Logical Induction, Machine Learning and Human Creativity

Jean-Gabriel Ganascia

(Contribution to *Switching Codes*, Bartscherer & Coover, eds.)

Version 1

CREATIVE MACHINES

The ability of machines to create new ideas has long been controversial. In the mid-19th century, Ada Byron-Lovelace argued that a computer was definitely not creative. As Lord Byron's daughter she was well-placed to speak about creativity, and she is considered to be the first software engineer as she wrote programs for Babbage's Analytical Engine, which is considered as the world's first computer. According to her "*The Analytical Engine has no pretensions whatever to originate any thing. It can do whatever we know how to order it to perform. It can follow analysis; but it has no power of anticipating any analytical relations or truths.*" (Taylor 1843, 722). One of the many commentators of this note, Alan Turing (Turing 1950), refuted the argument. For him, even Babbage's Analytical Engine, which was equivalent to a universal Turing Machine, could be unpredictable, and therefore creative. He mentioned his own experience with the first electronic computers that were built in the nineteen forties: even a very simple program, written with a few line of codes, could show surprising behaviors when executed on a finite state deterministic machine. It would have been possible – but tedious – for someone to replicate the machine activity by following step-by-step the execution of the program. Nevertheless, the general conduct of the machine is surprising. During the last fifty

years, various attempts have been made to confirm Turing's argument by producing creative machines. For instance, there are story telling machines (Turner 1992), automatic music composers (Cope 1991), painting machines (Cohen), etc.

However, the various attempts to build creative machines differ considerably (Buchanan 2001). The way a creative machine can be built, their capacity to originate interestingness and novelties, and what creation means for a machine are still being debated. For instance, the first attempts to build an automatic music composer varied from a stochastic approach built on probabilistic calculations, like the one developed by Iannis Xenakis, and a rule-based composition similar to what Pierre Boulez was familiar with. But none of them was completely satisfactory, since artistic creation has to be unpredictable, to be harmonious and to look familiar, all at the same time. If a machine systematically applies the rules of harmony, we get a totally consistent musical composition, but a musical composition rapidly becomes disappointing if it contains no elements of surprise. On the other hand, a randomly generated piece of work that does not obey any rules would have no real aesthetic value. Does such a thing as a systematic and general method somewhere between totally predetermined generation and purely random behavior exist? That is what I want to deal with here.

To be more precise, the questions I would like to try to answer here concern the status of creative processes and the possibility of reconstructing them on finite state automatons, i.e. on computers. Does a logic of creation exist? If so, how could this logic of creation help to build a creative machine? If we refuse to accept that it exists, it would mean that creation is somehow magic and that it is beyond any

systematic and rational, i.e. logical, analysis. I think that this is not the case and that imagination, inventiveness, ingenuity and original thinking may be decomposed into logical steps which can be simulated by mechanical processes on a computer. The aim of some work in artificial intelligence is to provide some empirical evidence to support this argument (Buchanan 2001; Boden 2004). This paper constitutes an attempt to justify it from a logical point of view and to analyze and define the logical status of creativity.

It is obvious that the logical inferences involved in creative abilities cannot be reduced to mere deduction, i.e. to inferences from the general to the particular, because deduction is by nature conservative, whereas creation corresponds to an increase in knowledge. Remember that the word "create" comes from the Latin verb *crescere* which means to grow; in other words, the output of a creative process has to contain more than was originally given as input. Inductive inference – i.e. inference from the particular to the general – could play a key role in creative activities. According to many classical views, induction is a generalization process, i.e. a colligation of facts within a general hypothesis that loses details of particulars, so all the specificities of the particular have to be removed during an inductive process. In other words, induction corresponds to a reduction in knowledge. Suggesting that induction plays a key role in creative activities would therefore be somewhat of a paradox since creativity is the ability to increase knowledge and not decrease it. However, as we shall see, symbolic machine learning brings out the importance of two other mechanisms that are not directly concerned with the loss of information, but with conceptual mapping and structural matching. Even if most of the classical philosophical theories of induction do not refer to conceptual mapping and structural

matching, practical induction, as it was theorized in Aristotelian natural science (Aristotle b), made use of those operations. More precisely, in the introduction to biology, Aristotle describes parts of animals as functional parts that can be mapped onto a functional representation of a living entity and matched with respect to this representation. For instance, wings, legs and fins can be matched with respect to locomotion. Lastly, psychologists have mentioned the role of conceptual mapping in creativity. Among them, Annette Karmiloff-Smith (Karmiloff-Smith, 1990) has shown that young children need an explicit, though not necessarily conscious, underlying structural representation in order to draw imaginary pictures, to tell stories or to play music.

This paper investigates the role of conceptual mapping and structural matching in inductive reasoning, in machine learning and in creativity and is divided into five parts. The first is an inventory of artificial intelligence theories of creativity; the second is an overview of the classical theories of inductive reasoning; the third presents structural induction in symbolic artificial intelligence, the fourth, inductive logic programming, which simulates induction by inverting deduction, and the last, the analogy between structural induction and Aristotelian induction as practiced in Aristotelian biology.

ARTIFICIAL CREATIVITY

Ever since the origin of Artificial Intelligence, there have been different attempts to design creative machines, each of which is grounded on a more or less

explicit model of creativity. These models can be classified into four paradigmatic categories:

- *Exploratory*: creativity requires abilities to explore and to find a path through a labyrinth.

- *Mathematical*: a creative proof makes explicit some elements that were absent – or only implicit – in the initial formulation.

- *Entropy-based*: since the goal is to compress information, i.e. to reduce entropy, interestingness is related to an unusual and surprising way of reducing information.

