HAI as Human Augmented Intelligence: from Cognitive Biases to the Nature of Cognitive Technology

Jean-Baptiste Guignard, Ophir Paz, Kim Savaroche

HAI as Human Augmented Intelligence: from Cognitive Biases to the Nature of Cognitive Technology

Jean-Baptiste Guignard [1], Ophir Paz [2] and Kim Savaroche [2]

Address:

[1]IHEIE, Mines Paristech
[2]IMS, CNRS

Correspondence:

Jean-Baptiste Guignard
jbg@clayair.io

Introduction

Discussions of ethical issues in AI have traditionally focused on moral dilemmas such as the trolley problem. However, this focus on fictional dilemmas has little correspondence with the realities of AI architecture, or the actual challenges facing the development and future direction of AI. The aim of this paper is to reposition AI ethics within the framework of actual AI development and human decision-making. This inevitably involves demystifying some of the myths surrounding AI, and explaining and critiquing the underlying assumptions about human cognition from a philosophy of science perspective. The traditional framework of "AI ethics" first and foremost accepts that a given computational system, at some point, is "intelligent" — a term for a concept that is loosely defined — and by so doing implicitly accepts that it mimics, or has to be comparable with, human cognition and cognitive activity. Defenders of the strong AI stance (and the mass media) have dreamt of a human replacement, and opponents have been diminishing the power of what is considered to be a "lame" tool. Over the

last ten years, as machine learning has become increasingly popular, the debate has remained mostly binary, oscillating between fantasy and denial. However, from the perspective of the philosophy of science, this heated but superficial debate is yet another instance of the computational paradigm (reductionist, naturalist) opposing its externalist counterpart (embodied, distributed, enacted, etc.). Within the realm of those opposing arguments lies representationalism: the simple idea that something has to represent another and that the very symbol (word, variable, icon, schema, etc.) is an acceptable unity to build something that ultimately resembles, even in nature, what it tries to copy or mimic. Like language, like code, a unit would be a member of a set that results in something similar (the world, intelligence, attitude, high-level cognition). This paper argues, along with the externalist paradigm, that such a mimicking process is hardly a Lego construction; i.e., building blocks assembled into a pattern that seeks to resemble a real-world rabbit, car, tree, etc. The result is sometimes impressive but in no way is it realistic. Linguists have long argued that the concept of dog does not bite (the eternal problem of linguistic representation) while externalist proponents have long argued that a map only has representation value when it re-presents (i.e., presents again, from a new angle) the world through a perceptual prism. A re-presentation proves handy in conceptualizing, but nobody would ever argue that that a map represents real-world objects in fine detail. Those building blocks and their associated tools (papers, pencils, 2D traits, etc.) share no feature with the world, and we are not even addressing perception as a bias, because the construction of the world, in such paradigms, would appear to be constantly emerging; i.e., to be transductive. Accepting the idea that building blocks of a certain nature do not ultimately constitute a product or pattern that resembles what it aims to copy does not entail a rejection of an early computational cognitive science, where AI emerged in the 1950s and where it still stands paradigmatically. Chunks of code that classify do not result in human cognition, no matter how "unsupervised" they are, but when AI is considered as a dedicated tool with mechanical functions and a few affordances, it becomes a prosthesis that extends cognition the way a hammer does for bodily action: it may not be like — even less replace — human cognitive ability, but it certainly is a constitutive part of it. In a similar vein, glasses help or extend visual perception once we forget them and see through them. If AI is such a prosthetic and constitutive tool, then the cognitive process that we describe is, indeed, both computational and externalist, in such a way that we actually transcend the paradigmatic wars on the nature of cognition and, conjointly, the ethics

of AI. As a consequence, "responsibility" has to be understood in the light of human cognition overall. Who mobilizes such a tool? How? Who developed the system architecture so that it responded in the way it was programmed? This also applies to the non-fertile conceptual loops of utilitarianism, as discussed in "Myth 2" below. Firstly, however, we run through the full gamut of classical epistemological pros and cons that seem to constitute the nodal argument on which proponents and opponents of AI (full/weak) differ and confront, and we argue that symbols fail to represent or construct anything that is "alike". A summary of the argument follows:

1. Cognitive science emerged in a computer science fever.
2. AI emerged from a confusion between computing and cognition.
3. AI is therefore a powerful but an ill-named tool (classifier).
4. A tool (software or hammer) is a cognitive extension.
5. Such an extension, for AI, is code in nature.
6. Code is symbols with instructional power and representative purposes.
7. The world is to be mimicked (represented).
8. The world is perceived, perceptions are unstable, and code units differ from what they mimic.
9. Representation as fallacy is a historical truism.
10. Like language, code is symbolic and representational.
11. Like language, code does not represent the world, at best it re-presents the world, as a map re-presents a country.
12. If code as symbol constitutes AI, and AI is a tool that extends human cognition, we can legitimize both the computational and the externalist stances of cognitive science while rehabilitating AI as human-oriented.
13. Such a demonstration discards ethical discussions on AI per se and refocuses on human responsibility.

1 Coding and human interpretation

1.1 Traditional conception of code

The building blocks of a computer program are machine instructions. To create and maintain a piece of software, developers do not manipulate those machine instructions directly. They write code in a programming language that is-readable by humans and is then transformed into a list of instructions executable by a machine (Harris and Harris, 2012, p. 296). Code is usually pictured as a mechanical series of unambiguous instructions, but a program is often a source of crashes or unexpected operations in computer systems. Varying degrees of failure can still occur even after undergoing thorough testing, and are caused by wrong instructions given by the programmer, flawed testing, or malfunctioning hardware.

After decades of software engineering, coding methodologies have evolved to avoid or bypass those failures. From the V Model (splitting every development task from specification to delivery) to Scrum (putting the emphasis on communication and shared goals), the software industry has managed to deliver more elaborate programs while maintaining an acceptable quality (Beck, 2004; Martin, 2008). However, even with experienced programmers, functioning hardware and quality assurance processes, breakdowns still occur.

This can be illustrated by the failure of Ariane 5 space rocket in 1996, which exploded at launch because of a software bug (Lions, 1996). In fact, a code module from Ariane 4 — whad been heavily tested — was reused for the vehicle. During the launch process, this module received numeric values higher than expected and stored them in an undersized memory space, causing a chain of miscalculations. Technically, the code was written by skilled developers, tested through a structured process, and run on hardware that worked as expected.

1.2 Coding as semiotics

The issue is not competence, but misconceptions of a system's inner functioning, based on subjective perceptions of how some underlying operations are executed. The instructions in modern computer architectures are so complex that their combination cannot be comprehended by the developers working on a software program. Consequently, even experienced developers have to make assumptions about what the system is actually doing.

Framed as a perceptual question, the traditional conception presented above

implies that if transmission of the message is perfect (i.e., if the written code is not altered), then any fault is due either to the sender (mistakes by the developer) or to the receiver (a flawed computer). From this perspective, code is a message as conceptualized in the Sender-Message-Channel-Receiver model of communication developed by Shannon (Shannon and Weaver, 1948). This was originally intended to model communication between electronic devices but was later applied to human communication (and extensively criticized in this context). To address the case of the Ariane 5 failure, such a model suggests that the responsibility for failure lies with the programmer who should have been aware of every single operation that their code produced, from software to hardware. Although it seems an easy first step towards debunking the very question of AI ethics, this premise is arguably false in a complex system such as a space rocket or any modern computer architecture.

In the case of Ariane 5, programmers expected an instruction of the form `a = b` to simply copy the value of `b` into the memory allocated for the variable `a`. As these variables did not occupy the same amount of space in memory, the `=` operator induced a conversion process, resulting in a truncated binary representation of `b` that was copied to `a`. Initiating a chain reaction that caused the rocket to explode, this simple example highlights how the most elementary code instruction has a complex implementation that is always subject to human interpretation.

From this point of view, code is not simply a list of unambiguous items communicated between a developer and a machine, but rather a complex construction based on human interpretations of a system's behavior. The developer writes a set of instructions expecting a specific behavior. When the results match expectations, the developer associates the instructions with the system's behavior without the need to know about the underlying operations that they induce. This set of instructions can be refactored (Fowler, 2002), extracted into a function and given a name to make it easily reusable. Once refactored, another developer can use this function without knowing about its implementation. Relying on his/her own understanding, the latter adds another layer of interpretation.

