is possible to show that for any continuous function, there exists a feed-forward neural network with a single intermediate layer which is able to approximate it. Unfortunately, it does not follow that there exists a

²² M. L. MINSKY, S. A. PAPERT, *Perceptrons*, MIT Press, Cambridge MA-London 1969.

²³ A famous and simple example of a problem on which Rosenblatt's approach fails is that of computing the exclusive OR (XOR for short) of two Boolean variables (given two Boolean variables X_1 and X_2 , $X_1 \text{ XOR } X_2$ is true if and only if either X_1 is true or X_2 is true, but not both).

learning algorithm which is able to find such an approximation. The application of the technique of gradient descent to feed-forward neural networks resulted into the proposal of the fundamental algorithm of *back propagation*, still in use today.

The next step forward was performed by Hopfield in the 1980s.²⁴ Hopfield proposed a new model of recurrent neural network, later called *Hopfield network*. Unlike feed-forward neural networks, Hopfield ones feature a bidirectional flow of information. Information may indeed also flow from one layer to the previous one, from one neuron to another belonging to the same layer, and even from a neuron to itself. From the point of view of their structure, Hopfield networks are symmetric recursive neural networks, whose convergence is guaranteed. They make it possible the emergence of new computational capabilities from the collective behavior of a large number of simple processing elements. Among the emerging properties of such a collective behavior, there is some form of associative memory. According to Hopfield, a physical system can be considered as a memory device, if it features a certain number of stable states, which act as attractors for the system itself. Building on that, he states that the stability and location of these attractors are emerging properties of systems consisting of a huge number of mutually interacting neurons. A systematic account of the expressive power of the various recurrent neural networks proposed in the literature on the basis of the dynamics of their attractors has been recently given.²⁵ It basically shows that their computational power depends on the (synaptic) weights and the activation functions, and it ranges from finite state automata to Turing and super-Turing machines.

Since then, interesting new proposals have emerged, like, for instance, generative adversarial networks (GAN for short)²⁶ and transformers,²⁷ but no major breakthrough has happened in research on neural networks. Nevertheless, the last decade has seen the explosion of neural networks thanks to the availability of great computational power at affordable prices (think of the graphics processing units, GPUs for short). Deep learning algorithms proved themselves more performing than traditional ones in a number of application

²⁴ J. J. HOPFIELD, *Neural Networks and Physical Systems with Emergent Collective Computational Abilities*, «Proceedings of the National Academy of Sciences», 79/8 (1982), pp. 2554-2558.

²⁵ J. CABESSA, A. E. P. VILLA, *Expressive Power of First-Order Recurrent Neural Networks Determined by their Attractor Dynamics*, «Journal of Computer and System Science», 82/8 (2016), pp. 1232-1250.

²⁶ I. GOODFELLOW, J. POUGET-ABADIE, M. MIRZA, B. XU, D. WARDE-FARLEY, S. OZAIR, A. COURVILLE, Y. BENGIO, *Generative Adversarial Network*, «Advances in Neural Information Processing Systems», 27 (2014), pp. 2672-2680.

²⁷ A. VASWANI, N. SHAZEER, N. PARMAR, J. USZKOREIT, L. JONES, A. N. GOMEZ, L. KAISER, I. POLOSUKHIN, *Attention is All You Need*, 2017, arXiv:1706.03762.

domains ranging from natural language processing to online recommendation systems, from bioinformatics to videogames.

An interesting alternative approach to neural networks is that of genetic or *evolutionary algorithms* (EAs for short). It consists of population-based metaheuristics inspired by the process of biological evolution and genetics, that turned out to be quite successful in solving combinatorial optimization problems.²⁸ Unlike blind random search, EAs exploit historical information to direct the search towards the most promising regions of the search space, relying on methods that mimic the processes that lead to adaptive evolution in natural systems. In nature, the evolution of a population of individuals can be viewed as the process of increasing the level of adaptation to its environment; in a similar way, EAs deal with a population, whose individuals are possible solutions to a given optimization problem, and evaluate each individual with respect to its 'degree of adaptation' to the problem by means of a single- or a multi-objective fitness function. The EA population iteratively undergoes a series of generation steps; at each such step, individuals picked by a suitable selection strategy go through a process of reproduction. In this way, the elements of the population iteratively evolve toward better solutions. Evolutionary-based algorithms differs from each other in their selection strategy, even though in all of them individuals with high degree of adaptation are more likely to be chosen. During the reproduction process, crossover and mutation operators are applied, with a certain degree of probability. Crossover is the EA counterpart of natural reproduction, by which the characteristics of two individuals are combined by generating a new one. In general, high crossover rates tend to pull the population towards a local minimum or maximum. Mutation applies random changes to the encoding of the selected solution, with the goal of maintaining genetic diversity in the individuals. It prevents premature convergence of the algorithm to a local optimum, thus allowing it to more broadly explore the search space. In both cases, offspring are generated, which are merged with the previous population, and the cycle repeats until a stopping condition is met. Such a condition can be based on different criteria, like, for instance, population diversity or a given fitness threshold.

