

Some Notes on Artificial Intelligence as a New Mathematical Tool

Angel Garrido

Faculty of Sciences, National University of Distance Education, Madrid, Spain
Paseo Senda del Rey, 9, 28040, Madrid, Spain
algbmv@telefonica.net

Abstract

Mathematics is a mere instance of First-Order Predicate Calculus. Therefore it belongs to applied Monotonic Logic. So, we have found the limitations of classical logic reasoning and the clear advantages of Fuzzy Logic and many other new interesting tools. We present here some of the most useful tools of this new field of Mathematics, so-called Artificial Intelligence.

Keywords: Computer Science; Artificial Intelligence; Game Theory; Graph Theory; Heuristics; Mathematics.

1. Introduction

Among the things that *Artificial Intelligence (AI)* needs in order to implement a representation are *Categories, Objects, Properties, Relationships* and so on. All of them are connected to Mathematics, also constituting very adequate and illustrative examples. For instance, by showing Fuzzy Sets together with the usual Crisp or Classical Sets, there results a particular case of the precedent. Or, by typical concepts and strategies that are often supported by Discrete Mathematics, as well as the more convenient use of Graph Theory tools in many fields [1].

The ideas about thinking machines [5], the working mechanism of the human brain, the possibility of replicating its behavior may be materialized if we are able to produce some computational structure similar to a neural system, to reproduce its synapses by creating Artificial Neural Networks... These are no Science Fiction fictional subjects. Rather, they are real objects of study, and they have been so for a long time. Recently, these topics have sprung up with renewed interest.

The fundamental purpose of AI would be to create an admissible model for human knowledge. Its subject is, therefore, "*pure form*". We attempt to emulate the way a human brain reasons [12, 14]. This should be reached in successive, approximating steps, but the attempts always proceed in this sense. Initially, these works on AI were over-idealizations of the real world. The fields were, therefore, "*formal worlds*". Such search procedures were into the Space of States, which contains the set of all states, or nodes, in the case of representation by graphs, that we can obtain when we apply all the available operators.

Early AI programs used the same basic algorithm. To achieve a certain goal (winning a game or proving a theorem), they proceeded step by step to it (by making a move or a deduction each time) as if searching through a maze, backtracking whenever they reached a dead end. This paradigm was called "*reasoning as search*".

AI began as the quest to create machines that could think for themselves, and perhaps someday out-think humans. Over the years, AI has evolved into a more pragmatic discipline.

AI uses different strategies to solve the complex practical problems [12, 15]. And intelligence itself is now known to be too complex to be described by a single theory. Instead, multiple theories characterize the subject, with distinct levels of abstraction [14].

At the lowest, Neural Networks, Genetic Algorithms and other forms aid in understanding Adaptation, Perception, Interaction, etc., with the physical world.

On a more abstract level, designers of Expert Systems, Natural Language..., reflect the role of processes in *Creating, Sustaining, and Transmitting knowledge*.

The techniques for solving problems, in *AI*, can be of two types:

- *Declarative*: they permit the description of the known aspects of the problem (it is the *Heuristic Treatment*), and
- *Procedural*: which itemize the necessary paths to reach the solution of the problem (it is the *Algorithmic Treatment*).

But to pose problems is equivalent to constructing their solutions.

This requires *an agent*, the system or program to execute; *a set of actions*, which allow us to reach such objectives; and *a procedure of election*, which allows us to decide between distinct paths towards reaching its solution.

The *Inference in Rule Based Systems (RBS)* consists of establishing the certainty of some typical statement, from the information into the Base of Facts (BF) and the Knowledge Base (KB).

We have two available methods, *going forward*, and *going backward concatenation*.

The Rules show advantages on Classical Logic, where the reasoning was monotonic, with inferences without contradiction with pre-existing established facts. And, in the RBS, we can delete or substitute facts from the Base of Facts, according to the new inferences. All are provisional and modifiable; this is what made the *Reasoning Non-Monotonic*.

Another question is this: if there exists more than one rule that can be applied at a given time, which one should be executed first? Such set of Rules constitutes, in each step, the Conflict Set (which would be dynamic). The underlying decision problem is called *Resolution of Conflicts, or Control of Reasoning*.

2. Searching strategies

There exist different strategies to select a Rule for each step: *Ordering of Rules, Control of Agendas, Criterion of Actuality, and Criterion of Specificity*.

The *Criterion of Specificity* leads to executing, first of all, the more specific Rules, i.e. those with more facts in its antecedent.

We also have *Mechanisms of Control in RBS*. So, by the *Mechanism of Refractority*, we forbid repeating the execution of an already used Rule, unless more information exists that would allow or recommend such a case; *Rule Sets* allow us to activate or also to neutralize Blocks of Rules; and *Meta-Rules*, or Rules which treat (or reason) about other Rules. Such Meta-Rules can collaborate in Control of Reasoning, with the change or assignation of priorities to different Rules, according to the evolution of circumstances.