Here, perception means being actively engaged in understanding how to interact with the environment, as argued by Noë:"Perception is something we *do*" (Noë, 2004). This requires the developer to focus on ways to grasp the system, to reach *affordances* (Gibson, 1979). This can only be achieved by extracting patterns from the code rather than by performing a detailed analysis of every individual component.

1.3 Negotiation and crystallization

In this way, each developer projects a specific meaning onto the code, depending on the context when he/she needs to grasp it. As a team of programmers have to work together, this fragmentation of understanding must be limited. In a complex coding process, a shared conception of the operations must exist to enable cooperative development of the software. This social understanding of the code has to be negotiated by the members of the team.

This is usually achieved by using "good practices" such as code reviewing and/or pair or even mob programming. Code reviewing requires a developer to explain the code that he/she has written to their teammates, and refactoring it together if needed. Pair/mob programming is a way of writing code that involves multiple members of the team at the same time, refactoring in real-time and negotiating meaning accordingly. Through code refactoring, reuse and negotiation, situated interpretations of repeated constructions are crystallized. This process produces a semiotic system surrounding the project, defined by a set of constructions associated with their negotiated meaning. The names given to the elements of the system during development are used to discuss the functioning of the software, from evolution of the requirements to technical documentation and eventually marketing material.

The crystallization of a semiotic system facilitates communication by simplifying complexity through a process of abstraction, where communication is understood not as preset for comprehension but as a continuously emerging and negotiated process. On the other hand, this hidden complexity is possibly where failure takes place. The negotiation and crystallization process results in making concessions, agreeing on putting away and potentially forgetting some edge-case operations. When a whole module was reused in the Ariane 5 project, interpretations of its behavior were already highly entrenched and unquestioned among the team. Its robustness (or instability) in a new context had not been anticipated.

Code, then, is symbolic in nature, understood as a constitutive part of a semiotic system, the items of which are interdependent and meaningful in relation to all the other components of such a sign system. As the `=` operator means attributing a value in a coding context or a red flag means "Do not bathe" in a real-world water context, a variable (or predicate) is a constitutive part of a semiotic system, just like words have been for decades in the history of linguistics or the epistemology of cognitive science. Like coding, like "languaging": coding

is the activity that consists in resorting to code units to construct meanings; languaging is the activity that negotiates meaning in real time. Both question the nature of representation (code to app to world, and word to concept to world) and the traditional debunking of AI legitimacy. Dreyfus (72, 84, 11) and Searle (80) have continually emphasized the limitations of that very symbolic and representational nature.

2 Representations

One might argue that representation is first and foremost a question of language, that is natural language and languaging. Signs pair with categorial pieces of reality and that referential status is moving along with talk and real-time meaning negotiation. We'll argue later that it has to be the same for code and coding at runtime - a command line (more or less) directly operates on hardware, and that is a concrete and/or natural reference to the world. That's a first excursion out of the expected disciplinary range of Computer Science and Engineering. But Michael Dummett remarked that the linguistic turn originated in the Fregean "extrusion of thoughts from the mind" (Dummett 1995, chap.4), and the entire "representation problem" - common to language and code - now extends to the nature of Mind and its locus. Questioning Representation is questioning the sublogic that leads to the nature of knowledge and "intelligence". AI, by being overtly representational and symbolic the way we understood signs (in linguistics) and the firing of neurons in the late 50s, inherits the same epistemological fallacies.

As the linguistic turn holds that an analysis and/or explanation of meaning and reference is the fundamental way to solve or dissolve philosophical issues, it appeared necessary to place language into some objective realm, away from the realm of ideas which was massively rejected. The objective dimension of language, for Frege, was instead to be found in the thoughts (*gedanken*) that sentences (*sätze*) express[1].

Most contemporary cognitive scientists consider that a scientific explanation of symbolic production is only viable if any semiotic system and meaning originate from our minds. In that vein, an adequate analysis of language or code necessarily involves a description of what happens in the cognitive system of the speaker(s).

[1]For Frege, the meaning of a statement (its sense) is the thought it expresses. Thoughts (*gedanken*) are not representations (*vorstellungen*), and grasping a thought does not entail having a representation in the mind.

Such a description is also involved in accounting for the semantics/pragmatics distinction, for sharing and understanding meaning, for indexicality, reference to and tracking of individuals through time, Fregean modes of presentation, etc. The purpose of this paper is not to investigate the various reasons that motivated the appearance of a cognitive turn in the study of semiotics. Rather, the critical focus is one of its points of origination: the belief that cognitive systems harbor mental representations and play a crucial role in the production and understanding of semiotic performances (code included). Our argument opposes cognitive representationalism (CR) in the course of which we discard the need for an AI ethics per se. Such an ethical stance implicitly accepts AI as a system that not only mimics real-world attitudes, but results in programs that are similar in nature. After insisting that code-is a-semiotic system, showing that representation is pure de-objectification means disqualifying code as a pure/faultless system that creates human-like entities — there is no such thing as an autonomously thinking machine.

(Guignard & Steiner, 2010) have opposed cognitive representationalism to linguistic representationalism, and that might be of use here: rejecting the latter does not amount to rejecting the former. Structural linguistics and the classical philosophy of language have been criticized repeatedly on the basis of their linguistic representationalism (see below). Even though contemporary linguists generally overcome LR criticisms, they are still in the grip of an even more precarious and non-explanatory representationalism, called "cognitive representationalism" (ibid.). Representation, we argue, remains one of the dogmas of contemporary Cognitive Science, a dogma that should be overcome, mainly because it has direct repercussion on the tenability of AI as representative or mimetic of human knowledge.

Back to linguistics then. That linguistic productions have referential powers is beyond doubt: they refer to something, and by telling us something about something they are meaningful; they have content. In an innocuous sense, nobody will then deny that linguistic productions have representational dimensions. However, are these representational dimensions exhausted by the referential properties[2] of language, or are they the consequences of deeper representational properties of language, such as by being a representation of reality or an externalization of mental activities? LR and CR adopt the second position whereas we argue that the first is the case.

[2]Provided it is possible to explain reference and content in terms of use or inference (Brandom 1994), and not in terms of representation or mirroring.

2.1 Symbols

Recent research in cognitive linguistics easily dismisses linguistic representationalism. According to the LR view, language is a representation of reality. It is a medium; an intermediary domain between us (what we think, what we do) and the world. Language relates us to reality by representing it. Hence, linguistic productions (paradigmatically, statements and sentences) are full of content ("contentful") because they are representations of reality (Neale, 1999: 657). Their content is what they are supposed to mirror. The success of referring and meaning basically depend on the success of representing.

2.1.1 Truth conditionals and command lines

Sentences are the primary bearers of meaning. Defining the meaning of a sentence (paradigmatically, a declarative sentence) is a matter of determining under which conditions the sentence is true; i.e., identifying the truth conditions of the sentence. These truth conditions are a matter of compositionality and reference: they are a function of the meanings of the parts of the sentence and their syntactic articulation. The meaning of a symbolic constituent of the sentence is the real-world entity it is said to refer to (objects, properties, relations, etc.). Knowing the meaning of a sentence is knowing what its truth conditions are. According to this truth-conditional approach to meaning, there is a basic and clear difference between the literal meaning of a sentence (corresponding to its truth conditions), which can be established in a vacuum, and its use in some contexts where meaning goes beyond the literal meaning and is understood following some inference, or pragmatic enrichment of the literal meaning. This corresponds to the clear distinction between semantics (reference, meaning, truth) and pragmatics (force, implicature, indexicality, etc.).

2.1.2 Objectivism

Objectivism notably[3] considers that:

1) There is a way in which the world is, independent of our conceptual schemes, ways of talking about it, theoretical engagements, or practical commitments. In other words, the basic "ontological furniture" of reality, made of objects, properties, relations, facts, and states of affairs, is intrinsic to it; it is not relative to the observer.