3. 2. PRACTICAL AND THEORETICAL ISSUES ON DATA-DRIVEN AI

Data-driven AI has to face various technical issues. Some of them are related to the quantity and the quality of available data.

A common one is the *lack of data*. It can be caused by the actual lack of relevant data or by the complexity of the process of data collection, which may

²⁸ A. E. EIBEN, J. E. SMITH, *Introduction to Evolutionary Computing*, Springer, Berlin-Heidelberg 2003.

be too difficult, expensive, or time-consuming. In many domains, the amount of available data may be insufficient to create large enough training datasets, whose size heavily influences the performance of machine learning models. To overcome this problem, data augmentation techniques are commonly exploited, which allow ones to artificially extend a dataset. A related problem is that of *data distribution*. The distribution of training data should indeed be basically the same as that of testing data, and both of them should reflect the one underlying the whole set of relevant data. To guarantee the ‘consistency’ of the distributions of training and testing data, one can generate both the training set and the testing set as simple random samples of the whole available dataset. The fact that their distribution reflects or not that of the actual data depends on the representativeness of the dataset. Paradoxically, the availability of a large amount of data that do not reflect the distribution of the actual one may be a problem rather than an advantage, as it may lead to a major technical problem called overfitting.

To guarantee the generality of the generated models and their ability to make appropriate estimations on real data, models that fit too well training data (*overfitting*) must be avoided. These models, indeed, rely too much on training data, including data noise, and they are not able to deal properly with new data. This is the case, for instance, with EAs. They tend to produce solutions that are as good as possible for a given training set, against which the fitness function is evaluated, without considering the performance on possible new cases. Various methods have been proposed to avoid overfitting. The simplest one, whenever possible, is to improve the training set in terms of both data quantity and variability. Another possibility is to introduce perturbations in data, e.g., by adding some noise to training data. In the case of EAs, a way of reducing overfitting and improving generalization is to make use, through the generations, of a small and frequently changing random subset of the training data.²⁹

A fundamental issue in data-driven AI is that of *interpretability*. The fact that a machine learning model performs well is not sufficient in many contexts. We cannot simply trust the model and ignore why a given decision has been taken. As an example, being able of explaining why a certain prediction has been made is of crucial importance in safety-critical domains; in some cases, interpretability is even more important than the predictive performance. The trade-off between efficiency/accuracy and interpretability is the reason why, in certain application domains, classical decision trees are preferred to neural networks. Besides safety issues, interpretability is essential to make it possible

²⁹ A survey on generalization in evolutionary computation can be found in V. K. DABHI, S. CHAUDHARY, *A Survey on Techniques of Improving Generalization Ability in Genetic Programming Solutions*, 2012, arXiv: 1211.1119.

to exploit fruitfully what one extracts from data and to make progresses on the basis of it. In particular, it supports one in the identification of relevant data that deserve, if possible, to be collected and analyzed. Notice that these data are not necessarily available; there are cases in which they need to be produced and/or measured.

As a matter of fact, interpretability is a big deal also in other data-intensive scientific fields, such as, e.g., physics. In order to successfully model a given phenomenon, it is crucial to identify the relevant variables (the phenomenon's 'essential ingredients'), a task where human intuition plays a crucial role. The amount of data, indeed, grows exponentially with the size of the problem (the number of variables), and may result in situations where there is no way to acquire and/or master them.