- *Compositional*: imagination combines chunks of remembering in a way that is relevant.

Each of these paradigmatic models has its own particularities that are briefly presented below.

Exploratory Model

One of the pioneers of Artificial Intelligence, Herbert Simon, pictured problem solving as the exploration of a labyrinth that can be automated on a computer (Simon 1957). In other words, a problem is viewed as an obstacle on our natural path. Note that the word problem comes from the Greek root probléma, which means obstacle: it is derived from *proballein*, literally "pro", forward, and "ballein", to throw, i.e. to throw forward. According to Herbert Simon, problem solvers have to find a path in an abstract description of the problem universe, which is called the

problem space, from an initial state to a state satisfying the desired goal. As expressed, problem solving is analogous to finding one's way in a large maze full of intricate passageways and blind alleys. Usually, the problem space, i.e. the labyrinth, is so huge that it cannot be described exhaustively, even by a powerful computer. Intelligent behavior, either natural or artificial, eliminates many paths and leads quickly to the goal, without investigating all possibilities. Artificial intelligence makes use of heuristics, i.e. tricks and rules encoding insight, to simulate such intelligent behaviors.

Herbert Simon did not restrict problem solving to games, mathematics or physics. He thought that all creative behaviors were primarily based on problem-solving abilities. One of his first papers, co-authored with Alan Newell and Clifford Shaw, entitled "The Process of Creative Thinking" (Simon 1979, 144-174), was an attempt to assimilate creativity to problem solving. During the rest of his career, Herbert Simon tried to developed creative machines with automatic problem solvers; he also encouraged much research in this direction and was the promoter of "Scientific Discovery". This area of research, whose main figures are Pat Langley, Jan Zytkow and Douglas Lenat (Davis & Lenat 1982), aims at a rational reconstruction of old scientific discoveries in a way which could be reproduced with a computer (Langley, *et al*. 1987).

Mathematical model of creativity

According to Herbert Simon and many other artificial intelligence scientists, creativity is the ability to move efficiently around a huge, intricate labyrinth.

Assimilating creativity to the exploration of conceptual space presupposes that the support of the search, i.e. the conceptual space, has already been given. However, in many cases, especially in mathematics, creative behavior consists not only in searching intelligently through a given conceptual space, but also in producing new ideas on which new conceptual spaces are built. Some artificial intelligent researchers, most of whom were mathematicians and logicians, defined creativity as the capacity to reformulate a problem in a new way that makes the solution appear easily. To illustrate this, let us take the "mutilated checkerboard" problem, which is described in detail by John McCarthy in (McCarthy 1999): given a "mutilated checkerboard", i.e. an 8×8 checkerboard from which the two corner squares of one diagonal have been removed (see figure), the problem is to cover this checkerboard with exactly thirty one dominos.
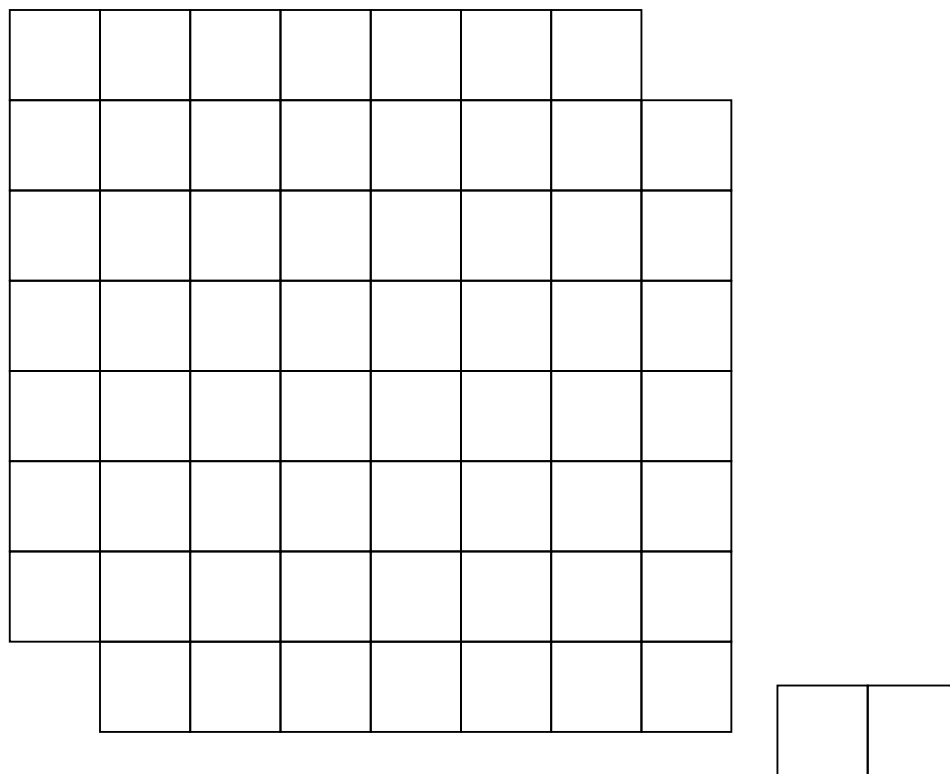

Figure 1: the "mutilated checkerboard" and a domino.