[3]See Putnam (1981), Lakoff and Johnson (1980: 186-188), Varela, Thompson, and Rosch (1991).

2) Symbolic truth is a matter of accuracy or correspondence to reality. Truth lies in an observer-independent correspondence relation between symbolic productions and the chunks of reality they depict. Facts make linguistic productions true. There is a way to contemplate this relation of correspondence between symbolic productions and non-symbolic reality by stepping outside of language (reality is immune to subjectification).

2.2 Criticizing representationalism

LR has been the object of much criticism from a wide range of perspectives. Non-truth conditional approaches to meaning, anti-realism, contextualism, pragmatics, etc., mark crucial steps in the possible demise of linguistic representationalism (or at least of some of its aspects). In that vein, the following points are fundamental and echo the works of major philosophers such as Dewey, Wittgenstein, Austin, Quine, Sellars, Goodman, Davidson, Dummett and Rorty.

1) The uses and purposes of symbolic performances are manifold. The representing, describing, or depicting functions of linguistic productions are contingent; they are peculiar applications of language. There are other kinds of utterances than declarative statements. The basic aim of performances, then, is not to copy something but to do something and/or make others do something, from understanding to acting. The representing purposes of language or any symbolic structure are even said to be always embedded in prior and wider contexts of use (acting, communicating, thinking). Symbolic performances (acts of meaning, referring) are necessarily mediated pragmatically by wider linguistic and normative practices (Brandom, 2008).

2) Truth cannot be a matter of correspondence between language and world, since there is no possibility to step outside of language in order to check whether a statement corresponds to some non-linguistic state of affairs. Epistemic awareness of reality is linguistic awareness: any consideration of what the world is, and how it makes our statements true, is caught within our conceptual schemes. Truth is more a matter of inferential integration (coherence, acceptability, "assertibility") or "disquotation" than of mirroring (correspondence).

3) Meaning cannot be thought of as representation or truth-conditionality. Indeed, the contexts of production and understanding of meaning are too diverse for meaning to be only a matter of representation or truth-conditionality. Sentences are not the primary bearers of content: only in the context of a speech act does a sentence express a determinate content (Récanati, 2004). The boundary between

semantics and pragmatics tends to be blurred. For Dummett's anti-realism, for instance, the meaning of a statement consists not in its truth conditions, but in its "assertibility" conditions; i.e., the publicly-accessible conditions under which a speaker would be justified to assert or to deny a statement. Meaning is not given by word-world relations; it is a construction or a role (inferential, functional).

4) Like language, code may be seen as symbolic and representational, or can expand those traditional boundaries. Writing, reading, and maintaining a piece of code may therefore be the result of an ongoing process. The developer externalizes and substantivizes a thought process, and a semiotic structure is then materialized as digital writing. As a way of interacting with a program's behavior, the act of writing becomes the very tool that helps the developer to update the program. Modifying his/her code impacts his/her tool and by extension his/her perception of the context.

3 Tools as cognitive extensions

3.1 Contextual perception

Code is subject to perception and its meaning is brought forth by the interactions between developers and code and their environment. As such, the same code can have different meanings for the developers as the context evolves. Even if the code itself is not changed, its perception does. For example, developers aware of the failure caused by a misinterpretation of the `=` operator (mentioned above) attach a different meaning (i.e., expect a different behavior) when the same code is encountered again (see Section 7: "Overcoming cognitive representationalism").

This implies that the perception of software behavior is always relative to a specific context. Therefore, a programmer's understanding of the system needs to evolve when the software is released in a production environment.

3.2 Tool mediated cognition

Code constructions become entrenched as they are being reviewed and re-used by a community of developers. Those constructions become the building blocks, the tools grasped by programmers to act upon a system's operations and re-build its meaning in a specific context.

A tool provides its operator with a new mode of action in the world. By

changing, but also constraining, how the operator can interact with the world, it also affects the perception of an agent. For instance, a pen lets someone create a drawing but at the same time sets a filter on the drawn subject. If the only pen available is black, the illustrator has to focus their attention on significant edges; otherwise, if a few colored pens are available, they are perceiving the subject through the filter of those colors.

In the same way, a collection of available instructions defines and constrains at the same time how a developer makes the software perform a task. This set of tools constitutes a framework for action and perception: a cognitive technology. The programmer perceives the software requirements as well as the existing system behavior through the operations that are available in the code.

3.3 Evolution of a cognitive technology

Any technology has to be maintained to stay relevant in a continuously evolving environment. These operations are not always about changing the behavior of a program. Refactoring is a common practice which entails reshuffling code without modifying what the software does. From an engineering point of view, the aim of refactoring is to make the code more comprehensible and easier to maintain (Fowler, 2002).

At a cognitive level, this operation implies a modification of the perception of the functioning of the system and how it can be acted upon. The cognitive technology evolves by creating new tools and adjusting how existing tools are grasped; i.e., by extracting new functions and renaming existing ones. From a given set of tools, it grows into a complex network of interdependent elements.

A cycle comes out of this process:

(1) Developers write code to create new software, and by doing so initiate the crystallization of the semiotic system / cognitive technology.

(2) The environment is changed by the existence of this new program as well as by other external factors.

(3) To ensure that the software is still relevant in the new context, developers have to maintain it. If the behavior of the program is still consistent with the environment, only incremental changes are needed. Otherwise, a refactoring step is needed to assimilate the new context into the cognitive technology used by the developer.

(4) The code is changed to create a new behavior: the cognitive technology evolves and a new version of the software is released, as in step (1).

3.4 Co-evolution of a technology and its environment

By repeating the cycle described above, software becomes part of the environment and its existence is constitutive of the evolution of co-existing elements. Starting as a new piece in an existing context, its arrival can modify its surroundings and eventually reshape the whole ecosystem.

This can be illustrated by the evolution of the mobile application Waze. It was first created as a GPS navigational program for smartphones. It integrates real-time machine learning to provide drivers with an optimized route according to data given by other users. The Waze algorithm often directs cars onto smaller roads or quiet neighborhoods, causing heavy traffic in places where it is not suitable. Nonetheless, the app is so popular that some city planners around the world change road organization and speed limits to trick the Waze algorithm and avoid attracting unwanted traffic. The code written by Waze developers mutated the environment it was created for.

This puts humans in the position of creating and maintaining tools but also of having a perceptual bias because of the tools. From this perspective, the context of software genesis— the first bias induced by the tools — will condition its early evolution severely. Additionally, this leads to the necessity of evaluating a system outside its developmental environment, in which it is the most influential.

4 Bio-inspiration and the modeling bias

Following the above, we want to argue that AI cannot mimic nature or human reasoning. More precisely, we will demonstrate how computer algorithms embody human bias. Three categories of bias will be suggested: pre-existing, technical and emergent. Pre-existing bias is rooted in social institutions, practices and attitudes. Technical bias arises from technical constraints or considerations. Emergent bias arises in a use context.

What is a biased computer system? In general, as defined by the Oxford English Dictionary, a bias is an "inclination or prejudice for or against one thing or person." For example, an employer can be biased by refusing to hire young people on the assumption that older candidates are more experienced. So, when an AI system is biased, it means that the algorithm systematically and unfairly discriminates against certain individuals or groups of individuals in favor of others. A system discriminates unfairly if it denies an opportunity or a good or if it assigns an undesirable outcome to an individual or group of individuals

on grounds that are unreasonable or inappropriate. For example, Amazon AI tools built in 2014 tended to not rate candidates for software developer jobs and other technical posts in a gender-neutral way (Vincent, 2018). The explanation was that Amazon's computer models were trained to vet applicants by observing patterns in resumes submitted to the company over a 10-year period. Most came from men, a reflection of male dominance across the tech industry. In this case, if the word "women" was in the resume, the AI system did not consider the candidate as a good one. Even if it was not intentional, the creators initially believed that using the former candidates' resumes was an objective way of training the system. The Amazon AI system was biased as it discriminated against woman. Ad systems do not hesitate to create profiles of users with the aim of displaying products that would satisfy wishes and needs linked to gender, age and race.

