Social Machinery and Intelligence

Nello Cristianini, James Ladyman and Teresa Scantamburlo
University of Bristol

Abstract

Social machines are systems formed by technical and human elements interacting in a structured manner. The use of digital platforms as mediators allows large numbers of human participants to join such mechanisms, creating systems where interconnected digital and human components operate as a single machine capable of highly sophisticated behaviour. Under certain conditions, such systems can be described as autonomous and goal-driven agents. Many examples of modern Artificial Intelligence (AI) can be regarded as instances of this class of mechanisms. We argue that this type of autonomous social machines has provided a new paradigm for the design of intelligent systems marking a new phase in the field of AI. The consequences of this observation range from methodological, philosophical to ethical. On the one side, it emphasises the role of Human-Computer Interaction in the design of intelligent systems, while on the other side it draws attention to both the risks for a human being and those for a society relying on mechanisms that are not necessarily controllable. The difficulty by companies in regulating the spread of misinformation, as well as those by authorities to protect task-workers managed by a software infrastructure, could be just some of the effects of this technological paradigm.

Keywords: intelligent agents, social machines, Artificial Intelligence, Human-Computer Interaction, Cybernetics, autonomous agents, teleology

E-mail addresses

Nello Cristianini, nello.cristianini@bristol.ac.uk (*)
James Ladyman James.Ladyman@bristol.ac.uk
Teresa Scantamburlo, teresa.scantamburlo@bristol.ac.uk,

(*) corresponding author

Social Machinery and Intelligence

Abstract

Social machines are systems formed by technical and human elements interacting in a structured manner. The use of digital platforms as mediators allows large numbers of human participants to join such mechanisms, creating systems where interconnected digital and human components operate as a single machine capable of highly sophisticated behaviour. Under certain conditions, such systems can be described as autonomous and goal-driven agents. Many examples of modern Artificial Intelligence (AI) can be regarded as instances of this class of mechanisms. We argue that this type of autonomous social machines has provided a new paradigm for the design of intelligent systems marking a new phase in the field of AI. The consequences of this observation range from methodological, philosophical to ethical. On the one side, it emphasises the role of Human-Computer Interaction in the design of intelligent systems, while on the other side it draws attention to both the risks for a human being and those for a society relying on mechanisms that are not fully controllable. The difficulty by companies in regulating the spread of misinformation, as well as those by authorities to protect task-workers managed by a software infrastructure, are just some of the effects of this technological paradigm.

1. Introduction

For Turing intelligence is intelligent behaviour (Turing, 1948). This accords with the way intelligence is understood in both Artificial Intelligence (AI) (Russell and Norvig, 1995) and biology (McFarland D and Bösser T, 2002). This paper focuses on a form of intelligent behaviour that is behaviour aimed at pursuing a goal, or purposeful behaviour: in this view, artificial, biological and even social systems can behave intelligently when they make choices that are better than random at achieving goals. For example, we can consider that an ant colony, collectively deciding to relocate its nest and choosing an optimal location, displays some form of intelligent behaviour; the same we could say of an animal navigating an unknown space to reach a destination, or of a robot doing the same.

This paper argues that there is a class of machines, that we call ``social machines', that can be intelligent in this sense. To establish this we characterise social machines and show that they can make choices and be attributed goals.

The creation of intelligent machinery has been an explicit scientific objective for at least 70 years (Turing, 1948; Wiener 1948), with generations of researchers debating the most appropriate design principles for such an ambitious goal, yet for most of this time accepting the idea that - once created - intelligent machines will be some version of an electronic computer. Now that several of the intended behaviours have been accomplished, along with many more which were not initially envisioned, many agree that we have created at least a version of machine intelligence.[1]. Yet we have created it in a way that none of the pioneers

[1] "*Artificial intelligence (AI) refers to systems designed by humans that, given a complex goal, act in the physical or digital world by perceiving their environment, interpreting the collected structured or*

of the field had imagined, and that we are still struggling to understand, perhaps in part because of a lack of adequate terms and categories.

This article aims at introducing terms and categories that can allow us to better understand the current direction of AI and its consequences. It deals with the question posed by Alan Turing in 1948: “whether it is possible for machinery to show intelligent behaviour” (Turing, 1948:1) in a novel perspective: it defines intelligent behaviour as that of an autonomous and goal-driven agent, and considers the special case of machines called ‘social machines’ (these terms will be defined in sections 2 and 3).

In other words, to slightly paraphrase Turing, we propose to investigate the question as to whether it is possible for social machines to show autonomous and purposeful behaviour. We believe that this perspective can give us the conceptual tools - terms and categories - to make sense of many problems emerging from Artificial Intelligence.

The first part of this article (sections 2-6) argues for a positive answer to this question. The second part (section 7 and 8) explores the implications of the existence of intelligent social machines of this type.

For instance, consider recommender systems, like those that power social networks or video streaming websites: they constantly interact with millions of users, framing their choices and learning from them, using this knowledge to make meaningful recommendations to other users, producing increased engagement, sales, or advertising revenue. Their actions are autonomous (in the sense defined in the next section), and are informed by the current state of the world, their past experience and their goals. There are many autonomous social machines that are commonly used in AI applications, and succeed in a vast array of intelligent tasks (finding relevant news, translating articles, driving cars, answering queries, etc.).

These intelligent agents are not just computers, but rather the combination of a digital infrastructure with the myriad participants whose choices and behaviours are elicited, harvested and leveraged in various ways by the central learning algorithms. There is no homunculus in charge of such systems: their behaviour results from the interaction of components, some of which happen to be human users (who do not need to be willing or aware participants). The seat of intelligence in those systems is neither in the algorithm nor in the participants but in the carefully orchestrated interaction of all the components (see for example (Dennett and Hofstaedter, 1981; Boden, 1990; Kurzwei, 2002)).

We argue that AI systems can be the result of interaction between computers and humans, who can be regarded as components of a machine that they cannot directly control. This includes situations where Human-Computer Interaction (HCI) is deliberately framed to incorporate humans into a social machine (Berners-Lee and Fischetti, 1999) with the aim of

unstructured data, reasoning on the knowledge derived from this data and deciding the best action(s) to take (according to predefined parameters) to achieve the given goal.” (EU Report of AI HLEG, December 2018)

eliciting valuable contributions from them. Interaction may include input obtained from carefully designed menus of choices or explicit requests such as driving a car or delivering a pizza. In this way HCI would be tasked not with implementing usable or accessible interfaces but, rather, with enabling humans to interact in a managed manner, each disclosing valuable information while possibly also benefiting from the system.

Social machines are an important concept elaborated in the context of Web technologies (Berners-Lee and Fischetti, 1999; Smart and Shadbolt, 2014), and are mechanisms formed in part by a technical infrastructure, and in part by human participants, whose behaviour is strictly shaped by the interface that mediates their interactions with the system. For example, Uber can be regarded as a social machine where drivers execute tasks specified by a software layer, while machines manage the planning, performance monitoring and payments. The paradigmatic example of a Social Machine is given in Section 4, which describes Games with a Purpose.

What makes this class of AI systems likely to be increasingly important in our society is the central position that they have come to occupy by being the mediators and coordinators of various human activities. These digital systems can not only learn by observation, they can also shape the interactions among users in a way that influences both individual and collective behaviour (Burr et al. 2018; and Cristianini and Scantamburlo, 2019). When an autonomous system can filter news and emails, suggest purchases or establish prices, its effects on the world can be real and far-reaching.

The next sections address the question above (can social machines exhibit intelligent behaviour, in the sense of autonomous and purposeful behaviour?) by establishing a terminology, assembled from Systems Engineering, AI, Game Theory, and Cybernetics. The next section is about agents in general. Section 3 characterises social machines, section 4 considers various features of their design, and Section 5 argues that they can exhibit intelligent behaviour. Section 6 considers some examples of intelligent social machines. Section 7 explores the implications of their existence for the design and philosophy of AI systems; for individual users and for society in general. Finally, Section 8 returns to the big picture and addresses the general direction of AI.

2. Teleological (goal-driven) Agents

An **agent** is any system that can act on the environment, in the sense that there is at least one variable in the environment the value of which is changed by its action. In general there are various possible incompatible actions among which a choice is made, and the change in the environment that results probabilistically depends on the action performed. This definition of agent is general and covers, for example, organisms, artifacts, and organizations.

An agent is **autonomous** if it chooses its actions by itself, in the sense that its internal processes and states determine the choices in a way that depends on its current and

previous interactions with the environment. An agent which is not autonomous, but whose actions are chosen by an external controller, is **heteronomous**.

Examples of autonomous agents include cells, plants, and software agents. A simple and classical example is the thermostat, which has only two possible actions which are chosen based on the temperature of the environment. Satnav systems, or automatic doors, are also simple examples. Examples of a heteronomous agents include an assembly line worker following precise instructions.