If we try to list all the possible combinations of the dominos on the "mutilated checkerboard", it would take a very long time, even with a computer. The classical solution would be to color in the checkerboard and the dominos in black and white. It immediately appears that, due to the mutilation, the number of white squares differs from the number of black ones, the reason being that the two removed squares are on the same diagonal and therefore have the same color. As a result, it is impossible to cover the "mutilated checkerboard" with dominos because, since each domino has one black half and one white half, it would mean that the total number of whites equals the total number of blacks on the checkerboard. This could be demonstrated elegantly once the color of the squares has been given. A creative mathematical demonstration would show up a new concept that was not present in the initial formulation of the problem, like the color in the resolution of the "mutilated checkerboard" problem. Would a machine be able to spontaneously reformulate such problems efficiently? That is the question researchers like Saul Amarel (Amarel 1986), Marvin Minsky, John McCarthy, Shmuel Winograd and others (McCarthy 1990) have tried to answer. For instance, in the case of the "mutilated checkerboard" problem, they imagined many solutions that could have been automated without coloring the checkerboard. However, each of these solutions depended on a creative reformulation that was not automatically derived from the initial formulation. In a word, even though the best among the artificial intelligence researchers tackled this problem, i.e. the automatic generation of new formulations, very few results were obtained in that domain.

The Compression Model

Over the last few years, some researchers who were originally physicists have been trying to unify artificial intelligence in the same way that physics has been (Hutter 2005), since it would be useful for them to have a general theory that interprets all the problems which artificial intelligence deals with. They have reduced automated decision-making and machine learning to information compression problems: in both cases, this means summarizing huge amounts of data using a small formula that can regenerate the data and extract their meaning. They conceive truth as an ideal world of optimal and irreducible simplicity. If we have huge sets of data, the natural goal is to simplify it as much as possible which, since simplicity is the rule, would anticipate new data flow. But if the challenge is to reduce information, once a set of information has been totally simplified it is of no great interest since there is no more hope of reducing it further. Let us consider, for instance, a sequence of 97,000 typographic characters. If it is composed of only one letter, e.g. 97,000 occurrences of "a", it can easily be simplified, but it is not exactly fascinating. If, however, those 97,000 characters have been randomly generated there is no hope of simplifying them and it is not really any more interesting than the first case. Let us now suppose that this sequence of 97,000 characters corresponds to William Shakespeare's "Macbeth". It cannot be reduced easily, but the reader continues to hope to be able to acquire a better understanding of the tragedy. More generally, intricate data motivates those who imagine possible reductions. As a result, interestingness can be viewed as the promise of a possible reduction.

According to this view, creativity can be envisaged in two ways: it may be either the ability to generate interesting data, with the hope of further reduction, or the simplification of existing data in an original and efficient manner that was not previously imagined. In both cases, the key concept on which interestingness, i.e. the hope of reduction, is based is the Kolmogorov complexity of a number or a sequence that is the size of the smallest computer program able to generate it. From a theoretical point of view, Kolmogorov complexity is a perfect heuristic for problem solving and, consequently, for the simulation of creative behavior. However, there is no means to compute it. As a consequence, it remains a theoretical concept that is of no help to anyone wanting to build an effective and efficient creative machine.

The Compositional Model

The last paradigmatic model of creativity is based on what people call imagination, which is a combination of remembered chunks. This model is anchored on memory identified with a faculty of recollection. Note that with this meaning, memory is not reducible to any storage device. To be more precise, memory is understood here in its psychological sense, i.e. with the ability to forget and to recall facts or anecdotes by evoking features associated with the remembering of those facts, as for instance where the taste of the *madeleine* (a cake) brings past episodes back to Marcel Proust. According to this view, creative activities are not magic; they are very common in everyday life where they help to solve new problems by retrieving old cases and then reusing and enriching past experiences.

The two basic mechanisms on which these memory-based creative activities are grounded are memory retrieval and case adaptation. The first, memory retrieval,

concerns the memory indexes that help to retrieve episodes experienced in the past. The second is a tweaking process that maps old episodes evoked by memory onto the present situation, and then adapts and combines these cases to help solve the problem being investigated. To illustrate this, let us take some mythological animals to be found in ancient bestiaries that are characteristic of this type of creativity: "sphinxes", "unicorns", Egyptian gods like Râ or Anubis (see figure 2), etc. All these figures involve combining memories which make new strange figures appear and creativity is viewed here as an ability to reuse remembrances of past experiences to make new combinations. This compositional model of creativity would seem to underpin the development of actual creative machines, as can be seen in the following examples. Two of my students, Geber Ramalho and Pierre-Yves Rolland, and I worked a few years ago on an artificial jazz player, which was able to compose a bass improvisation melodic line (Ramalho *et al.* 1999). The context defined by the audience, the chord grid and the other players, i.e. piano and drums, evoke abstract chunks of music in a memory simulated using knowledge-representation techniques. The chunks are then assembled with respect to constraints defined by both the chord grid and the harmonic rules of music.

In almost all the systems based on this compositional model, the tweaking mechanism relies on a structural matching between the old cases evoked by the context and the present situation. This structural matching itself makes use of a conceptual mapping to find components that play similar roles in the old and present cases. These basic mechanisms were also identified by psychologists (Boden 1995) who suggest that they play a key role in human creativity. And, as we shall see in the following, they are involved in inductive reasoning simulated using symbolic