Pre-existing bias is rooted in social institutions, practices and attitudes. When computer systems embody (Clark, 2003) biases that exist independently, and usually prior to the creation of the system, then we say that the system embodies the pre-existing bias. Pre-existing biases are reflections of a culture, or a private or public organization. They can also reflect the personal biases of individuals who have significant input into the design of a system. Despite the best of intentions, this type of bias enters the AI system either through the explicit and conscious efforts of individuals or institutions (in choosing specific samples for the training process, for instance), or implicitly and unconsciously (choosing samples that illustrate a small set of situations). Let us imagine an expert system for loan applications. To detect applicants who are deemed to be too risky, the automated advisor would attribute penalty points to customers who live in low-income or high-crime neighborhoods, as indicated by their home addresses. The expert system embeds the biases of clients or designers who seek to avoid certain applicants on the basis of group stereotypes. The automated loan advisor's bias is pre-existing. Machine learning algorithms trained from human tagged data inadvertently learn to reflect the biases of the taggers (Diakopoulos, 2015).

Technical bias arises from the resolution of issues in the design of technology. Sources of technical bias include hardware, software and peripheral limitations. For example, formalizing a human construction originates from attempts to quantify the qualitative, discretize the continuous, and formalize the informal. An error in the design of a random number generator can cause particular numbers to be favored. An expert system offering legal advice would advise a

defendant on whether to plea bargain by assuming that the law can be understood by all in an unambiguous way and is not subject to human interpretation (Wittgenstein, 1929). Search engines tend to present results in alphabetic order or foreground sellers who have paid to display their product on the first page (Google, 2011)(Amazon, 2014). Flaws in data have implications for algorithms and are hidden in computer models and outputs (Romei and Ruggieri, 2014)(Barocas and Selbst, 2015).

Emergent bias arises in a context of actual use by real-world users. This bias appears after a design is completed. A version of a computer system that is frozen in time while the rest of the world evolves, bringing new social or cultural knowledge to the population of users, can lead to situations of misunderstanding. User interfaces are likely to be particularly prone to emergent bias because interfaces by design seek to reflect the capacities, character and habits of prospective users. Thus, a shift in the context of use may well create difficulties for a new set of users. Decision-support systems are unavoidably biased towards treatments included in their decision architecture. Although emergent bias is linked to the user, it can emerge unexpectedly from decisional rules developed by the algorithm, rather than by any 'hand-written' decision-making structure (Kamiran and Calders, 2010)(Hajian and Domingo-Ferrer, 2013).

A computer system's design is inevitably biased. It reflects the values of its designer and intended users. Development is not neutral, and there is no objectively correct choice at any given stage of development. There are many possible choices (Johnson, 2006) and, as a consequence, the values of the author of an algorithm, "wittingly or not, are frozen into the code, effectively institutionalizing those values" (Macnish, 2012).

The rise of machine-learning algorithms disseminated the possibility of changing a program's behavior without changing its code. In this architecture, the code itself is not an explicit representation of what the program is actually used for, but a generic process to transform input data into output categories through statistical computing. The actual representation of the software functioning is moved to the weight of the artificial neural connections, which are modeled at a sub-symbolic level. Does this evolution affect the essential nature of code and the responsibility of the programmer? The question of explicit versus sub-symbolic representations and the limits of such approaches remain to be addressed extensively.

5 Cognitive representations

What we label cognitive representations here, following (Guignard, Steiner, 2010), is the thesis according to which the referential relations between the symbolic productions of agents and the context of activities these agents refer to by means of these productions is necessarily mediated by sub-personal "mental' representations," both occurring in the producer (meaning-intentions, or tokens of concepts) and the receiver of these productions (understanding or interpreting conceptual acts). Linguistic productions have a referential dimension insofar as they are associated with mental processes trafficking in mental representations (the latter being intrinsically about their objects, not as linguistic representations, which have derived intentionality). A strong version[4] of CR can already be found in Locke's Essay on Human Understanding: language is an expression or representation of thought, itself being a representation of the world. Words and ideas are signs: ideas are primitive and intrinsic signs, whereas words are ultimately signs of signs since their function is to express and share (private) ideas.

What are mental representations for CR? They are classically defined as "contentful" intracranial (i.e., neurally-located) physical structures that stand for extracranial states of affairs: their referents. In other words, mental representations, whatever their format (symbols, non-compositional sets of sub-symbols, images, etc.), are intracranial and sub-personal items (representing structures) that, in virtue of their contentfulness, entertain referential relations (representation relations) with extracranial items. The presence of content is supposed to be explained by a naturalistic story (causal, teleological, functional). Content is often described in terms of informational properties: mental representations are indeed made of physical vehicles that carry or bear some information about some states of affairs. The information they carry is trafficked by communicating subsystems that access, manipulate, create, or transform this information.

The representationalist orthodoxy generally admits that any representational phenomenon requires a representing vehicle (first relatum), referent (second relatum), content (providing the representation relation), but also users: a consumer (understander), and a producer (Millikan 1995). Both consumer and producer here are neural sub-systems. As is well known, representation denotes

[4]For this strong version (popular in cognitive science), linguistic productions express messages which are mental representations having a conceptual structure (Pinker and Lackendoff, 2005: 205). For the weak version, the production and understanding of linguistic productions crucially involve mental representations.

both a relation (the representation relation existing between two objects) and an entity (the representing vehicle, something that stands for something else). These two senses are not independent of each other: the possibility of the existence of a representing relation depends on the existence of a representing entity, and an entity can only be a representing entity if it entertains representing relations with something else. According to representationalist theory, the production, and the presence in the mind of some representing structure explains how extracranial objects of perception, reasoning, imagining, believing, meaning and referring are present for a cognitive system without supposedly being in it or physically in front of it. One can then categorize, recognize, classify, anticipate or act on it. X has cognitive relations with Y by possessing something in him that stands for Y, and that can be seen as standing for Y, or even replicating Y, as it enables X to have relations with Y when Y is not here.

The manifold non-representational uses of depend crucially therefore on the existence and production of mental representations. Even if (possibly embodied) linguistic activity involves the production of representations for the sake of action, it basically remains an expression of mental representations: representations produced in action are necessarily understood and used by representational mental processes that take the use context into account. To derive meaning from use (often having no representational purpose), mental representations are crucially required – they play the role of mediators between perceiving and understanding, meaning, and saying. Mental representations can be the objects (lexical, phonological, etc.) of inferential or computational processes; they can also constitute the knowledge "we" use in order to produce and understand linguistic productions. This is what is overwhelmingly found in the disciplines that constitute cognitive science. Conceptual frames (Fillmore, Minsky), structures of expectancy (Tannen), scenes, scripts (Shank), domains, mental images (Croft), mental spaces (Fauconnier, Turner), concepts (Fodor), dossiers of information, lexical entries, vivid names, cognitive models, mental files (Perry) – all name representational entities, built on the fly or stored in the cognitive system, that are activated or constructed every time the cognitive agent performs understanding, meaning, referring, conceptualizing, or imagining.

6 Beyond Representations

Needless to say, there is a longstanding critical tradition of CR in philosophy and cognitive science (Dreyfus, Dynamical Systems Theory, Enactive Cognitive Science, and so on). Our target here is CR regarding symbols as it addresses code directly. A basic epistemological criticism addressed to any kind of representationalism, including cognitive representationalism in the philosophy of mind, is the following. It was first clearly expressed by Humberto Maturana (1978) and later by Maturana and Varela (1980, 1998).