A **teleological** (or goal-driven, or purposeful) agent is an autonomous agent whose choice behaviour can be modelled as choosing based on the expected utility of the outcomes of the different actions it can perform. These actions do not necessarily need to be optimal, nor to have deterministic effects on the environment, but on average the agent's behaviour must do better than random in achieving some outcome, which can then be thought of as the agent's goal and hence as having utility for it. Of course, the same behaviour may be teleological in one environment, and not teleological in another environment (and teleological behaviour is not possible if the effects of actions on the environment are completely random). This class of agents that can fruitfully be described in non-causal language are called purposeful, as used in the classic paper (Rosenblueth et al., 1943): "The term purposeful is meant to denote that the act or behavior may be interpreted as directed to the attainment of a goal."

Typically agents must have information about the state of their environment in order to calculate the expected utility of different actions, in other words they need to be able to sense. An agent `senses' the environment if at least one variable of the agent's internal state can be coupled with at least one variable in the state of the environment.

The fact that the goal is normally set by the external designer of the agent does not necessarily reduce the autonomy of the agent, in that it chooses its own actions. All eels will tend to migrate to the Sargasso Sea every year, but individual eels will choose the right actions based on their specific situation, making them autonomous agents.

The notion of purposeful agent has been used in several disciplines for many years, to model animal or machine behaviour. Besides AI, this terminology connects with ideas from Cybernetics and the theory of bounded rationality, particularly with the idea of a purposeful agent in Cybernetics and that of rational agent in economics (von Neumann and Morgenstern, 1944). [2]

In cybernetics the idea of a goal-oriented (or teleological or purposeful) system[3] has been used as a unifying principle to apply control-theoretic notions to a general class of systems, including biological and social ones: "the behavior of some machines and some reactions of living organisms involve a continuous feed-back from the goal that modifies and guides the behaving object (Rosenblueth et al., 1943: 20)

[2] http://134.184.131.111/Books/Wiener-teleology.pdf

[3] "The term purposeful is meant to denote that the act or behavior may be interpreted as directed to the attainment of a goal" (Rosenblueth et al., 1943: 18).

In economics the same intuition is behind the notion of “rational agent”, an idea that is by now a standard in Game Theory (von Neumann and Morgenstern, 1944) and AI (Russell and Norvig, 2010; Cristianini 2010, Burr et al. 2018).

In economics an autonomous agent is defined as rational if its behaviour can be regarded as maximising a measure of performance. In the case of a stochastic environment, this is often modeled as an Expected Utility (Maximum Expected Utility Principle). This includes any system which chooses its actions among a set of options in a way to maximise its own chosen notion of utility (which might be private information). For example, a rational customer would buy the cheapest item in a set of otherwise equivalent items (if their utility is to save money). A rational agent is also called self-interested as it aims at a private interest or goal (von Neumann and Morgenstern, 1944), but this usage of words does not necessarily commit to any ethical or anthropological view. In practice, since in many situations agents make decisions with limited resources, it is often more convenient to adopt the view of “bounded rationality” which incorporates the idea of acting within several constraints (Simon, 1956). A bounded-rationality agent will attempt to chose the actions that maximise its benefit, under the constraints of incomplete information, limited computing resources, etc.

As for Turing Intelligence is Intelligent Behaviour, we will focus on goal-driven agents as a working model for intelligent behaviour. While some may propose that intelligent behaviour is more general than just purpose-driven autonomous behaviour, we still believe that this is - at the very least - a very important form of intelligent behaviour.

This leads to the standard definition of an agent in the literature on AI.as described in (Russell and Norvig, 1995), that is any system or entity that can sense its environment and perform actions. In the language of (Russell & Norvig), a `rational’ agent is one that can be described as attempting to maximise its expected utility, based on the available information and its current beliefs. In this sense an agent that is acting on false beliefs may be rational even though its actions fail to maximise its expected utility. As above, it is common in the AI literature to define autonomous agents as those that act under their own control

For the rest of this paper, we will refer to any agents that autonomously make informed decisions to pursue some goals as “intelligent”, (and we will use interchangeably the expressions goal-driven, purposeful and teleological behaviour). Note that pursuing a goal might include maximising a utility, or maintaining a homeostasis.

3. Social Machines (SMs)

While the notion of a Social Machine (SM) was introduced only 20 years ago (Berners-Lee and Fischetti, 1999), they existed long before then.

A machine is an apparatus composed of different parts, each one with a specific function, interacting together to do a particular type of work. There is no limitation to the technical

substrate of these parts and they can include hydraulic, electrical and mechanical components, among others. For example, a car, a dam and a telegraph can all be called machines.

Social Machines are a particular type of machine in which some components, carrying out specific subtasks, are human beings (whom we call 'participants').

In particular, a **Social Machine (SM)** is a machine in which human participants and technical artefacts (e.g. a car, a piece of software, a robot) interact with one another to perform a task that would not be achievable by any single part.[4] The interaction among the participants is highly structured, and mediated by the system, via an interface.

Mechanisms incorporating 'participants' can be found in multiple areas and include assembly lines, bureaucracies, auctions, markets, voting schemes, product delivery services, games, peer production, crowdsourcing, etc. Communication can be mediated by forms, ballots, purchase orders, etc. In this study we are particularly interested in cases where a software infrastructure mediates the interactions among the various participants, constraining and standardising their actions, monitoring performance and, under certain circumstances, assigning incentives.

Example 1. An assembly line is formed by a set of workstations where the same operations are always performed in a consistent way, and various parts are added to a product, as it moves through the line from station to station. Some of the operations are performed by machines, and others by people, in a highly coordinated and systematic fashion. Operations are often assigned and coordinated in a way to minimise motion and increase productivity - e.g. Ford's T model reduced the time of car production from 12 hours to about 90 minutes[5]. So long as all the operations are performed in the same time and way, it does not matter who performs them, i.e. they can be interchangeable. Human participants are typically used for operations that cannot be easily automated, but can be strictly specified in the desired input-output, so that participants act in a structured setting. Most importantly, participants cannot control the overall process: they are in fact parts of a machine and do not need to be aware of the overall results of their actions in order to do their job.

Example 2. A bureaucracy is a system composed by a large number of officials operating according to specific rules and protocols in order to administer a company, an organisation or a country. Usually bureaucracies, such as national post offices and banks, share a number of structural characteristics: e.g. functions and roles reflect a hierarchy, tasks are divided among participants and performed routinely, the communications across components occur via a standardised interface such as a form (these include communication across various parts, and with users), workflow and coordination of workers are specified by

[4] For more formal definitions see Smart and Shadbolt (2014) and Smart et al. (2014). Note that the concept of SM connects to many popular ideas such as collective intelligence, distributed cognition and social computing.

[5] More details can be found in the webpage devoted to the 100th anniversary of the moving assembly line: https://corporate.ford.com/innovation/100-years-moving-assembly-line.html

rules. In a bureaucracy, even though many tasks are performed by humans, each participant has limited autonomy and is not in the position to determine the behaviour of the overall machine, maybe not even be aware of it.

The most recent generation of SMs builds upon a web-based infrastructure and can count millions of participants, a possibility that was out of reach for most of former SMs such as assembly lines. For example, online crowdsourcing services, such as Amazon's Mechanical Turk, operate as modern assembly lines, where more than 2,000 workers are active at any given time (Difallah et al., 2018) to perform well specified tasks that might be difficult to automate.

In this type of modern SMs participants can be asked - for example - to tag or add comments to photos and videos, enter data from handwritten receipts, rate items, categorize images, answer questions, watch a video and so forth. The net result of their collective activity can be to process and sort through vast amounts of information. Note that participants in crowdsourcing do not need to know the overall goals of the machine. Moreover they do not even need to know the boundaries of the system, i.e. who or what else is part of it. In other words, they are not in a position to control the machine's overall behaviour.

As we discuss the relation between participants and the machine they are part of, we need to make an important distinction between levels: we refer to the behaviour of participants and that of the system, respectively, as the `micro' and 'macro' levels respectively. Between the two there is a relationship: the macro-level behaviour of the SM depends on the micro-level behaviour of each participant, and the latter can be, in its turn, under the influence of the former. In spite of this circularity, our main focus is on the dependence of the macro-level of the machine on the micro-level of the participants.