Nets: Out of these, the most recent studies deal with Bayesian Networks. Before its apparition [12], the purpose was to obtain useful systems for medical diagnosis, by classical statistical techniques, such as the Bayes Rule.

A *Bayesian Net* is represented as a pair (G, D) , where G will be a directed, acyclic and connected graph, and D will be a probability distribution, associated with the corresponding random variables. Such distribution must verify the *Property of Directional Separation*, according to which the probability of a random variable does not depend on its non descendant nodes.

The Inference in Bayesian Networks consists in establishing on the Net, *for the known variables, their values, and for the unknown variables, their respective probabilities*.

The objective of Bayesian Networks in Medicine would be to find the probability of success of a certain diagnosis, given certain symptoms.

A classical and illustrative example of Bayesian Network will be the well-known Net so-called ASIA (see Figure 1):

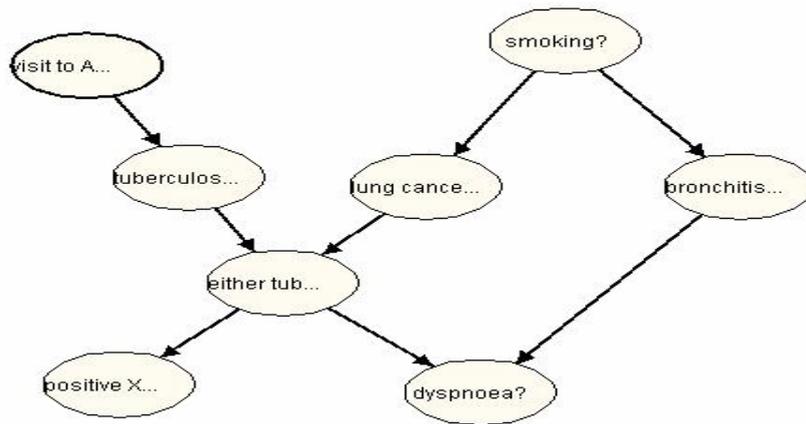


Figure 1. ASIA Bayesian Network

We need to work with the following *Hypotheses, Exclusivity, Exhaustivity* and *Conditional Independence*. According to the principle of *Exclusivity*, two different diagnoses cannot be right at the same time. With *Exhaustivity*, we suppose that all possible diagnoses are at hand. And by the CI (*Conditional Independence*), the discoveries found must be mutually independent to a given diagnosis. The usual problem with such hypotheses is their inadequacy to the real world. So, it is very convenient to introduce Bayesian Networks.

In the process, the correspondence between *State* and *Node of the graph*, and between *Arc* (edge into the graph) and *Operator* is clear.

By **Searching in extent**, we advance in the graph through levels. Or, we obtain the least cost solution, if it exists (see Figure 2). But, by **Depth Searching** we must expand one link each time, from the root – node (see Figure 3). If we reach a blind alley in the graph, we go back to the nearest node, and from this, we take one branch in the graph. It is usual to establish an exploration limit, or depth limit, fixing the maximal length from the root.

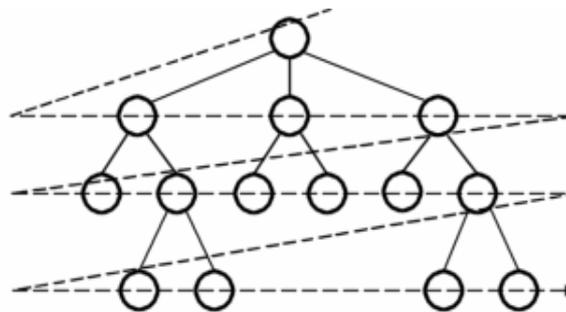


Figure 2. Search in extent

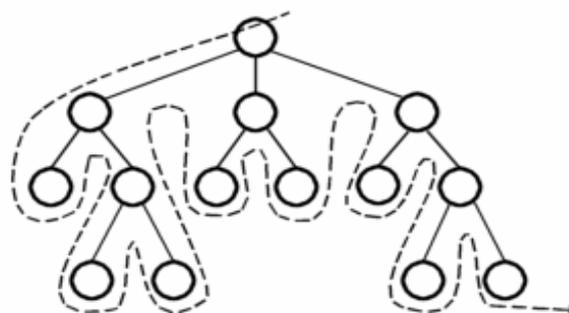


Figure 3. Search in depth

A “*knowledge engineer*” attempts to translate the most interesting information of an expert in a certain domain, by a computer program, in order to carry out some task. It is the purpose of *Expert Systems (ES)*. The first of them was *MYCIN* (1974) which diagnoses bacterial infections of the blood and suggested treatments.