We, as observers, in some domain of description, can behold the organism from the outside, in its relations with a context we call "its environment". In order to explain the coordinating and coupling relations between the organism and the environment (including inputs and perturbations), we find it natural to posit the existence of a system of representations of the environment (as it is or as it should be responded to by the organism [5]) within the organism (in the brain). But this position of observer makes us overlook the differences between our position (i.e., ourselves considering the organism, its brain and the world) and the point of view of the nervous system of the organism: except for believers in homunculi, the latter does not have the same relations with the environment as we have, especially when we (and not our brains) think of, speak about or act in this environment (Maturana and Varela, 1987: 131-132). Maturana and Varela have strongly criticized the conflation which is too often committed between the operational or mechanistic perspective of the central nervous system (inside which mental representations are supposed to be found, be it at a neural or functional level) and the observer-dependent perspective of the theorist, considering from the outside (and in its relations with the environment) the organism in which the brain is located. These three perspectives (the brain, the organism in which the brain is located, the observer) are not all congruent in their relations with the environment, or in their causal mechanisms. True, the observer can see that a set Y of neurons enables some organism O to deal with X (an external state of affairs). But O's brain (in which Y is located) does not have the same relations with X as those that the observer and O have with X. The nervous system does not have access to the correlation the observer constructs between parts of itself and the world. We make semantic projections onto neural signals by looking at O, X and Y from our linguistic and perceptual

[5] That is, the environment (and/or its objects) in the form of a model, a scene, a prototype or concept.

perspective. The distinction between the observer-dependent perspective and the operational or mechanistic perspective of the nervous system is thus crucial. Conflating these two different perspectives leads us to make semantic projections onto mere neural signals, transforming them into referential or informational content bearers (concepts, models, scenes, etc.). From Maturana and Varela's perspective, if some intracranial items are said to carry information, it is only relative to the observer's stance (being the only one who is able to consider the environmental whole in which the items carry information, and is able to know about the correlation laws that then enable him/her to see information in some covariant relations). Sub-personal or informational contents are just expedients for satisfying our own interpretative needs (Hutto, 1999). Information is only a product of the observer-dependent perspective, as it is only from this perspective that a semantic consideration of the relations between the nervous system and its world is available, necessarily with concepts and words that are not posited within the system.

What applies to "mental representation" and "information" also applies to notions such as "coding", "message" or even "memory": they do not enter into the realization of intracranial cognitive systems since they do not refer to processes in them. Once again, talking about the system "coding" or "containing" some information is confusing a process that occurs in the space of human design and understanding with a process that occurs in the space of the dynamics of the nervous system (Maturana and Varela, 1980: 90). These notions, applied to the realm of cerebral autonomy do not only express over-simplifications, enabling us to make cerebral dynamics meaningful and to find our explanatory way into the complexity of the brain, they also present us with a bad picture of what the brain is actually doing (Freeman and Skarda, 1990). Representation-making and using, or information-making and using, do not represent any aspects of the operation of the nervous system; they are only epistemological artefacts producing unexplained explainers and category mistakes ("intrinsically meaningful mental representations", "communication between sub-systems"). When the observer speaks about information, she/he disregards the dynamics that produces the unity and the coherence of the nervous system, and symbolically condenses its effects. Symbol- or information-talk abbreviates dynamic patterns of biochemical events. The stability and predictability of these patterns lead us to telescope them into a linguistic mode of description. This necessary heuristic strategy becomes problematic when one overlooks the fact it is just heuristic, as when one treats neural patterns as actually standing for environmental properties and

events (present, possible, prototypical) or linguistic knowledge.

This general argument against representationalism is particularly important when one considers CR in the philosophy of language and linguistics. The argument applies to any description by a theorist of the relations between an organism and an environment. Maturana and Varela underline the important and potentially misleading character of the symbolic dimension of the descriptive tools of the observer of the relation, who is prompted to "linguify" the relations between the organism and the environment, and thus to put "a system of representations of the environment" inside the organism. This basic fallacy is even more present when the observer considers a symbolic creature: in order to explain how the creature is able to mean, to produce and to understand symbolic (and therefore code) productions, it is tempting to consider that its intracranial cognitive powers and processes already consist in the production, use and understanding of representational items, possibly linguistic ones (strings of symbols in a Language of Thought) – and thus to move the explanandum (representational systems) into the explanans (intracranial life). We explain symbolic structures by turning mind into a representational system, be it linguistic (symbols, dossiers, concept), conceptual (schemata, frames), or visual (images, scenes).

One obvious reason for CR lies in epistemological grounds. Many have scathingly criticized such so-called externalist positions (see Lakoff (1987) on Putnam (1981), meanwhile amalgamating externalism and objectivism without however ascertaining that their own postures were not objectivist. Moreover, the resort to cognitive science at large and the so-called "importation of its results into connex methodologies" is a step further towards the naturalization of mind. Gibbs (1989) characterizes CR by identifying two recurring traits: the cognitive and generalization commitments. One asserts that a cognitive linguist must pay attention to the congruence of her conclusions with related disciplinary areas; the other reaffirms the classical view according to which cognitive scientists must proceed by generalizing from a series of observable and concordant occurrences. These methodological guidelines, here turned into defining features, at least facilitate what we have called CR.

Hutchins (1995: 364) has insisted on what could be seen as a basic mistake of CR: taking as a model of intracranial cognition the extracranial devices and operations that cognitive agents use behaviorally in order to reason, calculate, interact, or memorize. These devices and operations are environmental scaffoldings. They include external representations (sentences, maps, images, scribbles, texts, models, charts, graphs, gestures, etc.), whose public (interpersonal, not

sub-personal), lasting, shared and transmissible character make it difficult to see how there might be intracranial representations. Once one seriously considers the ways in which intracranial entities work (intracranial entities that are supposed to be mental representations and symbolic representations), it might appear that there are so many differences between them that they may not be different varieties of the same species (i.e., the representational species).

A response to this criticism of representationalism could be to ask, "But what else can they be? How can the conceptualizing abilities of speakers be explained? Where can linguistic knowledge (lexical, phonological, grammatical, pragmatic) be stored, if not in mental representations?" Two final remarks must be made here:

1) It is one thing to represent some content (to encode it), and another to represent some object. If X encodes some content about Y (for instance, some content, theoretically defined, partially enabling the subject S to mean, infer or understand Y), X is not necessarily a representation of Y. Physical structures encode information (or even informational content) "about" something, but they do not necessarily represent the something of their encoding. The about-ness of encoding or storage (already a theoretical interpretation) is not the of-ness of representation (which is at the core of CR, and still comes with the image of cognition facing some environment).

2) Obviously, cognitive agents often figure out, conceptualize, imagine, think, judge and reason about abstract, absent, past, future and hypothetical situations and states of affairs, without the use of extraneous tools. There are (re)presentational mental acts at the personal level of our cognitive life. Sometimes, they may even come to us with pictures or images, at a phenomenological level of experience and description. Obviously, nobody seriously denies that the occurrence of these personal re-presentational acts depend on the occurrence of sub-personal neural events. But why should the personal presence of these contentful mental acts be equated with the presence of sub-personal representing vehicles inside these individuals? Very often, we achieve contentful performances in virtue of the use of concepts. It is easy to equate the occurrence of some concept at the personal level with the occurrence of neural events, possibly described at some functional level as symbols possessing some linguistic properties (Language of Thought). But before embracing this classical story, leading us to identify personal contentful mental acts with the presence of some sub-personal representations, one should at least be reminded of the existence of other theories of concept use. One is an explicitly externalist inferential role theory: it consists

in equating the occurrence of some concepts with interpretable behavioral dispositions of cognitive agents (mainly inferential dispositions), necessarily situated within linguistic practices. What makes some personal mental acts representational acts is not the fact that they are causally dependent on sub-personal events which possess a magical representational power, but the fact that the person who is producing the act is able to master the concepts she/he uses in conforming to the inferential norms of some symbolic/code community, and is thus interpretable as judging or thinking something which is meaningful from the inferential norms developed and applied by the community in which she/he is actively situated. From this perspective, intracranial neurological events are not the vehicles of the contents that are judged by agents; they rather constitute the material conditions of possession and exercise of the correct behavioral (inferential) capacities from which concept and performance (including representational properties) can be attributed to the agents. The brain can certainly exhibit responses to various states of affairs, but these responses are not the inferential responsibilities we endorse in order to judge or think something that is meaningful by virtue of our inclusion in a given community.

There is no such thing as a demiurge coding the reality of a world that he/she presupposes as such. Such a production has to be built and enacted, which no individual brain may initiate; we inherit and negotiate, and in turn the production is crystallized. That consideration alone debunks the very interest and purpose of willing to mimic a human brain that can anticipate and build a world using networks of stabilized meaning routines that are constantly renegotiated. But overall, in the light of the extracranial, inferential, "on runtime" nature of cognition that is defended here, how can it be compared to the symbolic compositions and massive classifiers that constitute contemporary AI? Should we prefer to relabel those technologies for mass data categorization?