Much like a complex computation can be broken into many smaller mechanical ones, to be assigned to simple processing units, so a SM can distribute micro-tasks and incentives to a large number of participants, who might not necessarily be aware of the overall macro-task they are part of solving. The nature of micro-tasks and interactions varies depending on the context. For example, the online photo-sharing community Flickr explicitly asked its members to help the British Library by tagging billions of historical images[6], while the social news sharing and discussion website Reddit can obtain a notion of relevance of stories from the ordinary actions of 300 million active users who access, post, vote and comment posted content.[7] Uber drivers execute specific tasks as requested by a software infrastructure and

[6] See Flickr blog post : "The release of these collections into the public domain represent the Library's desire to improve knowledge of and about them, to enable novel and unexpected ways of using them The images currently have very little metadata associated with them […] and we want to invite you to discover the content in the library's photostream and add your knowledge to it by commenting and adding meaningful tags to the images" (http://blog.flickr.net/en/2013/12/16/welcome-the-british-library-to-the-commons/)

[7] While there are some conjectures on how Reddit ranking system works (see discussions in Reddit: https://www.reddit.com/r/TheoryOfReddit/comments/7e9f21/how_is_postranking_in_a_subreddit_determined/) the company announced major changes to its ranking system in 2017 (https://www.reddit.com/r/announcements/comments/5gvd6b/scores_on_posts_are_about_to_start_going_up/)

Wikipedia editors supervise the quality of article by a combination of computational tools and human deliberation (later sections will identify various types of SM based on the nature of the interaction).[8]

Example 3. An extremely successful class of SMs in leveraging human participation is formed by recommender systems. This is a type of software which provides suggestions on different types of objects, activities or services (books, videos, films, jobs, news, etc.) that might be of interest to a user (Ricci et al., 2011). In a recommender system participants are those users who feed the system with various signals, based either on an explicit request, such as completing a survey or rating a list of items, or simply just by providing implicit feedback (Oard and Kim, 1998), i.e. information retrieved from users' online behaviour. For example, YouTube's algorithm is fed by a variety of participants' actions or states such as whether a user is logged or not, IDs of watched videos, watch time, clicked and unclicked video impressions, time since last watched video (Covington, 2016).

With each action, users disclose personal preferences and interests to the system (Burr et al, 2018) and without their contribution the whole machine would not be able to do its job, i.e. offer good personalised recommendations. This is the very essence of SMs: harnessing the work of crowds to fulfill tasks that would be otherwise too costly, if not impossible, to perform. Depending on the context, participants can be called players, or users.

Note that this class of machines has some relation to the notion of socio-technical systems, which refers, in general, to any apparatus based on both technical artefacts and human agents (Vermaas, 2011). However, the analysis of this paper is based on the concept of social machines, as elaborated in the context of Web technologies (Berners-Lee and Fischetti, 1999; Smart and Shadbolt, 2014).

4. Design Properties of SMs

Before the discussion of autonomous SMs, that will be presented in section 5, this section outlines properties and design choices that can affect our relation with them. The terminology and concepts used here come from a variety of disciplines that describe design aspects of SMs that are relevant for the subsequent discussion. We describe how the interaction among participants, and between participants and machines, is mediated by a highly structured interface (which we shall call the API, in analogy to software-to-software communications (Reinhardt P, 2015) [9]); we distinguish between SMs whose participants are autonomous and those whose participants are heteronomous; and between those whose behaviour is centrally controlled and those with distributed control. The next section is about the affordances of participants.

[8] Tools can vary and range from programmes that update editors' personal watch lists (such as VandalProof) or help editors to revert contents with a single click (such as STiki) to software robots (bots) that inject data, fix spelling mistakes and correct structural features (such as capitalisation), among others. Currently there are more than 2 thousand bots working on English Wikipedia articles. See the list of bot tasks approved for use on English Wikipedia: https://en.wikipedia.org/wiki/Category:Approved_Wikipedia_bot_requests_for_approval

[9] This was first used in https://rein.pk/replacing-middle-management-with-apis

4.1 Structured Communication

All complex machines have the problem of managing internal communication and control across their multiple components. Traditional SMs may use structured forms: electoral ballots, multiple-choice questionnaires, pay-in slips. This format constrains the choices available to the participants, so that they can be managed in an automated way and in large numbers, and without knowing the inner workings of each component (this also has various implications discussed in section 5).

The constraints imposed by this structured communication (i.e. the type and the form of information requested) create a choice architecture (Thaler & Sunstein, 2008) that shapes the interaction among participants. In a SM, this translates into a careful design of user interfaces that can elicit the needed information (e.g. for a Uber passenger this involves: the intended direction and the rating of the driver).

Modularity is a common approach in designing complex systems. This implies dividing the macro-task into micro-routines and assigning them to one or more components which work as independent, standard modules. The problem of communication among modules is addressed by drawing on software engineering, where it is common to mediate communications between software modules by using standardised Application Programming Interfaces (APIs), a class of communication protocols which specify the input-output behaviour of each module. Systems as different as a weather sensor, a robot arm, or a remote database, can be controlled by a line of computer code: the API specifies how to form a request and how to interpret the answer. This allows large systems to be created, ignoring the inner details of each component, often also incorporating remote services. In analogy with that case, we will call API all instances of structured communication between participants in a social machine. This use was introduced by (Reinhardt P, (2015) which observed how today's developers can use lines of code whose effect is to request actions by humans, and observe the result, via an API.

Developers can use them when they build a system. In the case of crowdsourcing systems, APIs exist in which a computer system can make the request that a human performs a specific task. There are APIs for computer systems to order a Uber ride, a Pizza, book a hotel room, transfer some money, or request that a set of images is tagged by humans. Conversely, there are millions of users who subscribe to crowdsourcing services, eager to perform tasks in return for payment (typically these are micro-tasks performed for micro-payments). As an example, when Uber customers request a ride their customer app, this communicates with a central software system (via an API) and this sends out requests to possible drivers (via an API, through their app). The drivers can accept a task, in this way effectively completing a task that was given to them by an API. The system will also take care of the payment, and of the customer feedback. One of the key benefits of this modularity is that the rest of the system does not need to know whether an Uber car is driven by a person or by a robot, so long as it performs the required task.

In a SM, the operating components do not need to know either the nature or the objective of the overall task (each one needs to know only the accepted subtask), nor what other

components are doing (the I/O behaviour of each module is sufficient). All the communications are mediated and shaped by the API, a fact whose consequences are addressed in Section 6.

4.2 Centralised vs. Decentralised control

A key design choice in a SM involves how the behaviour of the participants is coordinated in a way to produce a consistent macro-level behaviour. This includes how problems (macro-level) are divided into subtasks (micro-level), and how their results are combined back to form a coherent solution. This affects also how participants communicate with each other and how each participant is monitored and incentivised.

Centralised. Let us consider the requirements of decomposing a macro-level problem (e.g. organising a journey, or training a neural network) into micro-level tasks (say: booking transport and hotels, or sourcing the appropriate training data) and then assembling all these results into a solution. This translation (from the macro-level language of a programmer to the micro-level of resources) is what is done in computer systems by compilers.

A compiler is a special piece of software that translates the source code of a programme, usually written in a high-level language (i.e. a language that abstracts from the physical features of the computer), into elementary operations that can be carried out by the hardware. Different hardware will require different compilers, because they have different ways to perform basic operations, while high-level code may not change at all.

When integrating human participants into a larger machine, Kearns and others proposed the concept of a "crowdsourcing compiler" (Kearns, 2011; Chen et al., 2016), to divide the various tasks into smaller ones and assign them to the right resources (either humans or machines). In this way instructions are made understandable and executable (efficiently) by human and artificial agents. Among others, the crowdsourcing compiler includes the property of specifying which subtasks are best carried out by machines and which by humans. It would be the responsibility of the compiler to know which participant or module is most appropriate for which task and, for example, who is best at some subtasks can be learnt by means of machine learning techniques as in the case of Mighty AI.[10]

As in a computer system, a social compiler would mediate between the two levels of abstraction: that describing the macro-behaviour of the whole system and that concerning the microscopic level of operating components. Because of its crucial operations, the social compiler contributes to centralise the control of a SM working as a mediator and coordinator of all machine's components

Decentralised. A SM does not necessarily need to have a central control system that gives tasks to different participants. A set of participants could be interacting together, or indirectly via the environment, and the overall activity still result into meaningful action at the

[10] For example, Mighty AI (https://mighty.ai/), a company crowdsourcing training data for computer vision models, "uses its own machine learning to determine what each member of the Mighty AI community is best at, then assigns them those jobs." (Stewart, 2017)

macroscopic level. One framework to study emergent behaviour in large systems of simple interacting modules is that of stigmergic mechanisms.

“Stigmergy” is a term introduced to analyse the collective behaviour of insect colonies, the main idea being that an agent can leave signs (“stigmata”) in the environment which are perceived by other agents and used to influence their next action (van Dyke Parunak, 2006). In the case of ant colonies, highly purposeful behaviour at the colony (macro) level can result from the interaction of individual (micro-level) agents, the ants. This observation has been abstracted to refer to a class of multi-agent coordination mechanisms. Swarming in various animals, quorum-sensing in bacteria, crowd movements in humans, all happen without a central controller.

For example, we can regard in this way the collective behaviour of YouTube users, each leaving a small trace of their choices, these choices then influencing the behaviour of future users via a recommender system. This can simultaneously result into: a) swarming behaviour in users (e.g. list of trending videos), and b) highly effective annotation of vast catalogues of videos, based on user preferences (e.g. likes, comments, sharing, etc). The problem of controlling the macro-level behaviour of a system (such as a swarm) by acting on the micro-level rules for its participants (such as he insects) is still an open problem. The analogy with this collective dynamics, and the general problem of controlling the goals of SMs of this type, is discussed in section 6.

