A Case for Specialisation in Non-Human Entities

El-Mahdi El-Mhamdi[1], **Lê-Nguyên Hoang**[2], **Mariame Tighanimine**[3, 4]

[1]École Polytechnique, France
[2]Calicarpa, Switzerland
[3]Lise, Cnam CNRS, France
[4]University of Neuchâtel, Switzerland
mariame@tighanimine.com*

Abstract

With the rise of large multi-modal AI models, fuelled by recent interest in large language models (LLMs), the notion of artificial general intelligence (AGI) went from being restricted to a fringe community, to dominate mainstream large AI development programs. In contrast, in this paper, we make a *case for specialisation*, by reviewing the pitfalls of generality and stressing the industrial value of specialised systems.

Our contribution is threefold. First, we review the most widely accepted arguments *against* specialisation and discuss how their relevance in the context of human labour is actually an argument *for* specialisation in the case of non human agents, be they algorithms or human organisations. Second, we propose four arguments *in favor of* specialisation, ranging from machine learning robustness, to computer security, social sciences and cultural evolution. Third, we finally make a case for *specification*, discuss how the machine learning approach to AI has so far failed to catch up with good practices from safety-engineering and formal verification of software, and discuss how some emerging good practices in machine learning help reduce this gap. In particular, we justify the need for *specified governance* for hard-to-specify systems.

1 Introduction

In 2023, the European Union announced its Artificial Intelligence Act. The AI Act consists in a series of regulations on AI that not only impacts European AI, but ultimately also other countries and regions in the world in what is known as the Brussels' effect (Bradford 2020, 2024; Siegmann and Anderljung 2022; Gunst and De Ville 2021). One of the most puzzling features of the AI Act is that, due to intensive lobbying from large AI companies (Perrigo 2023), the Act ended up putting more restrictions on some specialised AI models than it does on AI models claiming general capabilities. To capture the absurdity, consider an analogy: in medical practice, would a general practitioner have more freedom to operate on a patient's eye than an ophthalmologist? The answer is obviously negative. The less specialised a medical doctor, the more restrictions different medical legislation or codes of ethics[1] would impose on what they can do.

In the 2020s, AGI, in which the "G" stands for *general*, has increasingly become an overused term to include several *desirable* properties in AI entities. However, it is unclear whether *generality* ought to be regarded as *desirable*, especially in terms of auditability, reusability and security, but also in terms of industrial value. In this paper, we review social science and statistical arguments *for* generality, and highlight their limits in the context of non-human entities. We then instead make a case *against* generality, by leveraging arguments from adversarial machine learning, from complex system engineering and from social sciences. We then stress the value of *specification*, which requires specialised systems.

The rest of this paper is organised as follows. In Section 2, we lay down the broader context of our work and review useful notions such as *generality*, *task* and what we call "*friends of specialisation*" such as decentralisation or separation of powers. In Section 3, we propose three arguments against specialisation from social sciences, economics and statistics, and discuss their limitations in the context of non-human entities. This brings us to Section 4, in which we propose our arguments for making non-human entities specialised. Section 5 complements our argument by making a case for *specification*. The section also discusses the limits of specification, and the need of *specified governance* to address these limits. Finally, 6 concludes this paper.

2 Context

Historically, the word "algorithm" comes from the name of Muhammad ibn Musa al-Khwarizmi, and really consisted of decomposing complex tasks into elementary trivial subtasks, just like in al-Khawarizmi's pioneering book, providing simple step-by-step recipes to simplify problem resolution for a wider audience, and in particular, inheritance case resolution for lawyers.

This intuitive approach, insisting on breaking problem resolution to simple tasks, was further formalized by the pioneers of computer science, in particular Alonzo Church (Church 1936) and Alan Turing (Turing 1936). Both argued that any computable function can be written as a composition of elementary ultra-specialised hard-wired mechanical operations, typically on a Turing machine (or in lambda calculus). In today's computing machines, such operations are the (now numerous) logical circuits engraved in

*Corresponding author

[1]See e.g., (American Medical Association 1871; Ministère de la santé 2004 (initial version 1953)).

processing units like CPUs. The assertion that these fixed circuits suffice to perform *any* computable information processing function is now known as the Church-Turing thesis. This thesis arguably thus asserts that, in principle, any general capability is a composition of (a very large number of) specialised operations.