7 Two examples to demystify AI

Science fiction likes to play with AI concepts. In movies and comics, the plot usually involves a robot which somehow becomes a threat to humankind. The robot was created by scientists, but they are overtaken by events as the robot develops its autonomy and begins to act of its own volition. Humankind becomes aware of the danger but it is too late and humankind may not be able to survive (Barrat, 2015). In fact, AI products are only highly specialized tools. They can

be compared to cogs and springs in mechanical devices; they cannot magically modify themselves or improve their autonomy. AI models are not smart; they are static systems trained by human hands. Even if they are efficient, the product is perceived as dumb as it fails to answer the client's needs. Voice-recognizing Amazon Alexa can add a reminder with the description, "Do something," and Apple Face Id struggles to tell two Chinese women apart. A well-designed AI can repeat simple fastidious human tasks like reading number-plates (Aron, 2011; Siegel, 2013; Anagnostopoulos, 2014). Like a hammer, AI is crafted thanks to human knowledge; it repeats what it has learned without understanding the meaning of the question or the answer (Searle, 1980a)[6].

7.1 Myth 1: Self modifying AI

An AI model can perform one task well, the only one it was created for by the developer. It cannot modify itself (Vargas, 2014). Its role is to take an input, to compute this input following the layered architecture chosen by its creator, and to provide an output category predetermined by humans (Domingos, 2012).

Let us suppose a company wants to create an AI to identify whether a cat or a dog is present in a picture (Figure 1). The developers opt for a perceptron. The latter is an algorithm for supervised learning of binary classifiers. To launch the training, the team has to annotate a dataset: millions of pictures of each category (in this situation, pictures of cats and dogs). During the learning phase, weights are changed progressively in order to achieve a high enough success rate. The team can add more training examples to be more precise, but the output can only be "dog", "cat", or "unknown". The model cannot create a "wolf" category; only the developers have the possibility to change outputs (Villani, 2018). If someone shows the system a picture of a horse and the AI model was not trained with any picture of this animal, the output may be a "dog". Only a human can detect the error; the AI system is unable to modify its weights as it is not now in training. Even worse, AI cannot be aware that the output is wrong since it has no idea of what a dog or a horse is. Even the concept of animal does not exist for it; it takes a group of pixels and associates it with a string provided by the developers (Searle, 1980a).

[6]"Suppose also that after a while I get so good at following the instructions for manipulating the Chinese symbols and the programmers get so good at writing the programs that from the external point of view – that is, from the point of view of somebody outside the room in which I am locked – my answers to the questions are absolutely indistinguishable from those of native Chinese speakers. Nobody just looking at my answers can tell that I don't speak a word of Chinese." (Searle, 1980, p.3)

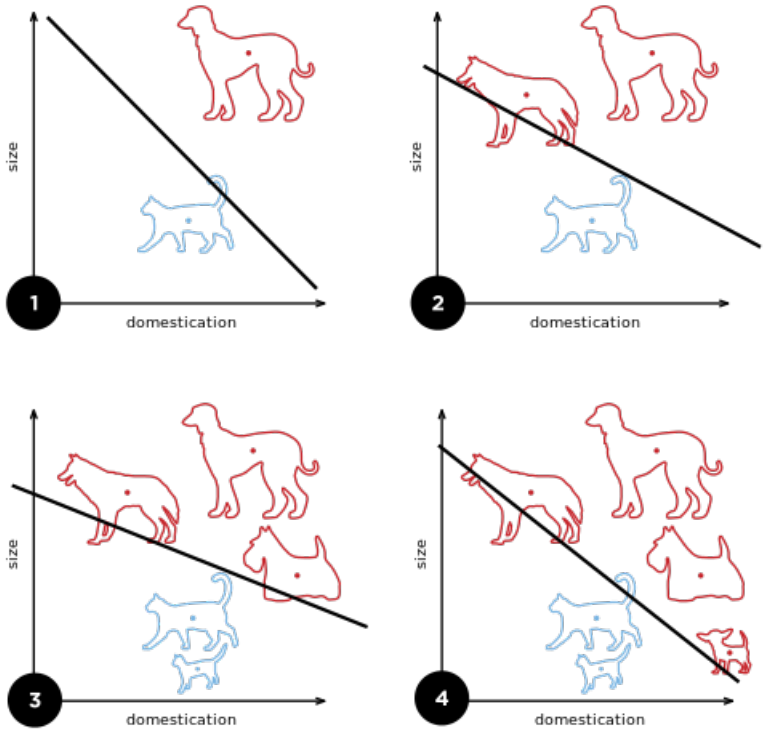

Figure 1: Perceptron evolution as training samples are added

Therefore, if an AI system fails to answer a request correctly, only the human is to blame. Since more than one human is concerned in the creation process, the real question is "which one?" When decision-making rules are written by programmers their authors retain responsibility (Bozdag, 2013). Let us focus on a traditional case: Facebook's EdgeRank personalization algorithm. It foregrounds content based on the author's popularity, the date of publication, the frequency of interaction between author and reader, media type, and a number of other factors If the company decides to change the weight of every factor, the users will change the way they use the platform. The party that sets confidence intervals for an algorithm's decision-making structure shares responsibility for the effects of the resultant false positives, false negatives and spurious correlations (Birrer, 2005)(Kraemer F and M, 2011)(Johnson, 2013). Operators also have a responsibility to monitor the ethical impacts of decision-making by algorithms

because "the sensitivity of a rule may not be apparent to the miner [...] the ability to harm or to cause offense can often be inadvertent." However, particular challenges arise for algorithms involving learning processes. To judge who is responsible for a bug, a computer system needs to be comprehensible and predictable, so that each part can be checked step by step. Models based on machine learning algorithms inhibit holistic oversight of decision-making pathways and dependencies (Matthias, 2004)(Burrell, 2016)(Tal, 2016). AI algorithms are not complex: a perceptron can be peeled into two functions and thirty lines of code, and simplified by statistical arrays, "if" else" blocks and thresholds. The issue for interrogation is the result which the model has created. As AI models are based on layers of weights that take specific inputs to produce predetermined outputs, it is very difficult to know what has influenced the weights. Is the sample database sufficiently large and diverse? Are the training samples sufficiently precise? Are the output categories well defined for the objective? Is the model architecture twisted? Was the training phase long enough? Or is it all these reasons combined? (Matthias, 2004) The company can obtain the results it expected with the database it created but it can never anticipate how the model will respond to unknown types of samples. Faced with this unpredictability, and given that an AI system cannot write itself, advocates of AI claim that every AI challenge will be solved thanks to hardware improvements and an upturn in raw calculation speed. To sum up, "more is better".

7.2 Myth 2: AI can surpass humankind

When Minksy was asked if a machine could surpass a human, his answer was: "... there's so many stories of how things could go bad, but I don't see any way of taking them seriously because it's pretty hard to see why anybody would install them on a large scale without a lot of testing" (Minksy, 1986). The key word in this statement is "testing". To claim that an AI model performs better than a human, developers need to evaluate its performance by creating measures of its accuracy or precision. Let us take an AI system that can recognize ten different colors. The output is the name of the color. The tests are simple if the inputs are pictures of solid colors. What about the picture of a sunset, with a color gradient? Is it orange or blue? Problems with human perception are mirrored in annotations and tests. The rabbit–duck figure (Wittgenstein, 1953) together with the color context would be another "Dress that Broke the Internet." As the

tests are dependent on human perception, the tests can only present samples already evaluated arbitrarily by the developers. In a fictional universe, if an AI system was presented to humankind as the best tool for recognizing colors and the outputs were names that had no meaning for humans, or the categories were indistinguishable by human perception, then it would be impossible to confirm that the color was classified correctly (Bentley, 2012). AI models are validated for publication after they have been tested with our particular bias. Therefore, it is possible to estimate the AI success rate on a dataset, and to improve the model by adding diverse samples, but ultimately the created AI can never replace nor outshine its creators.