Note that while it is possible for malicious participants to coordinate in order to influence the overall behaviour of the system, avoiding this possibility is a typical design concern in most social-AI agents, where it is typical to face attempts to spam search engines, alter markets, take advantage of other vulnerabilities. One of the countermeasures, in these cases, is to prevent coordination among participants, either by randomisation or by anonymisation or by creating competition among them. Other countermeasures may include computing trust scores for each participant. Generally, the overall behaviour of macro-agents depends on many independent contributions, but it is true that in poorly designed systems it is possible that a group of determined participants can hijack the overall system. This does not change the theoretical point we are making.

4.3 Role of participants: Autonomous vs. Heteronomous

Consider the participants in a traditional SM as agents, at a micro-level. In some cases we can consider them as autonomous (as in the case of a committee), in other cases as heteronomous (as in the assembly line).In the latter case, the participant makes no choice among actions. In the former case, only the goals are given and the participant then makes a choice.

Similarly, an important distinction between implementations of SMs is how the behaviour of the participants is elicited: in one case (e.g., the Mechanical Turk) the participants are required (either by a central controller, or by a decentralised system dynamics) to carry out a specific action (or a sequence of actions). In the other case (e.g., in Youtube), the participants may be assumed to be spontaneously performing these actions, perhaps while

pursuing their own interests such as searching for entertainment or buying products. In other words, participants can be modelled either as heteronomous or as autonomous agents. This distinction will have both practical and ethical consequences, so it is worth introducing it here, to discuss the properties and the implications of an autonomous SM. Note that these concepts also apply at the macro-level.

Comment on Participants as Autonomous Agents. By virtue of being autonomous, participants cannot be expected to automatically follow the instructions delivered by a task provider. For example, in a market, agents exchange goods and services under their own direction, they choose which move to make on their own, based on the perceived environment and their private interest. Note that in such a system a typical assumption is to conceive agents as motivated by some (private) notion of utility, which is used by the system to prioritise its actions.[11]

Example (Autonomous vs Heteronomous Participants). In systems such as Amazon Mechanical Turk or Mighty AI, participants act as heteronomous agents: their behaviour is specified by a designer, and they are rewarded for performing specific tasks, often following detailed instructions. For example, in Amazon Mechanical Turk, each task (i.e. the so-called "Human Intelligence Task" or HIT) has a standard description[12] and may include: instructions, allocated time, the amount of the reward and sometimes particular qualification requirements - in that case participants might be asked to perform a test before accepting a task. Normally, participants log into a specific online platform or use a smartphone app (such as Mighty's Spare5[13]) and, in some circumstances, might receive information about the general aim of their work - e.g. helping driverless-car to identify pedestrians and other obstacles or instruct computers to recognize objects - however how their work is used and what other human or technological components are involved in the system remain often opaque to the average participant.

It might be easier to conceive heteronomous participants in a SM with a centralized control such as Amazon Mechanical Turk. But it might be possible to contemplate situations where each agent follows a number of simple rules and produce a stigmergic behaviour without a central coordinator. In particular this can be observed where a social compiler is not necessary (e.g. flocks of birds or schools of fish). For example, Reynolds (1987) showed how a multi-agent model can simulate the same aggregate motion of flocks and similar phenomena by "programming" each agent (the so called "boids") with three simple rules, i.e. collision avoidance, velocity matching and flock centering.

---

[11] Informally, the notion of utility refers to a desirable outcome whose worthiness can be expressed in quantitative terms (e.g. the amount of money I can earn in a month by selling products on eBay). In decision theory, the notion is used in a more specific way, i.e. to express an agent's preferences over a set of alternatives, and in standard axiomatisation, it is framed in situations of uncertainty, i.e. in terms of expected utility (von Neumann and Morgenstern, 1944).

[12] A brief description of standard tasks in Amazon's Mechanical Turk is available here: https://docs.aws.amazon.com/AWSMechTurk/latest/AWSMechanicalTurkRequester/Concepts_HITsArticle.html

[13] Spare5 (https://app.spare5.com/fives) is a mobile app, developed by Mighty AI, where users earn small amounts of money usually by performing image-related tasks serving computer vision industry. Users are selected through a matching process based on users' data and companies needs (see the pipeline outlined in app support webpage: https://help.spare5.com/article/14-how-does-spare5-work).

5. Teleological Social Machines (TSMs)

This section argues that SMs can be teleological, that is autonomous and goal-driven (at a macro-level). As defined above, an agent is autonomous when it acts under its own control and heteronomous when it executes a script of instructions given by an external controller. We used this distinction for participants but we may apply the same to any system regardless of its physical substrate. For example, an excavator is a machine that needs a driver in order to perform its actions - so we call it heteronomous - while a robot or an organism, or the combination "excavator plus driver" are autonomous.

Also SMs can be characterised in the same way (as either autonomous or heteronomous) at the macro-level. For example, in the case of a heteronomous SM, a human coordinator, who might want to label 1M images, can distribute this task to human participants. In this case the social machine will be directly controlled by the human coordinator (for example, via a social compiler). On the other hand, in the case of an Autonomous SM (ASM) we can imagine the case where the system constantly labels a set of items (say, films or books) by presenting those that are more useful to annotate to the most appropriate human users, e.g. by leveraging exploration-exploitation mechanisms and machine learning techniques.[14]

Furthermore, we will give examples showing how an autonomous SM can be created whose effect is to pursue a specific goal, which its participants cannot influence.

Example (ESP-Game with a Purpose). An important class of SMs which has shown the concrete potentials of a large number of people working in parallel, thanks to a web infrastructure is the Game With A Purpose (GWAP)[15], a neologism which became popular with the ESP game (von Ahn and Dabbish, 2004). The ESP game consists of an online platform where thousands of pairs of players (randomly matched) play a sort of guessing game without communicating with one another. The objective is to guess which label the other partner will assign to a given image within a certain amount of time.

Since for each player the optimal strategy to maximise their own score is to choose the most probable word given each image - and coordination among players is ruled out by the randomised matching - a natural consequence of multiple players playing the game is the production of high-quality annotation for large dataset. Specifically, experiments with the ESP game showed that in 1 month 5,000 people can produce accurate labels for more than

[14] Exploration/exploitation mechanisms allow a designer of a system to both "explore" new possibilities and "exploit" existing knowledge. For example, this may occur when a newspaper editor needs to decide which article to propose to a given reader based on its past readings and new possible options (types of articles that the user has never read). For example the Washington Post uses similar techniques to figure out what headlines and stories will be displayed in users' devices https://www.washingtonpost.com/pr/wp/2016/02/08/the-washington-post-unveils-new-real-time-content-testing-tool-bandito/?utm_term=.1b446edd8793

[15] The expression was introduced in 2008 in a von Ahn's blog post (http://www.gwap.com/2008/05/hellow-world.html) to describes games leveraging online players to make computers intelligent, i.e. "teach computers things that they do not know yet".

400,000,000 images (von Ahn and Dabbish, 2004). In this case, the benefit for the overall system is just a by-product for the individual players - there is a difference in goals between the macro and the micro-level, that of the machine and that of the participants. Players do not need to realise that they are also participants in a wider mechanism.

The ESP game suggested that a system can be designed to serve two purposes in one (i.e. having fun and generating good annotations), a feature that can be relevant for the implementation of a SM (we discuss this in the next section).

Note that in GWAP the word purpose refers to the macro-effect of the interactions, or, in other words, the purpose of the SM as a whole (not that of the players).

NOTE: Systems like ESP-GWAP have a lot in common with markets, in that all participants need to guess the future actions of other participants, in what are called strategic games. A difference is that in ESP the participants can only win together, in markets it is the opposite. In both cases, their self-interested behaviour has the effect of furthering some overall goal: the effect that Adam Smith termed "the invisible hand".

Based on the same logic various other games with a purpose were introduced: e.g., a test asking people to transcribe characters in a distorted images (known as "Captcha"[16]) before completing a web transaction, enabled the digitization of vast archives such as Google book and The New York Time archive (see NYT article) Similar systems became common in the field of citizen science and collective problem-solving.[17] and collective problem-solving.[18].

We define Teleological Social Machines those SMs that are teleological agents, in the sense of section 2.

Note that the words participant, player and user may blur their distinctions. For example, YouTube users are also participants in a SM at the same time, to constantly update the annotation of its data and train its recommendation engine. In principle, there is no difference between the two forms of participation as they both implement the principle that (von Ahn et al., 2002) termed "stealing human cycles", in analogy with the idea of "cycle stealing" in the design of computer systems.