machine-learning techniques. The rest of this paper focuses mainly on these inductive mechanisms.
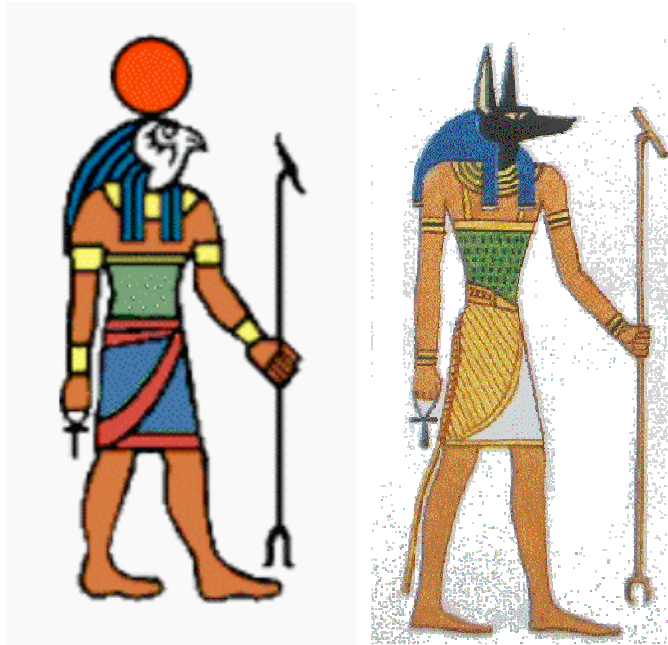

Figure 2: the two Egyptian gods Râ and Anubis.

INDUCTION

In the philosophical sense of the term, "induction" refers to "the shift from the particular to the general". Nevertheless, this definition seems to be rather restrictive with respect to the general meaning of induction, which is reasoning through experience. Philosophers have taken this on board: traditionally they distinguish two types of inductive reasoning, one which starts from the particular and moves to the general and the other which moves from the particular to the particular. Since the aim here is to understand the role that inductive – or, more generally, non deductive – inference plays in creativity, and since it has been shown that creativity is at least partially related to analogy, i.e. to reasoning from particular to particular, it is

impossible not to speak about the role of induction understood as reasoning from the particular to the particular when analyzing creative abilities.

However, three questions still need to be discussed. The first concerns the logical status of induction as an inference: is it certain or just conjectural? Does it correspond to some mechanical inference, like deduction, to a specific kind of syllogism that can be mechanized, or to an inversion of deduction, which would be non-deterministic? The second is related to the informational nature of induction: is it only a matter of information removal and compression, i.e. a reduction in the quantity of information? If so, it is difficult to imagine how induction could lead to the creation of information. The third question is about the evaluation of induction. Many people validate induced knowledge with respect to its predictive power. However, if an induction is correct only if it anticipates correctly, then good induction would leave no room for surprise, which has an important place in the ability to create.

Logical Status of Induction

Throughout the history of philosophy and logic, the status of induction has remained problematic. Since Aristotle, many philosophers including Sir Francis Bacon (1561-1626), John Stuart Mill (1806-1873), Jules Lachelier (1832-1918), Rudolf Carnap (1891-1970), Jean Nicod (1893-1931), Carl Gustav Hempel (1905-1997), Nelson Goodman (1906-1998), Jaakko Hintikka (1929-) etc. have tried to show that induction is an inference, i.e. a logical operation they have attempted to legitimate.

During the sixties, with E. Gold's "Identification to the Limit" learning paradigm (Gold 1967), and more recently with the "Probably Approximately Correct" learning (PAC learning) paradigm (Valiant 1984), or the statistical learning theory (Vapnik 1995), many researchers have attempted to define the theoretical limitations of learning machines. These formal theories try to clarify the mathematical characteristics of mechanical inductive algorithms in computational terms, i.e. in terms of inputs, outputs and spatial versus temporal algorithmic complexity. In a way, they may be viewed as theories of inductive inferences that are complementary to the inductive logics developed by Carnap, Hintika and others. Within these inductive inference theories, induction is viewed as an approximate inference the uncertainty of which has to continuously decrease with the number of observations. The mathematical properties of inductive machines – i.e. the number of required examples, the number of features, and the speed of convergence, etc. – of inductive inferences are well addressed, but the logical status of induction with respect to deduction is not really determined.

Remember the debates about the relative position of induction compared to other inferences, in particular to deduction. To simplify, let us just look at two important positions, the Aristotelian one, which considers induction as an inversion of deduction, and that of John Stuart Mill, for example, for whom induction is a kind of deductive inference.

Although induction probably originated before his time, Aristotle was most certainly one of the first to have spoken of it and to have given it philosophical status, even if its place in his work was not that important. More precisely, syllogism lies at

the heart of Aristotle's approach: the philosopher analyzed specific modes of reasoning such as refutation, abduction, *reductio ad absurdum*, *petitio principii*, in terms of syllogisms. It was in this context, which was the subject of Book Two of the *Prior Analytics*, that induction was considered (Aristotle a). Remember that a syllogistic inference contains a *Premise* made up of two propositions, the *Major* and the *Minor*, and a *Conclusion* made up of just one proposition. In valid syllogisms the *Majors* are usually general propositions, while the *Minors* and the *Conclusions* are particular propositions. Here is an illustration of such a syllogism:

All swans are white	**Major**
A, B, C, etc. are swans	**Minor**
A, B, C, etc. are white	**Conclusion**

Thus, for Aristotle induction moves from the *Conclusion* and the *Major*, which have been shown empirically to hold perfectly, to the *Minor*. To put it another way, starting with two propositions, *A, B, C, etc. are Swans* and *A, B, C, etc. are white*, induction enables us to infer the *Major* of the syllogism – i.e. the general proposition – namely *All swans are white*, which links the first two propositions that are two particular propositions. In other words, for Aristotle, inductive inference is an inversion of deduction, which could be summarized as "Induction = Deduction^{-1}".