However, in practice, when software consumers use an algorithm, the compositions of such specialised operations is abstracted away, so that the consumer ends up interacting with a complex system, whose range of capabilities can be narrow or wide. Lately, the rise of language models, but also that of fully integrated and connected cloud services like Microsoft 365, has led to the commercialisation of "general systems". Such systems (are said to) provide a large number of information processing services.

In fact, many software developers might not view their software solutions as a composition of specialised operations, especially if their solutions leverage external libraries. While some of these codes simply translate the developers' programming language codes into elementary binary codes (these are *compilers*), it has become extremely common for developers to use complex "general-purpose" libraries, especially when importing, e.g., large language models. Such systems could be argued to "empower" developers, as they can now effortlessly program solutions to many more problems. However, they can also be argued to reduce their capability to understand their own systems, especially in terms of capability, safe usage and emergent risks. In contrast, developers could favor building upon *specialised* libraries, which provide them with precisely the tools they needed for some well-defined tasks, such as optimization solvers. The developer would then only be blind to such specialised operations, which potentially having the capability to clean and sanitize their outputs. It is noteworthy that such specialised libraries could be made to be equally blind to how they are used.

What we might call "specialisation blindness" can also be found in other activities or professions. A fast-food franchisee, for example, considers that their business is selling hamburgers or other products sold in this type of restaurant. However, upstream, a franchisor has to call on several specialist trades to develop all the building blocks for a successful franchise: the right furniture (interior designers, furniture salesmen...), the right recipes (chefs, nutritionists, etc.), the right ingredients (farms, bakers, drinks wholesalers...), the right business model (bankers, accountants...), the right location (estate agents, notaries...), etc.

Specialisation blindness is arguably an unavoidable consequence of collaborating to perform complex tasks (Boullier and El Mhamdi 2020; Tighanimine 2025). However, the question we raise in this paper, is how to organise collaboration to make it an asset rather than a risk. Essentially, we will argue that the key lies in well-specified specialisation.

Defining Generality

In the context of the AI Act, generality is often associated with the *purpose* of a system. The greater the diversity of use cases, the more "general-purpose" the system. This definition of generality-by-purpose can be also found in Model Cards (Mitchell et al. 2019), which asks AI system developers to specify the "intended use cases" of the developed systems.

We note however that there may be ambiguity behind the definition of a use case. Assume for instance that a language model is used to translate messages on a social media platform. Is the use case "social media"? Or merely "translation"? Or even more simply "next-token predictions", given an original text and a translation request prompt? Clearly, if we consider the first answer ("social media"), then the system may appear very "general-purpose"; but more basic algorithms like json serializers or symmetric encryption should then be regarded as more "general purpose". Conversely, the more we dig into the very precise use of a system ("next-token prediction"), the less "general-purpose" it will appear to be.

A leaked document (Maxwell 2024) from OpenAI and Microsoft revealed their use of a more financial approach to defining AGI. Namely, they (privately) regarded as "general" a system that generates hundreds of billions of dollars in revenues. In a sense, this approach is similar to counting use cases of a system, but it proposes to weigh the use cases by their financial added values. It is however noteworthy that, given this definition, Blackrock's trading algorithm Aladdin [2] and Google's ad targeting algorithm AdSense [3] should be regarded as much more "general" than language models.

Another approach that may be considered to assess the generality of a software system is the list of *application program interface* (API) calls it defines. Indeed, each API function proposed by the system may be viewed as a task that the system can offer to solve. Of course, in practice, some API functions can be regarded as themselves more "general" than others, e.g. if they involve multi-modal inputs rather than text-only. While still not fully rigorous, in the sequel, we lean towards this definition. We will regard a system as general, if the number of tasks that external users can ask the system to perform is large.