In decision-making problems, it is argued that AI algorithms cannot create their own solutions, they can only produce predefined output. Moral Machine is an online platform, developed by Iyad Rahwan's Scalable Cooperation group at the Massachusetts Institute of Technology (Figure 2). It generates moral dilemmas and collects information on the decisions that people make faced with two destructive outcomes. The situations are presented like the trolley problem. The outcome of the survey is intended to be used for further research regarding the decisions of an autonomous driving technology.

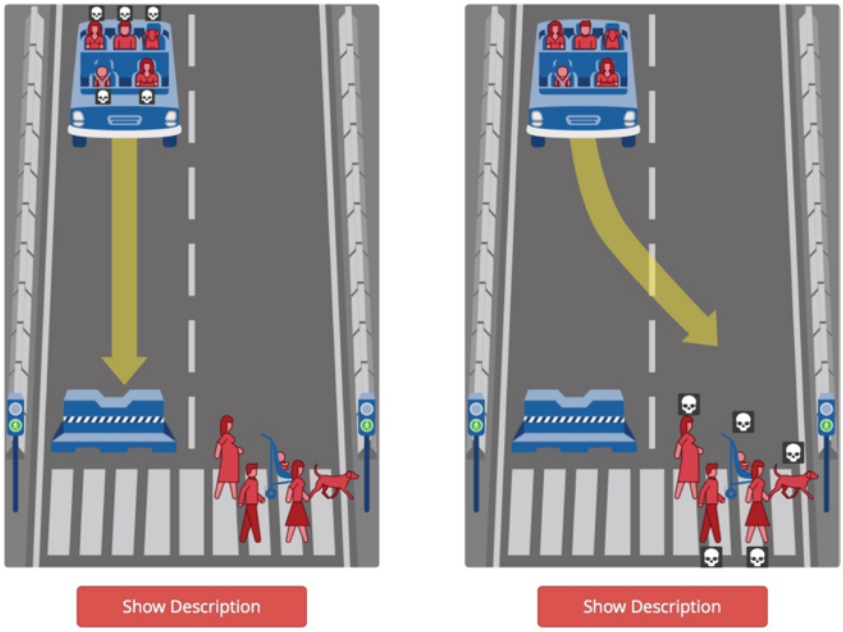

Figure 2: Moral machine: an example of the dilemmas

Psychologists like Christopher Bauman and Peter McGraw criticize the fact that these dilemmas are not realistic and inapplicable in real life; hence they do not enlighten us as much as we might hope about human decision making. Others (Khazan, 2014) argue that the presentation of the problem is not supposed to be representative of a real-world problem in the first place. It does not capture the human decision-making process. The trolley problem does not tell us what we would actually do if faced with an out-of-control streetcar; they simply highlight subtle quirks of our internal moral GPS system. According to Wittgenstein (Wittgenstein, 1929), ethical statements cannot be transformed into factual statements. An ethical value judgment is not a disguised factual statement. It is even "impossible" to be so. According to Wittgenstein, there is an insurmountable boundary between ethical usage and the relative use of language. Just as a judgment of ethical value cannot be a factual statement, a factual statement cannot be or imply a judgment of ethical value.

In 1961, Shannon said: "I confidently expect that with a matter of ten or fifteen years something will emerge from the laboratory which is not too far from the robot of science fiction fame." Fifty years later, we can barely make a robot

walk (DARPA Robotics Challenge, 2015).

Conclusion

This paper has argued that cognitive science emerged in a computer science fever and also came along with a confusion between computation and cognition. AI is admittedly a powerful tool but is also ill-named (a variably complex classifier). Such a tool (like software, like hammer) is a cognitive extension, a prosthesis that happens to be code in nature: symbols with instructional power and representative purposes. If the world has to be mimicked (represented), it has to be perceived. Perceptions are inherently unstable, and code units differ from what they are supposed to mimic. It has been argued repeatedly throughout the history of science that representations are mere (re)-presentations of continuously re-negotiated structures of meaning or information, tainted by technical contexts. "Representation as fallacy" is indeed a historical truism. We have argued that, like language, code is symbolic and representational; like language, code does not represent the world, and at best provides an overview or an insight into the world, the way a map represents a country. Therefore, as code (which is symbolic) constitutes AI (which is a tool that extends human cognition), we can legitimize both the computational and the externalist stances of cognitive science while rehabilitating AI as human-centered. The fruit of computations "extend and constitute" human cognition partially, so that fearing AI (understood as a computing process) is to discard the results of decades of cognitive science research on the nature of the human mind. AI should rather be framed as Augmented Intelligence; i.e., a human intelligence that is augmented by a computational tool without reason or conscience. Such a framework, we argue, pushes current ethical discussions towards complete and purely human responsibility: forgetting that AI is a prosthesis endangers the prosperity of AI as a transdisciplinary research field.

References

Adams, D. (1995). *The Hitchhiker's Guide to the Galaxy* (San Val)

Amazon (2014). *A STUDY ON AMAZON: INFORMATION SYSTEMS, BUSINESS STRATEGIES AND e-CRM*

Anagnostopoulos (2014). *License Plate Recognition: A Brief Tutorial.* (IEEE Intelligent Transportation Systems Magazine. Volume: 6, Issue: 1, pp. 59 – 67)

Aron, J. (2011). *How innovative is Apple's new voice assistant, Siri?* (New Scientist Vol 212, Issue 2836, 29 October 2011, p. 24.)

Barocas, S. and Selbst, A. D. (2015). *Big data's disparate impact* (SSRN Scholarly Paper, Rochester, NY: Social Science Research Network)

Barrat, J. (2015). *Why Stephen Hawking and Bill Gates Are Terrified of Artificial Intelligence* (Huffington Post)

Beck, K. (2004). *Extreme Programming Explained: Embrace Change* (Boston, MA: Addison Wesley), 2 edn.

Bentley, P. J. (2012). *Digitized: The science of computers and how it shapes our world.* (OUP Oxford)

Birrer, F. A. J. (2005). *Data mining to combat terrorism and the roots of privacy concerns* (Ethics and Information Technology 7(4): 211–220)

Bozdag, E. (2013). *Bias in algorithmic filtering and personalization* (Ethics and Information Technology 15(3): 209–227)

Brandom, R. (1994). *Making it Explicit. Reasoning, Representing, and Discursive Commitment* (Cambridge (MA)/London: Harvard University Press)

Brandom, R. (2008a). *Between Saying and Doing: Towards an Analytic Pragmatism by Robert Brandom* (OUP UK)

Brandom, R. (2008b). *Between Saying and Doing. Towards Analytical Pragmatism.* (Oxford University Press)

Burrell, J. (2016). *How the machine 'thinks:' Understanding opacity in machine learning algorithms results* (Big Data & Society 3(1): 1–12)

Clark, A. (2003). *Natural-Born Cyborgs. Minds, Technologies and the Future of Human.* (Oxford/New York: Oxford UP)

Davis M, K. A. and B, V. V. (2013). *Ethics, finance, and automation: A preliminary survey of problems in high frequency trading* (Science and Engineering Ethics 19(3): 851–874)

Diakopoulos, N. (2015). *Algorithmic accountability: Journalistic investigation of computational power structures* (Digital Journalism 3(3): 398–415)

Domingos, P. (2012). *A few useful things to Know about machine Learning* (Communications of the acm)

Dreyfus, H. L. (1972). *What computers can't do* (New York ; Evanston, Ill. ; London : Harper and Row)

Dreyfus, H. L. (1984). *Intelligence artificielle* (Paris : Flammarion , 1984)

Dreyfus, H. L. (2011). *All things shining* (New York : Free press , cop. 2011)

Dummett, M. (1995a). Bivalence and vagueness 61, 201–216

Dummett, M. (1995b). *Origins of Analytical Philosophy.* (Cambridge (MA)/London: Harvard University Press., chapter 4)

Fleck, L. (1935). *Genesis and development of a scientific fact* (University of Chicago Press)

Floridi L, F. N. and G, P. (2014). *On malfunctioning software* (Synthese 192(4): 1199–1220)

Fowler, M. (2002). *Refactoring: Improving the Design of Existing Code*

Fowler, M., Beck, K., Brant, J., Opdyke, W., and Roberts, D. (2002). *Refactoring: Improving the Design of Existing Code* (PEARSON)

Freeman, C., Walter & Skarda (1990). *Representations: who needs them? In J.McGaugh, N.Weinberger and G.Lynch (eds.), Brain Organization and Memory: Cells, Systems and Circuits.* (Oxford: Oxford University Press, 375-380.)