Besides the above issues, there is another one of a more foundational nature. To some extent, indeed, we are facing the same issues raised in respect to classical *inductive reasoning*. The question is thus whether automatic learning (machine learning and deep learning) suffers from the same limits as classical inductive reasoning. Since it is unreasonable to assume that, in a single initial step, it is possible to provide a system with a complete and unchangeable knowledge base from which any other piece of knowledge can be derived by means of appropriate procedures, the system must be able to automatically expand and possibly revise its knowledge base. This can be done by means of examples, experiments, and experiences. Various approaches to the process of knowledge acquisition have been proposed in the literature, including, as we have shown, symbolic learning techniques, neural networks, and evolutionary algorithms. However, all proposed solutions are somehow bound to failure due to problems related to the definition of the inductive bias and the choice of the training set.³⁰ It follows that it is difficult to maintain that a system can induce irrevocable rules of a general nature, which can be used to make predictions, on the basis of the regularities detected up to a certain moment.

We conclude this section by disputing a position of an ideological nature, which is fairly widespread not only among those who work in the area of big data, but also within the general scientific community, according to which machine learning and data science would be the new *data-driven paradigm* of science, destined to supplant the classic one based on the formulation of models and theories, starting from observations, and on their (experimental) validation/verification. The fundamental assumption of this position is that, having enough data with an adequate guarantee of statistical significance, it

³⁰ We can define the inductive bias of a learning algorithm as the set of assumptions made by the learner in order to predict the outputs of future inputs, that is, to deal with data not belonging to the training set. It is worth pointing out that inductive bias is inevitable and, to some extent, necessary to make learning algorithm effective and successful.

would be possible automatically to extract laws and models of general value from them, by means of learning algorithms and/or statistical analyses. As an example, such a position is supported by Kambhampati, who views (data-driven) AI as an ersatz natural science.³¹ In his opinion, the emergence of large learned systems “based on transformer architectures that are trained with billions of parameters over massive web-scale multimodal corpora” is changing the nature of AI research in an essential way. The nature and complexity of these large learned systems is such that not only we are not able to understand how the generated models are able to show certain capabilities (the issue of interpretability), but also we are not able to fully understand what are their capabilities. According to Kambhampati, systems with these characteristics are quite different from usual ones, which are built on the basis of a clear design to achieve some specific objectives (the ultimate goal of the designer is to build systems which are correct by construction), and their study looks rather like an (ersatz) natural science.

Such a point of view has to cope with the above-mentioned issues. Moreover, it has an intrinsic limit: a science built on available historical data is a science that only looks at the past, and thus inherently conservative, as it is ‘structurally’ forced to exclude novelties. It builds models of human beings and artificial systems by analyzing their past behavior, and it uses them to predict their future behavior. The problem is that future behavior is not necessitated by past one. Novelty is a distinctive feature of human behavior and human history, and, unless the future is fully determined by the past, it is also a distinctive feature of nature. As for machines, some form of rigid repetition can be assumed for simple (and closed) systems, but it is not a feature of complex open systems.

Regardless of this, in a sense, data do not exist per se. First of all, they depend on the observer, man or machine, who collects them. They also depend on measuring devices, as one can collect only data that can actually be measured/acquired. Finally, and most importantly, in order to establish which are the relevant data to collect (if it is possible to collect them), a goal and a plan to reach it are necessary, and both of them can only be formulated with reference to an explicit or implicit model of the world (a theory). Thus, there exist no ‘neutral data’, and existing data are both in fact and in principle incomplete.

4. AI AND CREATIVITY

Suitably rephrasing Turing’s question about intelligence, we can ask ourselves under which conditions one can speak of creativity in reference to an AI sys-

³¹ S. KAMBHAMPATI, *AI as (an Ersatz) Natural Science*, «Communications of the ACM, BLOG@CACM», June 8, 2022.

tem. The first observation is that creativity must not be confused with *uncontrollability*: the fact that the behavior of an artificial machine cannot be completely controlled does not mean that it is a creative behavior. The trade-off between the computational power of a machine and its controllability is well known and systematically studied in computer science. An improvement to the computational power of a machine usually causes an increase in its computational complexity, and, possibly, a reduction of its controllability. A similar consideration applies to the above-mentioned large learned systems.