Note that a very specialised system in this sense may nevertheless be extremely complex. For instance, even though a language model is only accessible through a "next-token prediction" API, or if its only API yields content recommendations, it may be itself composed of trillions of parameters (Lian et al. 2022). In fact, any large human organization that is specialised to deliver a very specific task must perform a large number of internal tasks, such as accounting, sales or human resource management. Nevertheless, we will regard it as specialised if the number of services it offers to *external users* is small (and well-specified).

AI Agents

Since 2024, most large AI companies have been promoting AIs with *acting* capabilities (Yao et al. 2023; Acharya, Kuppan, and Divya 2025; Murugesan 2025; Borghoff, Bottoni, and Pareschi 2025). As opposed to conversational AIs

[2] In 2020, Aladdin managed 21.6 trillion U.S. dollars in assets (Ungarino 2020).

[3] Google's ad revenue amounted to 237.86 billion U.S. dollars in 2023 (Bianchi 2024).

that merely send messages to end users, such so-called *AI agents* (or *agentic AI*) can execute commands on information systems. More precisely, the Model Context Protocol (MCP) (Hou et al. 2025; Ray 2025) was recently proposed to standardise the way AI agents do so. Namely, each AI agent is given a list of API calls that they can call, along with the documentation of these API end points.

It is noteworthy that such agents differ from other concepts of *autonomous agents* that rather refer to the capability of such agents to behave well in some general frameworks, like reinforcement learning (Hutter 2003; Veness et al. 2010), often with respect to a given objective function that sends rewards (Ben-Porat et al. 2024; Kierans et al. 2025). Such frameworks typically do not invoke the access to API calls.

In some regards, MCP thereby somewhat formalizes the concept of generality we discussed earlier. In particular, one could be tempted to give to an AI agent a large number of API call accesses, to enable them to solve complex tasks. Our case is especially *against* such developments. In particular, like others (Blili-Hamelin et al. 2025; Radosevich and Halloran 2025), we argue that it is not in the interest of companies and societies to integrate such agents in their organizations.

Defining the Notion of Task and its Granularity

Providing a single definition of what a task is can be tedious. We can start by trying to understand its centrality in automation and in the more general context of work (which inspires our thinking on specialisation), in division of labour.

In a work context, a task can be considered as an action associated with goals, means and conditions of execution. Within the context of prescribed work, a task corresponds to all the goals and procedures defined in advance, and meets codes, performance requirements, and quality standards. The elements of prescription - even though they may be *underdeveloped* - are generally found in all work activities. Various institutions, professional hierarchies, public authorities or professional groups set tasks, objectives, procedures, directives, rules, and decrees defining what can be done or must be done. Social sciences of labour have repeatedly shown by means of empirical studies that what workers, managers and even executives do is not simply the execution or application of the task prescription. Everything they do systematically goes beyond the divide between "task" (what is prescribed) and "action" (what is done). In the language of computer science, these may be translated as "specification" and "code execution".

When we look at the history of the organisation and rationalisation of work, particularly through the emblematic examples of Taylorism (Littler 1978), Fordism (Watson 2019) and Toyotism (Dohse, Jürgens, and Nialsch 1985), all of these paradigms have involved defining and sorting out specific tasks, selecting workers to carry them out, and more generally (over)specialising work and then automating it. For the past two centuries, automation has been the hallmark of work transformation. Broadly speaking, it starts with an agent's craft know-how. Automation then consists of extracting the skills from the agent, and of encoding them into a procedure, i.e. a series of simple, structured and repetitive tasks be they physical or cognitive. In debates about the "future of work" (Brynjolfsson and Mitchell 2017), jobs that are identified as least likely to be automated are those whose tasks are unstructured, non-routine, and whose performance is associated with critical thinking, long chains of reasoning or complex planning, a certain level of creativity, etc. Arguably, this is because the resolutions of such tasks are hard to encode into procedures, though the rise of machine learning opens the door to the encoding of procedures that are humanly hard to describe, e.g. in the Kolomogorov-Solomonoff sense (see Section 5).