---

[16] Captcha, which stands for "Completely Automated Public Turing Test to Tell Computers and Humans Apart") was originally designed to test whether a user is human or a computer programme. A team of researchers at Carnegie Mellon led by von Ahn exploited the test to decipher archival texts that were not readable by an optical character recognition programme. The intuition behind that exploitation was that of "stealing cycles from intelligent humans" (von Ahn et al. 2004b: 9). ReCaptcha was bought by Google in 2009 (https://www.google.com/recaptcha/intro/v3.html)

[17] Examples of citizens science projects are FoldIt, a videogame used to help scientists fold proteins' structure (https://fold.it/portal/info/about) and Zooinverse, a web portal hosting multiple research projects such as classifying old scripts in arabic and identifying different city traffic sounds (https://www.zooniverse.org/about). Examples of collective problem solving include the projects of the Scalable Cooperation group at the MIT Media Lab https://www.media.mit.edu/groups/scalable-cooperation/projects/

[18] e.g. see projects of the Scalable Cooperation group at the MIT Media Lab https://www.media.mit.edu/groups/scalable-cooperation/projects/

On Purpose at the Macro Level. In this article, the same notion can be relevant at both the micro-level (self-interested participants) and at the macro-level (self-interested social machines). The notion of goal-driven behaviour is useful both at the macro and at the micro level: we can identify participants as goal-driven, as in the case of YouTube users; as well as the overall system (e.g. the YouTube recommender system). The notion of rational agents naturally connects to the idea of a purpose in systems-cybernetics.

Stafford Beer (Beer, 2002) noticed how it is the behaviour of a system that actually reveals its purposes. This observation was turned into a popular slogan, known as the PIWID principle from the initials of each word: for an autonomous agent “the purpose is what it does” (Beer, 2002). In other words, if the emergent behaviour of a system has the net effect of pursuing a certain goal, then this is the system’s purpose.

The PIWID principle (Beer, 2002) implies that the purpose of an autonomous agent cannot be derived from the intention of its designer; rather it is revealed by the emerging behaviour of the system, i.e. what the system actually does. Indeed, the performed action, rather than the intended one, is the key of a control system: “this control of a machine on the basis of its actual performance rather than its expected performance is known as feedback.” (Wiener, 1954: 24)

For example, in a social machine like eBay the interaction is framed in such a way that the macro-agent will move towards its goal, that is to identify the user willing to pay the highest price, even when the goal of each user is not to reveal that information, and to pay the lowest cost.

Social machines like the recommender system within YouTube, behave autonomously so long as their resulting outputs (i.e. video recommendations) follow from internal mechanisms combining technical and human modules via standardised interfaces (APIs). In such systems there is no external planner deciding what trending videos should be displayed. Their goals are to maximise some form of user engagement.

Usually, in an ASM participants cooperate as a distributed system (i.e. without a social compiler, which would imply an external programmer) but when there are mechanisms for delivering incentives and monitoring performance of participants another form of control may arise, centered on the choice architecture given by the API (we will address this case in the next paragraphs and in section 6).

Modeling both participants and the whole social machine as rational agents automatically introduces the question of the alignment between the utility at micro-level with that at macro-level: they do not need to be aligned (see section 6 for a discussion). For instance, Youtube chooses its actions autonomously, based on its past experience, and aims at maximising the click-through-rate and other measures of engagement. At the same time, users might want to maximise their entertainment value, while quite possibly minimising the time spent.

Again, Facebook might have the purpose of maximising participants' engagement and increase "reciprocity" with other apps, and Amazon might have the goal of increasing profits; but the participants might be motivated by information need in one case, and by various other interests in the second (for online shops the utilities might be even anti-aligned, with both parties trying to make the best bargain).[19]

In certain circumstances there might be the need to program the behaviour of a social machine at the macro-level to achieve a desired goal. For example, YouTube executives may want to pursue the goal of preventing outrageous contents. When this issue arises in a distributed system, like YouTube, a crucial problem is to define the rules or incentives for its participants (micro-level). This is a largely open technical question, in some communities called the micro-macro problem, particularly in the case of sociological theory (Alexander et al., 1987).

The theory of incentives (Laffont and Martimort, 2002) can also be used to frame this problem. Often the solution implies findings appropriate incentives, e.g. monetary rewards or penalties, bonuses or reputation, that ultimately make participants act in a way that maximise the expected utility of the macro-level (e.g. avoiding bad contents in our YouTbe example).

In the setting of strategic games, where the gain of a player depends on the actions of other players, there is also a discipline concerned with designing the games at a micro level in such a way that the macro-level system moves towards a desired goal: Mechanism Design (Börgers, 2015). As mechanism design is used - for example to design eBay auctions - one can consider it as one of the ways in which we can "program" Teleological Social Machines, though cases of explicit use of this method in designing the emergent behaviour are not commonplace. Note that any SM designed according to Mechanism Design principles, by definition, aims to maximise its own macroscopic utility, not that (micro) of the participants. So, in a SM the use of Mechanism Design may help to shape the interaction between participants in a way to control the macroscopic direction of the system by eliciting a desired behavior from the micro-level components.

6. Examples of SMs

This section illustrates the claim that certain SMs - like recommender systems commonly used by millions of people - can fruitfully be regarded as teleological autonomous agents. We review some examples to show how our various concepts and terms can help us describe and analyse these systems. Recall that autonomous systems choose their own actions, while heteronomous systems cannot choose their own actions, but goals are not chosen.

[19] The term refers to the amount of data that an external app shares with Facebook to be integrated with Facebook ecosystem. Facebook policy includes "giving third-party developers the ability to connect their apps to Facebook free, in exchange for those apps' giving data back to Facebook" (Roose, 2018). This emerged as part of a wider investigation requested by British Parliament (https://www.parliament.uk/documents/commons-committees/culture-media-and-sport/Note-by-Chair-and-selected-documents-ordered-from-Six4Three.pdf).

Amazon Mechanical Turk (AMT). When a user enters a task into AMT and a reward for completing it, the task is distributed to the available participants, along with detailed instructions. The participants complete the task as specified, and get paid the promised amount. Neither the macro-level, nor the micro-level are autonomous: the resulting SM is controlled by the human specifying its behaviour and the participants are told what to do by the APIs. The actions of the participants are specified, and so are the actions of the overall system. The centralised control is guaranteed by the social compiler which takes the task specified by the task provider, usually in high-level language, and distributed to the operating components

ESP Game. Two randomly selected participants are matched, and they can only score points if they guess each other's word - after being shown the same image. As it is improbable that two random guesses will match, in the case when they do match it is possible to assume that they reflect the contents of the image. The participants are autonomous, motivated by a desire to score points (for entertainment purposes). The overall system tends to a state where the set of images is increasingly annotated, and the uncertainty about each image's tags is reduced. So the purpose at a macro-level is to increase quality of annotation. The API presents a rigid choice architecture entirely controlled by the system (the photo, the space for words, and the forbidden-words). The participants are certainly autonomous, and the overall agent can be made autonomous very easily, as soon as the choice of participants and images is made - for example - in a way to annotate as many photos as possible, or to test the performance of as many users as possible. Note that the system might also have the problem of keeping participants engaged, in this case it might choose matches-collaborators in a way to maximise engagement.

YouTube. Here we focus on the user's perspective of YouTube, not that of the video makers. The participants (users) are autonomous and motivated by entertainment or information need. The recommender system is autonomous and motivated by maximising click-through-rate and users' expected watch time (Covington, 2016). The choice architecture (API) is shaped by the system, which proposes options to the users, mostly to induce them to click, but sometimes to extract valuable information, e.g. by using a/b testing (Covington, 2016). As a side effect, the databases of videos and users are annotated, and the behaviour of users has been shaped (engagement has been maximised). The user has no way of using the system without revealing information about their preferences, so the system comes to learn the micro-level goals or preferences of each participant. These can then be used in the choice architecture, to ensure that the macro-level goals of the system are maximised. Note also that this creates an indirect interaction among all users, via the recommender system, which could result in coordinated behaviour among human participants.

7. Discussion

The following subsections consider some implications derived from the development of teleological SMs, considered as implementing a form of intelligent behaviour. Our discussion

includes the evolution of AI as a discipline, philosophical and methodological considerations and ethical concerns.

7.1 The Social Turn in the field of AI

We deal with SMs on a daily basis, often without recognising them as such: recommender systems, conversational agents, spam filters make meaningful decisions that can show a level of understanding. None of these actions is controlled by a human, but neither is it determined by just an algorithm. Instead, they are the result of a distributed computation performed by a number of components, some digital and some human, these often not aware of being part of it. For example, the decision that an email message is to be filed as spam might depend on how many other users have treated similar messages in the past. The judgment and knowledge of participants, carefully elicited by a digital infrastructure (sometimes during routine interaction), is a central part of the decisions that the system will make in the future. Some of these participants are willing and paid task workers, others are unaware users, motivated by a diversity of goals. This combination of databases, learning algorithms, and participants is a SM, and is in many cases autonomous, in the sense of not being controlled by any other system.