Another notion that must be properly understood is that of *autonomy*. The degree of autonomy of agents is considered to be an important parameter in the evaluation of the quality of multi-agent systems. Autonomous reactive agents, with varying degrees of supervision, come into play in industrial robotics, banking, energy management, health, defense, and mobility (vehicles without driver). They must be able to deliberate in a largely autonomous way and to act consequently. In the last years, the notion of adjustable autonomy has been introduced to temper autonomy. It prospects the development of autonomous AI systems which are able to understand the limits of their autonomy, that is, to understand when certain deliberation that they could take are questionable, unsafe, or unethical, asking for human supervision when appropriate. In fact, autonomous AI systems are still more a hope than a reality. Their goal is established and suitably encoded by the designer; their task is to find the better way to achieve it on the basis of available knowledge. In most cases, they enjoy little autonomy. Their behavior is often very sophisticated, but also extremely constrained, as it is largely programmed in advance. This severely limits the ability of AI systems to operate in contexts other than those envisaged, reducing the ways and areas of use.

A third point concerns the relationship between creativity and *intentionality*. Creativity can be viewed as the ability to look at a given subject from a perspective which is radically different from consolidated ones or to grasp new, original relationships among existing ideas or things. Such an ability does not emerge from nowhere, but it is the result of a long sedimentation and re-elaboration of knowledge and reflections. The search for a solution that cannot be found by following the traditional paths is at the origin of the creative generation.³² Thus, the fundamental role of intentionality in the creative process is evident. An interesting example is that of automated narrative crafting, which

³² There are surely inventions by chance and by error, but they are definitely not the most common and significant paths to creativity. (Almost) nothing comes out of the blue. Creative contributions, including the most innovative ones, require a deep understanding of the state of the art of the field as well as of its methods and techniques that only make it possible to find an original way to look at and to address a problem and/or to establish unexpected connections among apparently far away elements. Error and chance may come into play in this process, and possibly help in making progress.

is an emerging research topic in the field of data narration.³³ Recent work on that subject points out the importance of intentional aspects by introducing an intentional layer that explicitly support the intentions of the data narrator. Such a layer models those activities that help in formulating goals and questions, drawing messages from findings, and validating messages. Such a close relationship between creativity and intentionality allows us to conclude that creativity, as intentionality, in machines is always of derivative nature. Intentionality expresses itself in the construction of artifacts, of which it constitutes, in a certain sense, the final cause. Subsequently, and in a different way, intentionality is also active in artifacts' exploitation and it is at the origin of the intelligibility of machines. This form of intentionality of machines is called 'derived intentionality' by Searle.³⁴ The same can be said of creativity: the creativity of a machine is the creativity of the human beings that conceive, design, program, and implement it.

In view of the above, let us go back to our original question: how and where can we speak of creativity with reference to an AI system? In spite of its bad reputation, that may induce one to think of it as a blind, artificial procedure that ignores the richness and variety of reality, the design of an *algorithm* (a program) to solve a given (algorithmic) problem is the most creative and demanding activity of a computer scientist. It is not by chance that the most famous series of books devoted to algorithms (a sort of encyclopedia of algorithms) by Knuth is titled: *The Art of Computer Programming*.³⁵ Hence, we can safely claim that creativity is at the heart of computer science (and of computer scientists). This is not to say that craft and engineering abilities are not important. The design of algorithms requires art and craft, and the best software engineering methodologies developed over the years help a lot in writing algorithms. However, there is no way (in general) to synthesize in an automatic way an algorithm from a high level specification of its expected behavior, as the synthesis problem in its full generality is undecidable, and thus in many cases the creativity of the designer plays a fundamental role.

The general picture of the theory of algorithms is well established, starting from the very basic notion of algorithmic problem. An algorithmic problem is a problem for which, at least in principle, it is possible to think of an algorithmic solution. Any algorithmic problem can be turned into a decision problem, that is, a problem with a yes/no answer. Turing has shown that there

³³ F. EL OUTA, P. MARCEL, V. PERALTA, R. DA SILVA, M. CHAGNOUX, P. VASSILIADIS, *Data Narrative Crafting via a Comprehensive and Well-Founded Process*, in S. CHIUSANO, T. CERQUITELLI, R. WREMBEL (eds.), *Advances in Databases and Information Systems*, Proceedings of the 26th European Conference, Springer, Cham 2022, pp. 347-360.