Automation - the creation of machines and algorithms to carry out tasks previously performed by humans - is a constant reconsideration of division of labour. To quote Karl Marx (himself echoing Charles Babbage), machines emerge as a synthesis of the division of labour (Marx 1847). So the notion of task is at the heart of automation. An important question is to know what level of definition and operations to adopt in order to automate tasks and, ultimately, specialisation. There are limitations that make this exercise complicated. A prominent challenge is to select the granularity of a task (and thus its level of specialisation). There are informational and computational hurdles that make this exercise complicated. For example, there are many tasks that we understand tacitly and carry out without being able to state their explicit rules and procedures. In other words, formulated by Polanyi's paradox (Autor 2014), "we can know more than we can tell" (Polanyi 2009). Moreover, automating certain tasks and not others also means renouncing possible ways of working and other possible uses (Simondon 1958).

Defining a task can also refer to expressions such as granularity or modularity. We can define granularity and modularity using the definitions proposed by Benkler (Benkler 2006) which are the basis of his socio-economic production model "Commons-based peer production". Modularity is a way of dividing a project into smaller components (modules) that can be developed independently before being assembled to form a whole, in order to maximise the flexibility and autonomy of the contributors. Granularity refers to the size of the modules, defined in terms of the time and energy required by the participants to produce them. It is important to specify the modules and their quantity in a project, because the interest and investment of the participants depend on it. As we will see, the two concepts are also central in computer science and system engineering.

3 The Case Against Specialisation

Before making our case for specialisation, let us highlight the most common arguments against specialisation, and discuss why they feel ill-fit to today's AI systems.

The Case of Humans

Overspecialisation harms workers' mental health and flourishing. Although some arguments may have been anterior, the most severe criticisms of the division of labor opposed Taylorism, assembly line work, and what came to be

known as Scientific Management(Bookwalter Drury 1918). This work organisation generated an extreme division of labour with a fragmentation of tasks, a sustained pace of execution, a high degree of dependence between workers, and the impossibility of envisaging the unexpected events. Criticisms from the social sciences of work were to emerge in the middle of the twentieth century and the rise of increasing industrialisation (particularly in the automobile sector) and mechanisation. Friedmann (Friedmann 1955; Friedman 1961) talks of "work in crumbs"[4]" to describe the evolution of forms of work with a high level of automation, and in particular the work in the factories. There have also been concerned with the psychological damage caused to workers by alienating, unskilled, and depersonalized work.

Today, criticisms of the excessive division of labor and hyper-specialisation concern many economic sectors (food industry, logistics, textile industry, etc.). They particularly focus on the aspects that are detrimental to the body and mind of individuals. One of these industries is precisely the information and communications technology industry, and particularly digital platforms (on-demand platforms, social media), and more generally artificial intelligence using digital labor, micro-work and data workers, especially in the global south countries (Casilli 2020; Williams, Miceli, and Gebru 2022).

Overspecialisation leads to a loss of meaning/purpose. High specialisation can cause workers to lose interest and motivation in their work, and can impact performance (Loukidou, Loan-Clarke, and Daniels 2009). A high level of specialisation tends to lower workers' stimulation and motivation levels, while increasing their boredom and disengagement (Hackman 1969; McCauley et al. 1994).

However, while the psychological and physical damages on human workers are important to consider, algorithms are arguably not subject to such concerns. At least following (Gibert and Martin 2022), while they could be regarded as *moral agents* (i.e. they have moral duties), they should not be regarded as *moral patients* (i.e. there is no moral duty to protect them). Therefore, the above arguments do not seem to apply to non-human agents.

Arguments from Economics

In economics, instead of specialising in a given step of the production chain, companies may be tempted to invest in *vertical integration* (Coase 1937; Bresnahan and Levin 2012). Namely, they could want to own their entire supply and delivery chain. The theoretical analysis of (Arrow 1975) indeed found that imperfect information can incentivize such vertical integration, while that of (Carlton 1979) instead stresses the role of transaction costs. (Grossman and Hart 1986) further underlined contractualisation costs: if specifying the conditions of an agreement between a supplier and a producer is challenging, then there will be strong incentives for vertical integration. The theories of (Arrow 1975) and (Carlton 1979) have both found support, e.g. by (Lieberman 1991) in the chemical industry. On top of this, (Fetz and Filippini 2010) empirically observed better investment coordination and less financial risk following the vertical integration of the Swiss electricity sector. In practice, varying working conditions may instead incentivize outsourcing, and thus vertical disintegration (Ricardo 1817; Kakabadse and Kakabadse 2005).