Note that in some ways Aristotelian induction appears to be reasoning that is certain. However, in order to be certain, Aristotelian induction requires that the extension of the Major, i.e. all the possible instantiations of the induced knowledge, be covered exhaustively by the extension of the middle, i.e. by the observations,

which limits its use to finite cases. What would happen if the set of all the cases covered by a rule were infinite? Because of this, certain induction is an extreme figure of empirical reasoning that can never be fully realized in practice, since a new case could always appear that would invalidate the existing induction. For instance, the number of possible swans exceeds the number of observed swans and it may also happen that a black swan could exist which would invalidate the induction *All swans are white*.

In the 19th century, many philosophers were interested in induction: in France we find Pierre-Paul Royer-Collard, Victor Cousin and Jules Lachelier, and in England the philosopher who today is considered to be the greatest classical theoretician of induction of them all, John Stuart Mill. In his system of logic he presents induction as being the source of all knowledge.

Once induction has been clearly identified and distinguished from the other modes of reasoning such as *abstraction*, *description* or *colligation*, with which it is often associated, Mill defines it as a formal operation. To do this he uses Aristotle's definition but twists it so that induction becomes a syllogism in its own right and not a mode of reasoning. Remember that for Aristotle induction consisted in looking for one of the premises of a syllogism by starting from the other premise and the conclusion. Mill transformed this into a syllogism, but of a particular kind. Let Mill explain what he means.

"[...] every induction is a syllogism with the major premise suppressed; or (as I prefer expressing it) every induction may be thrown into the form of a syllogism, by supplying a major premise. If this be actually done, the principle which we are now

considering, that of the uniformity of the course of nature, will appear as the ultimate major premise of all inductions, and will, therefore, stand to all inductions in the relation in which, as has been shown at so much length, the major proposition of a syllogism always stands to the conclusion, not contributing at all to prove it, but being a necessary condition of its being proved [...]." (Mill 1865, p. 345).

What Mill does is to transform this syllogism so that the *Major* and the *Conclusion* change places. In this way the proposition that is inductively inferred becomes the *Conclusion* and the initial *Conclusion* replaces the original *Major*. For the resulting syllogism to remain valid, he adds a new *Major* that he derives from the principle of uniformity and which says that what is true for A, B, C, etc. is true for all swans, which gives:

What is true for A, B, C, etc. is true for all swans	**Major**
A, B, C, etc. are white	**Minor**
All swans are white	**Conclusion**

This syllogism thus enabled Mill to reduce induction to a conjectural deduction or, more precisely, to a certain deduction under a conjectural hypothesis, namely here the uniformity hypothesis which states that "What is true for A, B, C, etc is true for all swans".

Nevertheless, whatever the status of induction, a kind of mechanistic syllogism or an inversion of deduction, machines always have the ability to simulate it. But, in

each case, induction seems to be viewed as a generalization procedure, which leaves out the specificity of the particular, i.e. it leads to removing information.

Information Loss and Prediction

As we have seen, the common view of induction corresponds to a generalization. Either in the Aristotelian view, which assimilates induction to an inversion of deduction, or from the perspective introduced by Mill, induction always leads to a loss of information, since it is the colligation and the discovery of a common property of examples, and as a result reduces and/or compresses information. In other words, all the specificities of the particular have to be forgotten in the induced knowledge which is a common description of individuals. One of the consequences of this view is that induction is associated with a decrease in information. Though many empirical learning procedures fit this view, it is not at all the case with creativity, which is nothing if not an increase. Therefore, the first point concerns the alternative view of induction: are there any conceptions that do not limit it to a contraction and a reduction? We shall see, below, that artificial intelligence techniques implement inductive generalization procedures that do not reduce to a loss of information.

The other point concerns the evaluation of induced knowledge. Since induction is usually conjectural, the result is not certain and has to be confirmed using some other procedure. The main criterion that validates it is its predictive power. In other words, generalization is supposed to help anticipate the evolution of the world. In the case of natural science, such a view has shown to be very fruitful.

However, in the simulation of creative abilities, it appears too restrictive, since one of the criteria of creativity is the capacity to surprise. How would a machine be creative, if it could only anticipate change without introducing any new unexpected elements? One answer would be that the way the machine predicts the change could be quite new, which would correspond to an original theory. But this means that a criterion of novelty and originality has to be added and that this criterion cannot be reduced to the predictive power.

This last point concerns the new validation procedures that creative machines need, and which cannot be reduced to measuring the percentage of correct expectations derived from the induced knowledge. The so-called "Turing test" (Turing 1950) was an attempt to evaluate such machines and was built on the machine's ability to deceive men and women in the imitation game. Without going into more detail, it would seem, once again, that artificial intelligence techniques address creativity in an original manner.

STRUCTURAL INDUCTION

Symbolic versus Numeric Artificial Intelligence

It is common to regard traditional artificial intelligence as being restricted to symbolic, exact and deterministic approaches, whereas new artificial intelligence takes approximation and uncertainty into account using a numeric approach combining neural networks, belief networks, reinforcement learning and genetic algorithms, for instance. However, specialists in artificial intelligence do not all

identify with this way of looking at things. Knowledge representation is a crucial part of their work and the ontologies to which they sometimes refer are only put forward as hypotheses and models, with nothing definitive or rigid behind them. In fact, in the last few years nothing that has happened in artificial intelligence and in machine learning seems to have reduced the opposition between these different views, even if it is now common to combine symbolic and numeric approaches.

Moreover, this opposition has also persisted throughout the history of philosophy and has still not been resolved. Symbols predated numbers and continued after numbers appeared. Numbers were introduced into the theories of induction in the 18th century by Buffon and Reid and have remained, especially in the 20th century, during which the philosophical theories of induction all had recourse to numbers without forgetting symbols. This was true for the probabilistic theories of people like Carnap and Reichenbach and also for those, including Nicod and Hempel, whose work was an attempt to legitimate a logical approach to induction. This is also true in artificial intelligence where the numeric approaches to machine learning do not signify the definitive rejection of symbols.