Fule, P. and Roddick, J. F. (2004). *Detecting privacy and ethical sensitivity in data mining results* (Proceedings of the 27th Australasian conference on computer science – Volume 26, Dunedin, New Zealand, Australian Computer Society, Inc., pp. 159–166)

Gibbs, R. (1989). *"Wha's cognitive about cognitive linguistics". In Eugene H. Casad (ed.), Cognitive Linguistics in the Redwoods.* (Berlin/New York: Mouton/De Gruyter, 27-115.)

Gibson, J. (1979). *The Ecological Approach to Visual Perception* (New York, Houghton Mifflin)

Google, I. (2011). *Guide de démarrage Google - Optimisation pour les moteurs de recherche*

Guignard, J. and Steiner, P. (2010). *Representation as Dogma: beyond Linguistic and Cognitive Representationalism* (Philosophy of Language and Linguistics. Volume II: The Philosophical Turn, P. Stalmaszczyk (éd.), Ontos Verlag (Linguistics & Philosophy Series), Frankfurt a.M./Lancaster, 2010, pp.241-257)

Guignard, J.-B. (2012). *Les Grammaires cognitives.* (Toulouse, Presses universitaires du Mirail)

H, A. (1971). *Eichmann in Jerusalem: A Report on the Banality of Evil* (New York: Viking Press)

Hajian, S. and Domingo-Ferrer, J. (2013). *A methodology for direct and indirect discrimination prevention in data mining* (IEEE Transactions on Knowledge and Data Engineering 25(7): 1445–1459)

Harris, D. and Harris, S. (2012). *Digital Design and Computer Architecture* (Elsevier)

Hutchins, E. (1995). *Cognition in the Wild.* (Cambridge (MA): MIT Press.)

Hutto, D. (1999a). *The Presence of Mind.* (Amsterdam: John Benjamins)

Hutto, D. D. (1999b). *The Presence of Mind* (John Benjamins Publishing)

J, S. (2015). *Distributed epistemic responsibility in a hyperconnected era* (Floridi L (ed.) The Onlife Manifesto. Springer International Publishing, pp. 145–159)

Johnson, J. A. (2006). *Technology and pragmatism: From value neutrality to value criticality* (SSRN Scholarly Paper, Rochester, NY: Social Science Research Network)

Johnson, J. A. (2013). *Ethics of data mining and predictive analytics in higher education* (SSRN Scholarly Paper, Rochester, NY: Social Science Research Network)

Kamiran, F. and Calders, T. (2010). *Classification with no discrimination by preferential sampling. In: Proceedings of the 19th machine learning conf* (Belgium and the Netherlands, Leuven, Belgium)

Khazan, O. (2014). *Is One of the Most Popular Psychology Experiments Worthless?* (The Atlantic)

Kraemer F, v. O. K. and M, P. (2011). *Is there an ethics of algorithms?* (Ethics and Information Technology 13(3): 251–260)

Lakoff, G. (1987). *Women, fire and dangerous things* (University of Chicago Press)

Lakoff, M., Georges & Johnson (1980). *Metaphors we live by.* (London: University of Chicago Press)

Langacker, R. (1987). *Foundations of Cognitive Grammar (1).* (Stanford: Stanford University Press)

Lions, J.-L. (1996). *Ariane 5 Flight 501 Failure.* (Paris, Inquiry Board report)

Macnish, K. (2012). *Unblinking eyes: The ethics of automating surveillance* (Ethics and Information Technology 14(2): 151–167)

Martin, R. (2008). *Clean Code: A Handbook of Agile Software Craftsmanship* (Prentice Hall), 1 edn.

Matthias, A. (2004). *The responsibility gap: Ascribing respon- sibility for the actions of learning automata* (Ethics and Information Technology 6(3): 175–183)

Maturana, H. (1978). *Biology of Language: The Epistemology of Reality.* (In: George Miller and Elizabeth Lenneberg (eds.), Psychology and Biology of Language and Thought. New York: Academic Press, 28-62.)

Maturana, H. and Francisco, V. (1980a). *Principles of Biological Autonomy.* (New York: Elsevier)

Maturana, H. and Francisco, V. (1980b). *The Tree of Knowledge. The Biological Roots of Human Understanding.* (Boston and London: Shambhala)

Millikan, R. (1995a). *White Queen Psychology and other Essays for Alice.* (Cambridge (MA): MIT Press)

Millikan, R. G. (1995b). Pushmi-pullyu representations 9, 185–200. doi:10.2307/2214217

Minksy, M. (1986). *The Society of Mind* (Simon & Schuster)

Neale, M. A., Northcraft, G. B., and Jehn, K. A. (1999). Exploring pandora's box; the impact of diversity and conflict on work group performance 12, 113–126. doi:10.1111/j.1937-8327.1999.tb00118.x

Neale, S. (1999). *On Representing.* (In: Lewis Hahn (ed.), The Philosophy of Donald Davidson. Library of Living Philosophers, vol. xxvii. Chicago/LaSalle: Open Court, 657-666.)

Noë, A. (2004). *Action in perception* (Cambridge, MIT Press)

Pinker, S. and Jackendoff, R. (2005). *The Faculty of Language: What's special about it?* (Elsevier Volume 95, Issue 2, March 2005, Page 205)

Putnam, H. (1981). *Reason, Truth and History.* (Cambridge (UK): Cambridge University Press.)

Romei, A. and Ruggieri, S. (2014). *A multidisciplinary survey on discrimination analysis* (The Knowledge Engineering Review 29(5): 582–638)

Récanati, F. (2004). *Literal Meaning.* (Cambridge (UK): Cambridge University Press.)

Sandvig C, K. K. e. a., Hamilton K (2014). *Auditing algorithms: Research methods for detecting discrimination on internet platforms* (Data and Discrimination: Converting Critical Concerns into Productive Inquiry)

Searle, J. (1980a). *Minds, Brains, and Programs* (Behavioral and Brain Sciences)

Searle, J. R. (1980b). Minds, brains, and programs 3, 417–424. doi:10.1017/S0140525X00005756

Shannon, C. and Weaver, W. (1948). *The Mathematical Theory of Communication* (Bell System Technical Journal)

Siegel, E. (2013). *Predictive Analytics: The Power to Predict who will Click, Buy, Lie or Die.* (John Wiley & Sons, Inc)

Simondon, G. (1958). *Du mode d'existence des objets techniques* (Paris, Aubier)

Skarda, C. and Freeman, W. J. (1990). Chaos and the new science of the brain

Tal, Z. (2016). *The trouble with algorithmic decisions an analytic road map to examine efficiency and fairness in automated and opaque decision making* (Science, Technology & Human Values 41(1): 118–132)

Travis, C. (2000). *Unshadowed Thought.* (Cambridge (MA)/London: Harvard University Press.)

Varela, T. E., F. and Rosch, E. (1991). *The Embodied Mind: Cognitive Science and Human Experience.* (Cambridge (MA): MIT Press.)

Vargas, P. A. . a. (2014). *The Horizons of Evolutionary Robotics.* (MIT Press)

Villani, C. (2018). *For a Meaningful Artificial Intelligence* (Ministère de l'Enseignement supérieur, de la Recherche et de l'Innovation)

Vincent, J. (2018). *Amazon reportedly scraps internal AI recruiting tool that was biased against women*

Wittgenstein, L. (1929). *Lecture on ethics* (Cambridge)

Wittgenstein, L. (1953). *Investigations philosophiques* (Editions Gallimard)

Zarsky, T. (2013). *Transparent predictions* (University of Illinois Law Review 2013)