With the introduction of machine learning at the centre of AI systems, and AI systems at the centre of the new data infrastructure resulting from rapid convergence in telecommunications, AI systems had access to vast samples of human behaviour to learn from: text, images, speech, and all sort of decisions. For many years, now, AI systems have been designed with the need to extract valuable information from humans, sometimes prioritising exploration over exploitation, devising strategies to avoid spammers to hijack the behaviour of AI systems (e.g. see anti-spam mechanisms in Google PageRank[20]) and deploying various types of collaborative filtering. By the early 2000s, a significant part of Intelligent Systems design was not just the design of learning algorithms, and not just the collection of relevant training data, but the careful design of the interaction with users, in a way that enables the AI systems to delegate a series of tasks to humans - be they data curation or labeling, or a more complete generation of data, reviews, feedback, and choices - or elicit valuable behavioural information.

Perhaps without intending it, AI designers found a formula for the generation of autonomous goal-driven behaviour: the inclusion of human participants into larger systems, which participants cannot normally control, which can only function when both machinery and humans interact in a tightly regulated manner. The participants do not need to be aware of the purposes of the machine they are part of. For example, as users go through their email inbox in the morning, the overall system pursues its goal to detect and remove spam in the future. Similarly, as users enjoy a few videos, the system pursues its goal to maximise click-through or other forms of engagement.

[20] For example see Google Webspam Report 2016: https://webmasters.googleblog.com/2017/04/how-we-fought-webspam-webspam-report.html

7.2 Consequences for the Philosophy of AI

When discussing the intelligence of systems that incorporate machines and humans, it is possible to confuse the intelligence of the participants with that of the system. As we take apart the various components of a SM we find that none of them actually is endowed with the capability to control it, perhaps not even to sense its environment as well as the whole system can. The quality of being autonomous and goal-driven resides at the level of the whole system (macro-level), not in any homunculus that might be contained in it: in other words, it is in the network of relations and interconnections, enabled but not controlled by the central data infrastructure which acts as an intermediator for the SM.

This situation may remind of various systems-cybernetics positions - “The stability belongs only to the combination; it cannot be related to the parts considered separately” (Ashby, 1954: 55) - but also of the classical John Searle’s Chinese Room Argument against machine intelligence (Searle, 1980). In that thought-experiment Searle described a system where he is in a closed room with a manual, containing strict instructions that allow him to turn incoming Chinese messages into outgoing Chinese messages, without actually understanding their meaning. Would the resulting system understand Chinese? While we do not engage with this question, we observe that the combination of person, instructions manual, and support stationary, would qualify as a SM in our framework, and, with appropriate design choices (e.g. APIs, utility, feedbacks, incentives, ect), its behaviour could be also goal-driven and autonomous. One of the standard responses to Searle's argument comes from a systems perspective, which argues that Searle is confusing the different levels at which the computation takes place, essentially confusing what we call macro and micro-levels in this article. The macro-level - in that response - would understand Chinese, while the micro would not: mechanical behaviour at a lower level might underpin meaningful behaviour at a higher level (Dennett and Hofstaedter, 1981; Boden, 1990; Kurzwei, 2002)

As we define intelligent agents in terms of their goal-seeking behaviour, we face a typical situation in systems cybernetics (Beer 2002), where effects and goals are identified. The PIWID principle (“the purpose is what it does”) implies that a system “wants” to end up in a stable state. By this definition, an ASM can be intelligent: its decisions (but also its goals) are the result of its internal dynamics and processing, and they can be aimed at achieving specific goals (particularly if designed by Mechanism Design, to result in a Nash Equilibrium, or if governed by a Utility Maximising central planning algorithm).

When designing SMs by using Mechanism Design, then it is clear that the Nash Equilibrium that is created by the mechanism is also the goal of the system. For example in the GWAPS (games with a purpose) the resulting behaviour would be classified as the macro-level goal of the system, regardless of the goals of its participants. Much like the human in Searle’s Chinese room, the goals and experience of the participants in a GWAP will not be relevant to the definition of the goals and experiences of the system, as its purpose is what it does, and is defined by its internal dynamics - designed for example to pursue a Nash Equilibrium, not controllable by any participant.

7.3 Consequences for the Design of AI Systems

The problem of creating an intelligent system, when this system is a TSM, implies that of gathering, and motivating, and coordinating a large set of participants, typically thought of as self-interested. Even if this difficult problem is solved, then there is the technical problem of designing the micro-level rules or incentives to shape their behaviour, in a way that the resulting macro-level emergent dynamics does what the designer wants. This is a very difficult task, and it is easy to imagine cases where the emergent goal of the system at best approximates the intended one and misaligns with participants' goals. This situation relates to the value alignment problem (Burr et al. 2018).

In our running example, of a video recommendation service, we need to distinguish the goals of its users (e.g. entertainment) with those of the system (e.g. maximising click-through rates) with those of the designers, which might be very different (e.g. detecting relevant and beneficial content). In some cases, the emergent behaviour of such systems might be entirely unexpected, including the spread of false or polarizing content, the induction of addiction, the promotion of junk content that does not satisfy either users nor designers.

The important consideration is that these systems are not under the control of a driver, as can be seen when social network companies struggle to prevent certain content from circulating. Once activated, these complex mechanisms will pursue their goals, hopefully those intended by their designers but not necessarily. One concern is that they might pursue them in a literal sense, for example if the goal is to maximise clicks in a news aggregator, they will not be necessarily concerned by the truth of the news items, or by the effects on the users. Could current social-media fake-news problems be a result of these machines having their own goals? We will see drift towards stable state of the system, that we cannot govern. Similar effects reflect the nature of cybernetic principles: "Behaviour that is goal-seeking is an example of behaviour that is stable around a state of equilibrium. Nevertheless, stability is not always good, for a system may persist in returning to some state that, for other reasons, is considered undesirable." (Ashby, 1956: 88)

Note that the presence of designers behind the goals of an autonomous system does not make it less autonomous because it is its choice of actions that defines its autonomy. Importantly, even when the goals can effectively be set by designers, they are still out of the control of participants. Furthermore, we must consider the possibility that the designers can fail to dictate their intended goals to a complex system.

There can be benefits in framing these systems within the language of Control Theory. A control system consists of a controller that monitors the state of its environment, compares that with some target state (the set-point) and acts upon it accordingly. The difference between the actual and the target state, also called "feedback", determines the action should be taken to correct system's behaviour.

In a SM we have at least two levels, macro and micro, that can be modeled from the perspective of Control Theory. A difficult technical question is how to organise the micro-level incentive structure so that the machine does what its designers want (e.g. generate revenue) without crossing certain social lines. The designer might want to control

the system at the macro-level, while the system itself needs to end up controlling the participants, at the micro-level. The micro-macro problem is a very difficult issue in system dynamics.

It is possible that current machines can only approximate the intended goals, but not fully align with them. What if companies did not find a way to change their micro-level incentive schemes, so to deliver on their social obligations? For example, we do not know if Facebook can maintain its profitability if it radically redesigns its personalised advertising system. It may be that the very features that make a SM profitable also make it difficult to control. The internal micro-level incentive structure of those machines is part of their “programme” and changing it changes their macro-level behaviour, the problem is how to program Social Machines in general. We do not have that technology yet, except for special cases like auctions.

7.4 Consequences for Individuals

There can be various kinds of individual consequences from participating in a social machine e.g. the alienation of joining an assembly line or a bureaucracy. However here we only consider the problems specifc to participating in autonomous and goal-driven SMs.

The emergence of a technology based on ASMs, even without a business model based on persuasion, has the potential to affect individuals in various ways (e.g. see Burr et al, 2018). The key consideration is that in this technology the interactions that give rise to the agent are all mediated by the software infrastructure, and human participants contribute by interacting with an API. This relation, where the interface is framed by the designer and its contents are adapted by the software to each participant, is both the key to these machines and to the concerns for individuals. We explore two orders of concerns: for human autonomy (e.g. Cambridge Analytica) and for employment (Amazon Warehouse, Uber riders).

The Power of Mediators - Working below the “API”. An important consequence of this turn is the emergence of a number of workers who are directly managed by a software layer. The article (Kosner, 2015) uses the metaphor of an API: these workers receive tasks and return their output via the API layer. This is not too big a stretch, since from the perspective of the coder, there are indeed instruction lines available that would result into a human reliably performing a task. Participants - such as delivery drivers - are monitored and rewarded by an algorithm, and do not necessarily need to be aware of the overall goals they are part of pursuing. This can give rise to both ethical and management considerations about wellbeing, with workers being forced by a mechanism to compete for the lowest wages, or nudged into working more than they would like, but also exposing them to the risk of being one day replaced by automated systems. Since their entire contribution is mediated by a software API (a structured interface, which shields the rest of the system from the specifics of a given module), the rest of the system would not be affected if they were replaced by a mechanism. This can involve Uber drivers (Wakabayashi & Conger, 2018), Amazon warehouse workers (Soper, 2018), and various types of translators, writers, data curators. In fact, many of these workers are already now working mostly to train their replacement. For these participants

that contribute for free, instead, there are concerns involving the risk of exploitation (Rosenblatt, 2018), particularly if there is the risk of behavioural addiction.