In the software industry, yet another phenomenon may incentivize both vertical and lateral integration, namely the *network effect* (Shapiro 1999). This describes situations where the value of a product is increased by the wide use of this product, or of similar related products. A classical instance of this is Facebook's huge investments in social media. For instance, by acquiring Instagram and by connecting it with their other platforms, Facebook has exploited the network effect to increase the value of their anterior assets (Li and Agarwal 2017). Similarly, Google and Apple now produce both hardware, operating systems and applications, thereby offering a unifying solution to their customers. The value of the hardware is then augmented by a corresponding optimized optimized system, which is itself augmented by its connection to a dedicated cloud system (Vergara and Del Rosario 2012). Further, software companies have incentives to invest in lateral integration, to propose all-in-one highly connected digital workspaces, as is the case for instance of the Microsoft 365 product. This further helped them push AI-based solutions, such as Microsoft co-pilot (Skendzic and Kovacic 2012).

However, while there may be some positive consequences for some users, critics point out that such integration incur societal risks, especially in terms of market power (Landes and Posner 1997). In particular, the technology sector has been repeatedly criticized for its dependence on a small number of actors, as exemplified by the landmark antitrust case against Microsoft (Economides 2001) and the current ongoing cases against Google and Facebook (Brennan 2025), and by the numerous recent calls to enforce antitrust laws to other actors (Munir et al. 2024; Davies and Georgieva 2024; Wörsdörfer 2024).

Additionally, integration can increase systemic risks. This is well illustrated by the report of the Cybersecurity and Infrastructure Security Agency on the 2023 Microsoft Online Exchange Incident (Silvers and Alperovitch 2024). The penetration of Microsoft's cloud infrastructure by foreign actors has not only compromised the targeted US institutions; it may also be endangering *all* of Microsoft's customers. Moreover, the high connectivity of all of Microsoft's cloud services means that a given customer may be compromised, even if they only use some of these services.

Having said this, at the scale of a country, especially in a context of tensed geopolitical tension, vertical integration could in fact be a consideration of national security. More generally, it could be in a given system's best interest to seek generality, if they want autonomy and resilience. However, it may not be other others' best interest to depend on a general system, which is the consideration we will focus on. Indeed, we recall that our case *for* specialisation is focused on *external uses* of a system.

[4]Better captured by the original French expression "Travail en miettes".

Arguments from Statistics

In artificial intelligence, much of the recent interest for general AIs derives from the observed "scaling laws" (Kaplan et al. 2020a), which argue that learning from all sorts of (non-specialised) data is the driving force of dramatic performances. While these observations have been criticized (Diaz and Madaio 2024), they do have some theoretical backgrounds.

In particular, they may be argued to be an instance of Stein's paradox (Efron and Morris 1977). Strikingly, this paradox shows that, when performing statistical learning on three or more disjoint subsets of data, it is statistically *inadmissible* to learn separately on each subset. More precisely, for each subset, consider any (specialised) estimator of a ground truth statistics of the subset. Then there is a (general) estimator that learns from the union of all subsets such that, no matter what the ground truth is, the general estimator's expected mean square error will be lower than the specialised estimators'. Moreover, for at least one value of the ground truth, it is strictly lower. The general estimator is said to strictly dominate the specialised estimators.

This is especially evident in the case of collaborative filtering (Ekstrand et al. 2011). To better optimize what content should be recommended to a given user, it is extremely useful to leverage the preferences of similar users, rather than to learn exclusively from the given user's data.

While this statistical argument is very compelling, it is important to highlight two key weaknesses. First, it is a statistical, and thus *information-theoretical*, argument. It thus neglects the computational costs. In particular, the general estimator may require significantly more resources than the specialised estimators. And while we use a very computer-science terminology, this remark holds for both algorithms and human organisations. One striking example is that of science: scientists have self-organised themselves in specialised communities.