This opposition between numeric sub-symbolic artificial intelligence and logic-oriented symbolic artificial intelligence will not be discussed here; the only point is to see what exactly is new about the inductive mechanisms used in artificial intelligence. As we shall see, it appears that symbolic artificial intelligence defines a new approach to induction whereas numeric machine learning relies on well-known and well-established philosophical principles that have already been clearly described by philosophers. More precisely, classical mechanisms, on which numeric machine-

learning techniques are based, do not offer anything new. For instance, the induction of association rules by detecting correlation among descriptors, the discrimination of positive and negative examples by finding the optimal separation, the generalization of propositional depiction by dropping one of the conjuncts and the introduction of numbers to quantify the degree of confirmation of an induced hypothesis largely predate artificial intelligence and machine learning.

This does not mean that the considerable amount of work which has been done in the last few years in numeric approaches to machine learning has not produced anything new. Today's research is more accurate and precise than ever, yet most of the elementary mechanisms which are commonly used in numeric machine learning are based on well-known pre-existing ones. The novelty lies mainly in the way these mechanisms are combined and applied. For instance, the notion of simplicity viewed as controlling the application of generalization operators is nothing other than a case of Occam's Razor principle.

Features and Structures

Knowledge representation is a key issue in artificial intelligence in general and, more specifically, in both numeric and symbolic machine learning. However, there is a big difference in the way objects are represented in numeric and symbolic machine learning: whereas representation used in numeric machine learning is restricted to sets of features, symbolic machine learning is able to deal with structured objects containing subparts that are related to each other by logical relationships. In terms of logic, this means that numeric machine-learning algorithms are trained on

examples described by conjunctions of propositions whereas symbolic machine-learning algorithms authorize first order predicate logic. However, this increase in knowledge representation has a mathematical counterpart: in the case of propositional logic, the generalization-space algebraic structure is a lattice while in the case of first order predicate logic it is far more complex.

To illustrate this point, let us take three examples given in propositional logic. A white rose: e_1 = white $\wedge$ rose; a yellow narcissus: e_2 = yellow $\wedge$ narcissus; a white narcissus: e_3 = white $\wedge$ narcissus. The generalization of two conjunctive descriptions retains all propositional descriptors that are common to both. Therefore, the generalization of e_2 and e_3 is "narcissus"; the generalization of e_1 and e_3 is "white"; the generalization of e_1 and e_2 is empty. More generally, there always exists one maximal common generalization for each pair of examples and the lattice structure of the generalization space is based on this property.

Let us now consider two flower baskets represented as structured examples:

flower_basket$_1$ = white(a)$\wedge$rose(a)$\wedge$yellow(b)$\wedge$narcissus(b)$\wedge$on_top(a, b)

flower_basket$_2$ = white(c)$\wedge$narcissus(c) $\wedge$yellow(d)$\wedge$rose(d)$\wedge$on_top(c, d)

Because these two examples are structured, they refer to predicates, i.e. to functions and not to propositions. As a consequence, descriptions are built on terms which are all different; for instance "white(a)" is not equal to "white(c)" even if they designate a similar property. Therefore, it is not possible to define generalization as an intersection of common conjuncts belonging to descriptions; it is necessary to consider the mappings of their subparts. For instance, one may consider that "a"

maps onto "c" and "b" onto "d", or that "a" maps onto "c" and "b" onto "c", etc. Since there are two objects in flower_basket$_1$ and two in flower_basket$_2$ the total number of map possibilities is 2×2=4. Here are the four maximal generalizations corresponding to the four possible matchings:

white(X)∧yellow(Y)∧on_top(X, Y)

white(X)∧rose(Y)

yellow(X)∧narcissus(Y)

rose(X)∧narcissus(Y)

For more than 35 years now, researchers have been trying to clearly define the generalization of structured examples so as to be able to establish the logical foundations of inductive machine learning. Different formalisms have been developed by Plotkin, Vere, Michalski, Kodratoff and Ganascia, Muggleton, etc. The next section presents the most wide-spread one today, the Inductive Logic Programming (ILP) formalism.

INDUCTIVE LOGIC PROGRAMMING

The ILP formalism is directly related to logic programming and to automatic theorem-proving techniques that use the so-called resolution rule. Therefore, to give an account of ILP, we first need to remember what the resolution rule, which serves as the basis for deduction procedures, is. It will then be possible show how, within this framework, induction may be assimilated to an inversion of deduction, as in

Aristotelian induction, which naturally leads, since deduction is based on the resolution rule, to an inversion of the resolution rule.

Automatic Deduction using the Resolution Rule

Many automatic proof procedures are based on the resolution rule. In particular it is the basis for most logic programming languages developed in the seventies or in the eighties, PROLOG, for instance. The key operation that serves as the foundation for the resolution rule is unification. Here are some definitions of the fundamental notions.

Definition: Two terms t_1 and t_2 are said to be *unifiable* if there exists a substitution σ of the variables of t_1 and t_2 that makes them equal, i.e. such that $t_1\sigma = t_2\sigma$. σ is called a *unifier* of t_1 and t_2.

Theorem: if terms t_1 and t_2 are unifiable then there exists a *most general unifier* (mgu) σ of those two terms, i.e. a unifier σ such that for all unifiers η there exists a substitution θ with $\eta = \sigma\theta$.

Once unification has been defined, it is possible to define the resolution rule.