Interchangeability and Bargaining Power. An ASM, where the details of each participant are 'shielded' by an API, is formed by interchangeable components. Thanks to the implementation of APIs an ASM does not draw any distinction between different humans or between a human and an artificial component. Indeed, APIs enable an ASM to evolve regardless of the nature of actual participants. For example, Uber drivers are mostly humans now, but in future they might be replaced by autonomous vehicles (Wakabayashi and Conger, 2018) and that change would make no difference for the functioning of an ASM though of course it would make a difference for costs.

The interchangeability of an ASM components should not surprise even though some have referred to this quality in negative terms as a phenomenon called "pseudo AI". For instance, this was echoed by a news of a company which employed humans who replaced chatbots for a calendar scheduling services (Solon, 2018). As a direct consequence of interchangeability, Mechanism Design in this case is turned against the participants, as they could be led to offer cheaper and cheaper services, in an auction situation. The purpose of the macro-level agent would be to maximise its margins of profit, either increasing revenue or reducing costs.

Harvesting Human Cycles. Besides paid task-workers being managed by an API, there is also a large amount of willing participants who are not aware of contributing towards a larger goal that they ignore and they might even disagree with. In the field of human computation, often the emergent benefits of interaction are described in terms of 'harvesting human cycles' (von Ahn et al., 2002). All this refers to putting human interests and weaknesses to good use, leveraging them to perform some valuable task. This may also include gaining access to private information, such as psychometric quantities, as discussed in (Burr and Cristianini, 2019) and as seen in recent research (Kosinski et al, 2013).

On Choice Architectures. The interaction of the participant with the machine is framed by the machine itself, the interfaces determines the affordances of the participant, shaping the architecture within which any choice is made. For any individual working "below an API", life is framed by the information and the options made available by it. In some ways, it behaves like a Skinner Box. Three important considerations follow from this observation about choice architectures and APIs: the system can design mechanisms, by setting the micro-rules that the autonomous participant needs to follow; the system can exploit known psychological effects known collectively as nudging, to steer the decisions of the participant; and the system can extract some (rudimentary) psychometric information from the participant. We have already discussed above the first point: autonomous participants can still be harnessed if they play a game designed by the system. The second point is discussed in Burr et al. (2018), which discusses a variety of interactions between a human user and an intelligent software agent and examples may range from coercion, nudging to persuasion. The third point is discussed in (Burr and Cristianini, 2019) where a series of examples are provided, in which the interaction between users and AI systems can disclose psychometric information about the users. The main idea is analogous to the standard Item Response Theory of

psychology:if you can carefully design a series of questions, you can observe samples of behaviour that will reveal latent information about the user.

7.5 Consequences for Society

As we increasingly adopt SMs as part of our social infrastructure, we have come to rely on them as powerful mediators for many social functions, and as a new mass medium. While their influence at an individual level has been discussed above, we need to consider the possible effects on society as a whole. As the delivery of media content is mediated by an ASM, aimed at pursuing its own goals, what can be the effects on public opinion or the economy? The same can be said for the recommendation of products, and its effects on markets.

There are two key macro-level questions: how the machine can affect society, and how it can control its own participants. The second concern is rather fundamental: the internal (eg homeostatic) feedback loops that the machine uses to coordinate its own participants, eg formed by incentive schemes, determine their overall behaviour and ultimately their “implicit purposes”. The alignment between the goals of the machine and these of the designers might depend on the emergent dynamics connecting all participants via the infrastructure.

*On Swarming behaviour in online users***.** We have discussed above how in some cases SMs can result from “stigmergic” communication akin to that generating ant colonies. How do we know that these interactions across ASM participants (say on social media) do not induce sudden coordination and unexpected purposeful behaviour at the macro-level? In this case the SM would still be autonomous and have a purpose, but not one that we can influence nor predict. Once more, it is worth remembering that the purpose of a system is what it does, and when this is a Nash Equilibrium, there is nothing that individual users can do in order to change it. Controlling swarm dynamics is an open problem in engineering. Also in the case of non-stigmergic systems (non-swarm systems), the goal of these systems is their Nash Equilibrium, the kind of state that no participant can change by acting alone. In other words, no participant nor central planner needs to be aware of the goals of the system.

8. Conclusions

Many of the original objectives of Artificial Intelligence have been achieved in the past decade, and we now routinely interact with autonomous systems and rely on their decisions in our everyday lives. We entrust them with screening our transactions, filtering our mail, selecting the news we read, driving our cars, translating our documents, suggesting purchases, finding bargains, proposing perspective partners, and more. Yet many of these behaviours have not been automated in the way that was originally envisioned by the pioneers of the field. Indeed, we are still struggling with understanding what we have created, and with its consequences and possible regulation, perhaps in part because of a lack of adequate terms and categories.

This article aims at developing a terminology and a conceptual framework that can be useful to discuss issues arising from this new class of artificial intelligence. In many cases, the

autonomous systems we interact with are examples of autonomous and goal-drivel social machines, and the boundaries of those systems include large amounts of human participants. Thinking in these terms, of agents and goals, of micro and macro levels, of mechanism design and control theory, of actions and utilities, can help us understand this form of AI, and the social challenges that it can pose.

As early as the 1960s, Wiener suggested some moral problems “caused by the simultaneous action of the machine and the human being in a joint enterprise”. These include the issues of individuals operating at a slower time scale and the creation of more opaque systems: “The result of a programming technique of automatization is to remove from the mind of the designer and operator an effective understanding of many of the stages by which the machine comes to its conclusions and of what the real tactical intentions of many of its operations may be” (Wiener, 1960:1358).

We have reached a point in the evolution of AI where the question of Alan Turing “whether machines are capable of intelligent behaviour” interacts with Wiener’s concerns, as the current generation of AI agents largely rely on Autonomous Social Machines. We call these SMs autonomous because they choose their own actions, while pursuing goals that are (hopefully but not necessarily) set by their designers.

Just because most of our examples are based in the area of entertainment or commerce, if we look at the future, we may be misguided in thinking that these machines will only power online shops. As Turing warned in 1948, “*the limited character of the machinery which has been used until recent times [...] encouraged the belief that machinery was necessarily limited to extremely straight-forward, possibly even to repetitive, jobs*”. In fact, it is now possible to imagine these systems to be used one day in much more consequential positions: governance, education, and increasingly institutional roles.

The key point of this article is that it is possible and fruitful to study a class of Social Machines as autonomous and goal-driven agents. This class includes agents that are often regarded as products of AI, such as the recommender systems populating our news feeds, or those running auction websites, possibly choosing prices, filtering emails and giving answers to common questions. In many cases, these complex behaviours result from a blend of human participants and software infrastructure. As we struggle to understand and regulate these new systems, terms like purpose, participant, macro and micro, can help us frame and phrase the new challenges that we encounter. Also drawing the boundaries of the intelligent entities under investigations, to include their participants, can help us understand them in a different way.

The current generation of AI agents satisfy our basic definition of intelligent behaviour, of being autonomous and goal-driven, and we analyse the implications of this observation. This goal-driven behaviour at the macro level emerges from the interactions between participants and machinery, and it is not always possible for designers to obtain the desired macro behaviour by acting at the micro level of participants. This is where dis-alignments can and do appear.

In this article we call ‘the goal’ of the system the actual direction in which it moves, as is in the example of the ESP game or some aspects of the eBay platform, or YouTube recommender system. These are agents whose actions are best understood in the light of their ‘goals’, and are selected based on information coming from vast numbers of participants. In all these examples, the central algorithm tasked with combining and processing information is not capable of goal-driven behaviour on its own, but it emerges from the interaction of multiple participants.

From the user’s perspective, it is most appropriate to ascribe goals to the system. What a YouTube of Facebook user experience, when presented with highly relevant and compelling recommendations, is the interaction with a sophisticated goal-drivel entity, whose purpose may be to increase engagement or some other metric. This purpose may be built into its overall structure, at the time of designing it, but the specific actions are chosen by the entity itself based on its state and past experiences.

Note that in the case of well-built systems, it is very difficult for malicious participants to coordinate their behaviour in order to hijack the macroscopic agent, although this can and indeed has happened in the past. The existence of faulty macro agents, which can be hijacked and therefore lose their autonomy, does not undermine the autonomy of successful macro agents that can manage to interact personally with billions of users.

Note also that the fact that the designer may choose the overall utility of the agent does not reduce the autonomy of the macro agent, since it is the actions that need to be autonomously chosen in order to meet the definition of autonomy. Furthermore, it is not at all guaranteed that the intended goal by the designer ends up being the actual goal of the resulting system - except for a few cases where theoretical guarantees can be given - in general there may be dis-alignment or even drift in the overall ‘purpose’ of a complex agent.

An important technical question for the future of the field is: how do we control such AI systems? Setting aside the question of who has the right to do so, can the organisation running one such system really control its specific behaviour? And a related ethical question is: Who is responsible for their behaviour? The users? The designers? The participants? The system itself?