But more importantly, the subsets of data have thus far been assumed to safe to learn from. However, in practice, most datasets raise privacy and poisoning issues. On one hand, general estimator can more easily cross information that would allow for user de-anonymisation, even if "privacy-preserving learning" solutions like differential privacy are used (Kifer and Machanavajjhala 2011; Zhu et al. 2015). On the other hand, malicious users can more easily poison the union of all subsets, thereby biasing or harming the general estimator (Biggio, Nelson, and Laskov 2012; Suya et al. 2021; Farhadkhani et al. 2022). In particular, the introduction of collaborative filtering has given fake accounts an enormous influence on the daily information exposure of billions of humans.

4 Why Specialise?

Now that we have highlighted the limits of the arguments *against* specialisation, we turn our attention to our case *for* specialisation.

Large Models are More Vulnerable

In the context of artificial intelligence, generality is obtained by training large models on massive web data crawls. Indeed, ever larger models and datasets have been empirically observed to yield ever more spectacular performances (Brown et al. 2020; Kaplan et al. 2020b; Dubey et al. 2024). However, these high performances seem inevitably entangled with a number of security issues (El-Mhamdi et al. 2022; Oprea and Vassilev 2023), like privacy violations, jailbreaking and data poisoning. Below we detail the reasons of this entanglement.

The more parameters, the more vulnerabilities There is now a large literature that shows in particular that the number of parameters of a learned model increases its vulnerability, especially when it comes to privacy (Kattis and Nikolov 2017) and poisoning (El-Mhamdi et al. 2018; Hoang 2024; El-Mhamdi et al. 2017). Intuitively, this is because the leading solution for privacy, namely *differential privacy* (Dwork et al. 2006), requires adding a bit of noise to all parameters when updating the model, in order to correctly blur any trace of the data that led to this updating.

Similarly, to defend the model against malicious training data injections, the leading solutions essentially boil down to outlier removal. However, in high dimension (i.e. high number of parameters), random (honest) data are scattered away, which give poisoners a lot of room to bias the training without appearing to be out of distribution.

In the case of jailbreaking (Guo et al. 2024), a key variable is the input size of the model, also known as the *context window* for language models. Intuitively, the larger this input size, the more the model can be tuned for specific applications via input injection (prompting in the case of language models). But then, the greater the *attack surface* for jailbreaking attacks (Anil et al. 2024).

The more generality, the more data heterogeneity, the more vulnerability There is an additional subtler way through which the quest of generality harms resilience to attacks. Namely, to gain capabilities in a wide variety of use cases, the model needs to be trained on a highly heterogeneous set of training data. Such data are typically obtained through web crawls (Baack 2024), which include dubious sources (Schaul, Chen, and Tiku 2023).

In particular, under such high data heterogeneity, sensitive and malicious data are significantly harder to identify and remove. More rigorously, many papers have drawn a clear connection between machine learning vulnerability and data heterogeneity (El-Mhamdi et al. 2021).

Arguments from Complex System Engineering

A fundamental and ubiquitous principle of complex system engineering is *abstraction* (Shaw 1989). This corresponds to hiding the complexity of the system, and to highlight instead the key features of the system as well as the limited number of ways through which it can be used. Perhaps, the most important example of such an abstraction is the creation of programming languages. Such languages allow programmers to define how they may get machines to do what

they want, without having to care about the precise ways their programs will lead to transistor operations on the machines.

Abstraction is especially instrumental to implement the principles of *modularisation* and the "separation of concerns" (Parnas 1972; Tarr et al. 1999). Modularisation consists of dividing a complex systems into interacting *specialised* components, called *modules*. By carefully defining the features of the modules, which amounts to defining the abstraction they must comply with, the development of each module can then be performed independently from other modules. Similarly, each module can be evaluated, stress-tested and even correctness-proved independently from the rest of the complex system.

Separation of concerns consists of segmenting a computer program into several parts. Each of these parts is isolated and takes charge of a piece of concern or information from the general problem being dealt with. This practice simplifies the development and maintenance of computer programs. A good, strict separation of concerns means that different parts of the code can be reused or modified independently, or that work can be done on one part of the code without having to know the other parts.