Definition: let us consider two clauses, C_1 and C_2, i.e. two disjunctions of literals, and let us suppose that L_1 belongs to C_1, i.e. that L_1 is one of the disjuncts of C_1 and that L_2 belongs to C_2.

If L_1 and L_2 (or L_1 and L_2) are unifiable with the most general unifier σ, then the resolvent C of C_1 and C_2 by L_1 and L_2 – noted res(C_1, C_2; L_1, L_2) – is defined by: C = res(C_1, C_2; L_1, L_2) = $\{C_1\sigma - L_1\sigma\} \cup \{C_2\sigma - L_2\sigma\}$

Notation: S being a set of clauses, the derivation of the clause C from the application of the resolution rule to clauses of S is noted $S \vdash_{res} C$

Theorem: S being a set of clauses, S is unsatisfiable if and only if $S \vdash_{res} \varnothing$ where $\varnothing$ corresponds to the empty clause, i.e. to the falsity.

Corollary: S being a set of clauses and C a clause, $S \vdash C$ if and only if $S \cup C \vdash_{res} \varnothing$

Inversion of Resolution

Following the Aristotelian conception in which induction is an inversion of deduction, inductive inference is formally defined as an inversion of deductive inference. Since the resolution rule plays a key role in deductive inference, researchers have tried to inverse resolution. The first attempts were made in the early seventies by G. Plotkin (Plotkin 1970) who inversed unification. Twenty years later, in the nineties, S. Muggleton proposed inversing resolution. The induction is formally defined as follows.

Given:

- A set of observations o that are supposed to be expressed in the form of a set of clauses L_o.

- Background knowledge expressed as a theory θ that does not explain the observations, i.e. such that $\forall o \in L_o \, \neg[\theta \vdash o]$

The induction consists in finding a hypothesis α which explains all the observations o belonging to L_o, i.e. such that $\forall o \in L_o \; \alpha \wedge \theta \vdash o$ (see (Muggleton 1992)).

Because the resolution rule is complete, it means that $\forall o \in L_o \; \alpha \wedge \theta \vdash_{res} o$. Since L_o and θ are initially given, this is equivalent to $\alpha \wedge L_o \vdash_{res}^{-1} \theta$ where $\vdash_{res}^{-1}$ designates the inversion of the resolution rule.

Without going into detail, inverting the resolution rule is a non-determinate operation that requires inverting substitutions, i.e. associating the same variable to different constants that are supposed to be matched. Therefore, the number of possible inductions is directly related to the number of possible matches, which may be huge.

The recent advances in relational learning and Inductive Logic Programming attempt to limit the number of mappings by introducing strong formal constraints. The notions of determinism, of ij-determinism, of k-locality, of 1-determinacy and of structured clauses are examples of such restrictions whenever they are formalized in the Inductive Logic Programming framework. Other restrictions have been expressed in other frameworks, for instance the number of conjuncts in rules, or the notion of "morion".

Some of these constraints that restrict the number of mappings correspond to syntactical limitations of the learned clauses; other constraints refer to outside knowledge that authorizes some mappings between objects belonging to compatible

concepts and prohibits others. In other words, these constraints correspond to a conceptual mapping that guides the structural matching.

As explained above, these two operations, structural matching and conceptual mapping, have been proved by many psychologists to be central to our creative abilities. This paper has shown that they also play a key role in the mechanization of inductive inference and, more precisely, of structural induction, as developed in artificial intelligence. Therefore, structural induction, relational learning and inductive logic programming emphasize some aspects of inductive inference that had been largely ignored before and that correspond to a general operation currently achieved in most practical inductive inferences, and especially in inferences that play a key role in our creative abilities.

BACK TO ARISTOTELIAN BIOLOGY

Despite its relative novelty in the theory of induction, the notion of mapping had already been investigated by ancient philosophers. Aristotle, for instance, in the introduction to his Zoology entitled "Parts of Animals" (Aristotle b), established a correspondence between organs that are all involved in the same biological function.

To take just one example of a biological function, that of locomotion, Aristotle maps birds' wings, fishes' fins and mammals' legs, since wings, fins and legs are all involved in locomotion. Aristotle argues that this matching or mapping between functional parts helps zoologists to reduce the effort required when trying to understand the organization of unknown animals, by reusing part of the work that has already been done. Thus, zoologists who know how biological functions such as

locomotion, perception or reproduction are performed for various classes of animals will be able to classify new animals by observing their similarity to known ones, and to understand their organization without having to investigate them fully.

However, even though Aristotle recognized the role of mapping as being central to zoology, he never related it to logic. More precisely, Aristotle's theory of induction which was presented in his logic (Aristotle a), is not at all related to matching, but to the inversion of a deductive syllogism. In other words, it appears that the inductive inference which is used in Aristotle's natural science refers to matching among subparts of objects, while the inductive logic does not.

Curiously, the situation seems to be quite similar today in artificial intelligence, since many machine-learning techniques based on an inductive process do not refer to mapping, which is seen as being beyond the scope of the domain. It is the case for neural networks, belief networks, reinforcement learning, genetic algorithms, etc. On the other hand, both research into structural matching operations and recent advances in Inductive Logic Programming show that matching is a crucial issue and that strong constraints are required to limit the number of possible mappings. It also happens to be true that the solution required to decrease the number of mappings is similar to the Aristotelian solution, i.e. to provide some prior knowledge about the function of each part or subpart of a scene and to restrict matching to parts that realize the same function. Lastly, it has been proved that mapping and matching play a key role in many creative activities, since imagination may be viewed as an ability to match and to recombine old memorized structures with respect to some conceptual mapping.