We also need to face the important question of the effect of these systems on the individual participants, on their autonomy and privacy, if they are - as it seems - capable of nudging and steering, and of accessing private information about their users. The Cambridge Analytica scandal is just an example of this bigger question, with suggestions that individual persuasion was enabled by access to psychometric profiles of users. The fundamental point in all this is the relation between human users and social-machines.

Acknowledgments . NC and TS were supported by the ERC Advanced Grant ThinkBIG.

References

Alexander, J., Giesen, B., Munch, R., & Smelser, N. (Eds.) (1987). *The Micro-Macro Link*. Los Angeles, CA: University of California Press

Ashby, R. (1954). *Design for a Brain*. New York: Wiley and Sons Inc.

Ashby, R. (1956). *Introduction to Cybernetics*. London: Chapman & Hall

Beer, S. (2002) What is cybernetics?. *Kybernetes* 31(2), 209-219

Berners-Lee, T., & Fischetti, M. (1999). *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web*. New York: Harper Collins

Boden, M. (1990). Escaping from the Chinese Room. In Boden, M. (ed) *The Philosophy of Artificial Intelligence* (pp 89-104). New York: Oxford University Press.

Börgers, T. (2015). *An Introduction to the Theory of Mechanism Design*. Oxford: Oxford University Press

Burr, C., Cristianini, N., & Ladyman, J. (2018) An Analysis and Model of the Interaction Between Intelligent Software Agents and Human Users. *Minds and Machines* 28(4), 735–774 https://doi.org/10.1007/s11023-018-9479-0

Burr, C. and Cristianini, N. (2019). Can Machines Read Our Mind? Under Review.

Chen, Y., Ghosh, A., Kearns, M., Roughgarden, T., & Wortman Vaughan, J. (2016). Mathematical Foundations for Social Computing. *Comm. of the ACM* 59(12),102-108

Cristianini, N. (2010). Are we there yet? *Neural Networks*, 23(4), 466–470

Cristianini, N. & Scantamburlo, T. (2019). Social machines for algorithmic regulation. Under review

Covington, P., Adams J., & Sargin E. (2016), Deep Neural Networks for YouTube Recommendations, RecSys's 16 September, 15-19, 2016, Boston, MA, USA

Difallah, D., Filatova, E., & Ipeirotis, P. (2018) Demographics and Dynamics of Mechanical Turk Workers. In Proceedings of WSDM 2018: The Eleventh ACM International Conference on Web Search and Data Mining, Marina Del Rey, CA, USA, February 5–9, 2018 (WSDM 2018),

Gugliotta, G. (2011). Deciphering Old Texts, One Woozy, Curvy Word at a Time. The New York Times. https://www.nytimes.com/2011/03/29/science/29recaptcha.html accessed 20 November 2018

Hofstaedter, D. & Dennett, D. (1981). *The mind's I: fantasies and reflections on self and soul.* New York: Basic Books.

Kearns, M. (2011), The Crowdsourcing Compiler, http://www.cis.upenn.edu/~mkearns/papers/CrowdsourcingCompiler.pdf, accessed 16 September 2018

Kosinski, M., Stillwell, D. & Graepel, T (2013). Private traits and attributes are predictable from digital records of human behavior. *PNAS,* 110 (15), 5802-5805, https://doi.org/10.1073/pnas.1218772110

Kosner, A. W. (2015). Google Cabs And Uber Bots Will Challenge Jobs 'Below The API'. Forbes https://www.forbes.com/sites/anthonykosner/2015/02/04/google-cabs-and-uber-bots-will-challenge-jobs-below-the-api/#5f308a3869cc
Accessed 17 October 2018

Mcfarland D and Bösser T, (2002) Intelligent Behavior in Animals and Robots; MIT Press

Kurtzweil, R. (2002). Locked in his Chinese Room. In Richards, J. (ed.). *Are We Spiritual Machines: Ray Kurzweil vs. the Critics of Strong AI* (128–171). Seattle: Discovery Institute

Oard, D., & Kim, J (1998), Implicit feedback for recommender systems. In *Proceedings of the AAAI Workshop on Recommender Systems*, 81-83

Reynolds CW (1987) Flocks, herds and schools: A distributed behavioral model. *Computer Graphics*. 21, 25–34. doi:10.1145/37401.37406

Reinhardt P, (2015) "Replacing Middle Management with APIs”; https://rein.pk/replacing-middle-management-with-apis

Ricci, F., Rokach, L., & Shapira, B. (2011) Introduction to recommender systems handbook. In F. Ricci, et al. (eds.), *Recommender systems handbook*, pp. 1–35, Springer, Berlin

Roose K (2018) Facebook Emails Show Its Real Mission: Making Money and Crushing Competition. The New York Times, https://www.nytimes.com/2018/12/05/technology/facebook-emails-privacy-data.html accessed on 6 December 2018

Rosenblatt, A. (2018). When Your Boss Is an Algorithm, New York Times, https://www.nytimes.com/2018/10/12/opinion/sunday/uber-driver-life.html Accessed 17 October 2018

Rosenblueth, A., Wiener, W., & Bigelow, J. (1943). Behavior, Purpose and Teleology. *Philosophy of Science*, 10, 18–24.

Searle, J (1980). Minds, brains, and programs. *Behavioral and brain sciences*, 3(3), 417-24

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T. & Hassabis, D. (2017). Mastering the game of Go without human knowledge. *Nature*, 550, 354

Smart, P., & Shadbolt, N. (2014). Social machines. In M. Khosrow-Pour (ed.) *Encyclopedia of Information Science and Technology*, 6855–6862, Hershey, Pennsylvania: IGI Global

Smart P., Simperl E., & Shadbolt, N. (2014) A Taxonomic Framework for Social Machines. In: Miorandi D., Maltese V., Rovatsos M., Nijholt A., Stewart J. (eds) *Social Collective Intelligence: Combining the Powers of Humans and Machines to Build a Smarter Society*, 51-85, Cham: Springer

Solon, O. (2018), The rise of 'pseudo-AI': how tech firms quietly use humans to do bots' work, The Guardian, https://www.theguardian.com/technology/2018/jul/06/artificial-intelligence-ai-humans-bots-tech-companies accessed on 6 July 2018

Soper, S. (2018). Amazon's Clever Machines Are Moving From the Warehouse to Headquarters. Bloomberg, https://www.bloomberg.com/news/articles/2018-06-13/amazon-s-clever-machines-are-moving-from-the-warehouse-to-headquarters accessed 11 December 2018

Stewart, J. (2017), The human army using Phones to teach AI to drive, Wired, https://www.wired.com/story/mighty-ai-training-self-driving-cars/ accessed 11 November 2018

Thaler, R., & Sunstein, C. (2008). *Nudge: improving decisions about health, wealth, and happiness*. London: Yale University Press.

Tufekci, Z. (2018) YouTube, the Great Radicalizer, The New York Times, https://www.nytimes.com/2018/03/10/opinion/sunday/youtube-politics-radical.html accessed on 9 September 2018

Turing, A. (1948). Intelligent Machinery, Report for the National Physics Laboratory, http://www.turingarchive.org/browse.php/C/11 accessed 10 January 2019

Van Dyke Parunak H. (2006) A Survey of Environments and Mechanisms for Human-Human Stigmergy. In D. Weyns, H. Van Dyke Parunak, F. Michel (eds) *Environments for Multi-Agent Systems II. E4MAS 2005. Lecture Notes in Computer Science*, 3830, 163-186, Berlin, Heidelberg: Springer,

Vermaas, P., Kroes, P., van de Poel, I., Franssen, M., & Houkes, W.A (2011). *Philosophy of Technology. From Technical Artefacts to Sociotechnical Systems*; Morgan and Claypool: San Rafael, CA,USA

von Ahn, L. & Dabbish, L. (2004). Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*, pp 319-326 http://dx.doi.org/10.1145/985692.985733

von Ahn, L; M. Blum & J. Langford. (2002). Telling humans and computers apart (automatically) or How Lazy Cryptographers do AI. CMU Tech Report. https://www.cs.cmu.edu/~mblum/research/pdf/tell.pdf accessed 23 October 2018

von Neumann, J., & Morgenstern (1953). *Theory of Games and Economic Behaviour*. Princeton University Press, Princeton

Wakabayashi, D., & Conger K (2018). Uber's Self-Driving Cars Are Set to Return in a Downsized Test, The new York Times, https://www.nytimes.com/2018/12/05/technology/uber-self-driving-cars.html accessed on 10 December 2018

Wiener, N. (1948). *Cybernetics, or control and communication in the animal and the machine*. Cambridge, MA: The MIT Press

Wiener, N. (1954). *The Human Use of Human Being*, Boston: Da Capo Press

Wiener, N. (1960). Some Moral and Technical Consequences of Automation. Science 131(3410): 1355-1358 DOI: 10.1126/science.131.3410.1355