³⁴ J. R. SEARLE, *Mind: A Brief Introduction*, Oxford University Press, Oxford 2004.

³⁵ D. KNUTH, *The Art of Computer Programming. Volume 1: Fundamental Algorithms*, Addison-Wesley, Boston MA 1997.

exist meaningful algorithmic problems, like, for instance, the halting problem for Turing machines, that are undecidable, that is, such that there exists no algorithm that solves them. The distinction between decidable and undecidable problems, however, is not the only relevant one. The distinction between tractable and intractable problems is at least as important as that between decidable and undecidable ones. We distinguish between good algorithms, that require polynomial time, or less, for their execution, and bad algorithms, that require exponential time. A problem for which there exists a good solution algorithm is called a tractable problem; a problem that admits only bad algorithmic solutions is said intractable.³⁶

While algorithms play a fundamental role in model-based symbolic AI, research in data-driven AI may seem to follow a different path that has its roots in pattern recognition and data classification. This is not actually the case, as algorithms are at the basis of data-driven AI, even though they are expressed in a different way. Even more significantly, there is widespread recognition of the opportunity to bring together model-based and data-driven AI. Data-driven AI contributes effectiveness, model-based AI helps in interpreting machine learning models, making them understandable and trustable. In some cases, white-box machine learning algorithms can be devised that achieve comparable performance in terms of accuracy and turn out to be interpretable. Moreover, data-driven solutions may sometimes evolve into model-based ones once the underlying characteristics of the addressed problem have been understood and encoded by a (formal) model.

We would like to conclude our contribution by pointing out that the integration of model-driven and data-driven AI is the main goal of *Explainable AI* (XAI for short). XAI is an emerging area of AI which aims at developing trustworthy AI systems, that is, AI systems that are able to account for the choices they made in such a way that they can be understood by human beings.³⁷ The nature and the complexity of many machine learning solutions is such that even the designer is not able to explain why and how a certain decision has been taken by the system. We refer to these solutions as black-box ones. XAI asks for the ability of the system to explain its decisions and actions to human users. The relevance and necessity of explanation have been disputed by some AI researchers, who claim that it is difficult to achieve and, in many cases, unnecessary. A more subtle issue concerns the dependency from the context of the notion of explainability. It cannot be given in absolute terms, but only on the basis of knowledge, expectations, and tasks of the system us-

³⁶ For a more detailed illustration of these topics, we refer the reader to A. MONTANARI, *Per un vocabolario filosofico dell'informatica*, cit.

³⁷ D. GUNNING, M. STEFIK, J. CHOI, T. MILLER, S. STUMPE, G.-Z. YANG, *XAI – Explainable Artificial Intelligence*, «Science Robotics», 4/37 (2019), DOI: 10.1126/scirobotics.aay7120.

er. As a consequence, what can be defined as a satisfactory explanation of the behavior (decisions and actions) of an AI system may vary from one scenario to another one. Despite the difficulties that XAI has to face, there are many domains where explanations are necessary to make the systems actually usable. Think of the domains of medicine, law, finance, defense, and mobility. In these domains, one may recognize a social right to explanation, not simply an individual right.³⁸

ACKNOWLEDGMENTS

First of all, I thank the anonymous reviewers for their feedback. I would also like to thank my colleagues Alberto Policriti, who carefully read a draft version of the contribution, and gave me a number of useful remarks and suggestions, and Andrea Brunello, for his comments on various aspects of data-driven AI. Last but not least, I would like to thank Ivan Colagè for his constant help and support.

ABSTRACT · Since its inception, artificial intelligence (AI) developed along two significantly different directions: symbolic AI and sub-symbolic AI. The former focuses on knowledge representation and automated reasoning methods and techniques, while the latter relies on machine / deep learning, neural networks, and genetic algorithms. In this paper, we give an account of the main characteristics and issues of the two areas of AI and explore their relationships. On the basis of such an analysis, we then discuss the issue of creativity in AI.

KEYWORDS · Symbolic Artificial Intelligence, Data-Driven Artificial Intelligence, Algorithm.

³⁸ A popular book that discusses the impact of these issues on social equality and democracy is C. O'NEIL, *Weapons of Math Destruction*, Crown, New York 2016.