Currently, most feasible kinds of *ES* try to put some information in one of a fixed set of categories using several sources of information. An example would be advising whether to accept a proposed credit card purchase. Information is available about the owner of the credit card, his record of payment and also about the item he is buying, the type of establishment from which he is buying it, etc. This approach is also very usual in medical prediction and diagnosis.

Heuristic Search, i.e. searching with knowledge of the domain [12]. Initially, it was hoped that every path could be explored by the computer. But it sounds too optimistic! Such an exploration would be unfeasible, because of the well-known phenomenon of “*combinatorial explosion*” of the branching, when we expand. Both, spatial and temporal complexity, can advise us against its realization. For this, we need to select the most promising trajectories. In this way, we cannot obtain the best solution (optimum), but a solution close to it, efficiently.

Now, we introduce a new mathematical tool, the so-called *heuristic evaluation function*, f . By such a function, we assign a value, $f(n)$, to each node n . So, such $f(n)$ gives us the estimation of the distance (unknown value) from the current node, n , until the final node, m .

Also, there are strategies for the treatment of *Searching problems with two adversaries*. In this case, the general purpose is to select the necessary steps to win the game. *Chess* is a good example and, in fact, it was its origin. But also *Go*, or other games [2]. We may assume alternative moves: the ideal for each move would be when the player knows his possibilities and realizes the most unfavorable move for its adversary [6, 7].

When we formulate a problem, we start from the statement, or the explanation of it, in natural language [13]. Fundamentally, its treatment is based on the “*level of knowledge*”, introduced by Newell, in 1981, as “*abstract level of interpretation of systems, in AI*”. Also basic is the “*Rationality Principle*”, according to which “*if a system has the knowledge according to which one of its actions leads to one of its goal then such action is carried out*”.

3. Some conclusions

All these problems, their methods of solution or approximation tools, may be adequately implemented in the class-room [3, 11]. For instance, implementing mathematical games, such as *Chess* or *Go*, would improve the logical capacities of students, in any school level:

In the elemental level, it may be an interesting stimulant to promote hidden potentialities [9, 10, 11].

In higher levels [8], it may contribute to strengthen the logical and deductive qualities of the researchers, by challenges such as the *Halting Problem*, or the *Travelling Salesman Problem (TSP)*, or even the still unanswered question of the 21st-century generation, *P versus NP*, with its multiple implications [4].

The aforementioned tools, currently very useful in Mathematics and AI, such as the Graph Theory tools, may be used to explain classical and very exciting questions.

4. References

- [1] Courant, R. (1996), *What Is Mathematics? An Elementary Approach to Ideas and Methods*. OUP.
- [2] Chen, Z. (2009), “Heuristic Reasoning on Graph and Game Complexity of Sudoku”. *The Smithsonian/NASA Astrophysics System*.
- [3] Guzmán, M. de (2001), *Para pensar mejor. Desarrollo de la creatividad a través de los procesos matemáticos*. Ediciones Pirámide, Madrid.
- [4] Kahneman, D.; Slovic, P., and Tversky (1982), *Judgment under Uncertainty: Heuristics and Biases*. CUP.

- [5] McCorduck, P. (2004), *Machines Who Think* (2nd ed.), Natick, MA: A. K. Peters, Ltd.
- [6] Mitchell, D. (1997): *Mathematical Origami*. Tarquin Publishing.
- [7] Mitchell, D. (2009): *Complete Origami*. Firefly Books.
- [8] Pólya, G. (2009): *How to Solve It: A New Aspect of Mathematical Method*. Ishi Press. First edition, 1945.
- [9] Pólya, G. (1990): *Mathematics and Plausible Reasoning*: in two volumes, Vol. I: *Induction and Analogy in Mathematics*. Vol. II: *Patterns of Plausible Inference*. Both in Princeton University Press.
- [10] Pólya, G. (1981): *Mathematical Discovery: On Understanding, Learning and Teaching Problem Solving*. Wiley.
- [11] Pólya, and Kirkpatrick (2009): *The Stanford Mathematics Problem Book: With Hints and Solutions*. Dover Mathematical Books.
- [12] Russell, S., and Norvig, P. (2003), *Artificial Intelligence: A Modern Approach* (2nd ed.), Prentice Hall.
- [13] Samuel, A. L. (1959), "Some studies in machine learning using the game of checkers", *IBM Journal of Research and Development* **3** (3): 210–219.
- [14] Searle, J. (1980), "Minds, Brains, and Programs", *Behavioral and Sciences* **3** (3): 417–457.
- [15] Turing A. (1950), "Computing Machinery and Intelligence", *Mind*, **LIX** (236): 433–460.