In conclusion, artificial intelligence leads us to revisit classical theories of induction and creativity in a new way which has not be theorized before, even if it had long been anticipated in the empirical practice of induction.

CONCLUSION

This paper refers to many fields in the sciences and the humanities that have been mapped, matched, switched around and brought together: not only work in logic, artificial intelligence, philosophy, psychology and musical composition, but also Aristotelian zoology which is the core of the notion of structural induction developed here. In a way, we could say that the codes have been switched, but a preliminary condition for such an operation to be viable is conceptual space mapping. I hope that this paper has provided the necessary preliminaries for such a mapping to be possible and acceptable.

REFERENCES

Amarel, S., Program synthesis as a theory formation task: Problem representations and solution methods. In R.S. Michalski, J.G. Carbonell, & T.M. Mitchell (Eds.),*Machine learning: An artificial intelligence approach* (Vol.II). Los Altos, CA: Morgan-Kaufmann, 1986.

Aristotle (a), *Prior Analytics*, Hackett, USA, 1989, Translated by Robin Smith

Aristotle (b), Parts of Animals, Movements of Animals, Progression of Animals, 1968, Havard University Press.

Boden M., The Creative Mind, (Weidenfeld/Abacus & Basic Books, 1990; 2nd edn. Routldge, 2004)

Boden M., Creativity and unpredictability, *Stanford Electronic Humanity Review*, 1995, Vol. 4, issue 2, Constructions of the Mind: Artificial Intelligence and the Humanities, http://www.stanford.edu/group/SHR/4-2/text/boden.html

Buchanan B., *Creativity at the Metalevel*, AAAI Presidential Address, AI Magazine, Fall 2001.

Cohen H., AARON, http://www.viewingspace.com/genetics_culture/pages_genetics_culture/gc_w05/cohen_h.htm

Cope D., *Computers and Musical Style*, Oxford University Press, Oxford, 1991

Corruble V., Ganascia J.-G., *Induction and the discovery of the causes of scurvy: a computational reconstruction. Artificial Intelligence Journal*, Special issue on scientific discovery, Elsevier Press, (91)2 (1997) pp. 205-223

Davis R., Lenat D., *AM: Discovery in Mathematics as Heuristic Search* in Knowledge-Based System in Artificial Intelligence, Part One, McGraw-Hill, New-York, 1982.

Feigenbaum E., Buchanan B., Ledergerg J., *On generality and problem Solving: a Case Study Using the DENDRAL Program* in Machine Intelligence 6, Edinburgh University Press, Edinburgh, 1971.

Ganascia J.-G., *Comments on Shank's talk on creativity*. R.S. Michalski, Y. Kodratoff Ed. Machine Learning 3, An Artificial Intelligence Approach, Morgan Kaufmann, 1990

Gold E., "Language Identification in the Limit", *Information and Control, 10,* pp. 447-474, 1967.

Hutter M., *Universal Artificial Intelligence: sequential Decisions Based on Algorithmic Probabilities,* Springer, 2005

Karmiloff-Smith A., "From Meta-Processes to Conscious Access: Evidence from Children's Metalinguistic and Repair Data", *Cognition*, 34 (1990), 57-83

Langley P., Simon H. A., Bradshaw G. L., and Zytkow J. M., Scientific Discovery: Computational Explorations of the Creative Processes. Cambridge, MA: MIT Press, 1987.

Mill J. S., 1865, "System of logic ratiocinative and inductive", Longman, Green and Co., , Vol. 1, Book III "Of Induction"

McCarthy J., *Creative Solutions to problems. Presented at AISB workshop* on AI and scientific creativity, April 1999, http://www-formal.stanford.edu/jmc/creative.html

Plotkin G., 1970, "A note on inductive generalization", in *Machine intelligence* 5, pp. 153-163, eds. Meltzer B. & Michie D., Edinburgh University Press.

Ramalho G., Rolland P.-Y., Ganascia J.-G., *An Artificially Intelligent Jazz Performer*. Journal of New Music Research, 1999

Schank R.C. (1986), *Explanation Patterns: Understanding Mechanically and Creatively. Hillsdale*, NJ: Erlbaum

Schmidhuber J., *Simple Algorithmic Principles of Discovery, Subjective Beauty, Selective Attention, Curiosity & Creativity*, Proceedings of the 10th Discovery Science International Conference, Vincent Corruble, Masayuki Takeda and Einoshin Suzuki (Eds.), pp. 26-38, 2007, LNAI 4755, Springer.

Simon H., Models of man. Mathematical Essays on Rational Human Behavior in a Social Setting, John Wiley & Sons, 1957

Simon H., Models of thought, New Haven and London Yale University Press, 1979.

Taylor R., *Scientific Memoirs, Selections from The Transactions of Foreign Academies and Learned Societies and from Foreign Journals*, edited by Richard Taylor, F.S.A., Vol. III London, 1843, Article XXIX. Sketch of the Analytical Engine invented by Charles Babbage Esq. By L. F. Menabrea, of Turin, Officer of the Military Engineers. [From the Bibliothèque Universelle de Genève, No. 82 October 1842] (translation and notes by Ada Lovelace).

Turing A., "Computing Machinery and Intelligence", 1950, *Mind 49:* 433-460.

Turner S., "Minstrel: A Model of Story Telling and Creativity", Technical Note, UCLA-AI-17-92, Los Angeles, AI Laboratory UCLA, 1992.

Valiant L., 1984, "A theory of the learnable", Communications of the ACM 27, 1134-1142.

Vapnik V., 1995, *The Nature of Statistical Learning Theory*, Springer.