Modularisation is a fundamental technique of computer science that strongly highlights the value of *specialisation*. For instance, the hashing function SHA-256 (Penard and Van Werkhoven 2008) is highly specialised. However, it is this amount of specialisation that has facilitated its thorough study for decades, and that has made it a core module of virtually all modern complex software systems.

Better yet, clearly defining the features of each module of a complex system helps identify the module's *least required privileges* (Saltzer and Schroeder 1975). Thereby, optimized security constraints can be enforced to the complex system's modules. Typically, an AI model used in inference mode should not be given access to the Internet, nor to the file systems, the webcam or the keyboard. It should only be given the input data, and the required computing hardware to perform its task. Thereby, if a given module is flawed, hacked or backdoored, then the scale of the harm to the complex system will then be limited to the privileges that were given to the module. This safety oriented mindset resembles two social organisation principles: *separation of powers* and *Subsidiarity*. Separation of powers is the foundation of constitutional states and democracies (Locke 1690), (Montesquieu 1748), and prevents the concentration of legislative, executive, and judicial powers in the hands of a single individual or political group. The separation of powers is not applied in the same way in all countries (Bellamy 2017). The stricter it is, the more distinct, specialised, and separate the powers are, while retaining reciprocal means of action. The more flexible, the more the different powers work together. Subsidiarity is associated with decentralisation. Inspired by the social doctrine of the Catholic Church, subsidiarity (Evans and Zimmermann 2014) is a political principle that assigns responsibility for a problem to the lowest competent level of public authority. This means looking for the level that is most relevant and closest to the people affected by a decision. In this way, the higher level is only called upon if the problem to be dealt with exceeds the lower level, and what can be done with the same efficiency at a lower level must be done without a higher level.

Even with limited privileges, a module may still be very dangerous. This is typically the case if its output is not carefully sanitized by the subsequent algorithmic (or human) modules. For instance, if the output of a language model is used directly for email response, then the language model can be exploited to spread a spam worm (Cohen, Bitton, and Nassi 2024). Likewise, a recommendation algorithm can threaten democracies by merely suggesting content, if human populations enact upon the hate speech that the algorithm amplifies. Tricking other modules can thus yield *privilege escalation* (Özdemir and Tuna 2024). But crucially, by carefully specialising the modules, it will be much easier to then reason about the risks of their usage.

Finally, the careful modularisation of a complex system can increase its availability by implementing the interoperable redundancy of each module, with diverse implementations of the module. The simplest example of this is the case of storage replicas. However, more inspiring systems with interoperable modules have been designed for complex tasks. Recently, the AT Protocol proposes to segment the tasks of a social media, with interoperable modules such as data storage, published message collection (know as "relays"), message labeling, feed generators and client-side applications (known as "AppViews") (Kleppmann et al. 2024). By encouraging and facilitating the creation of interoperable modules managed by other entities, the protocol helps reduce the risks of social media network effects (Dou, Niculescu, and Wu 2013). Similarly, (Hoang et al. 2024) proposes a modularisation of collaborative scoring, as a way to facilitate the construction of numerous interoperable subtasks.

All of these elements naturally bring to mind the notion of *decentralisation*, studied in various fields (computer science, political sciences, management science, law, public administration, economics, etc.) with different connotations. For example, in social sciences and particularly political sciences (Treisman 2007), it usually refers to delegation of power, transfer of skills and resources from a central power to authorities/local communities distinct from it.

Most of the notions we have mentioned (subsidiarity, decentralisation, separation of concerns and separation of powers) are sometimes used interchangeably with specialisation, and are involved in positive feedback loops with specialisation and while not delving into the interdependence between these notions and specialisation in this work, we highlight this noteworthy relationship that makes many of the arguments made in our paper at least partially valid for subsidiarity, separation of concerns, separation of powers and their alike, with maybe the notable exception of decentralisation. Decentralisation, and generally delegation of power, also offers some level of generality. For instance, components of a decentralised learning system can be learning all the same tasks, just as states in a federal system all have prerogatives on most administrative and daily life needs. Decentralisation and delegation offer advantages of their own, such as isolation of faults or confinement of corruption to preserve the

overall system.

Arguments from Economics

Specialisation Increases Competitiveness. The notion of specialisation is often associated with *division of labour*, which has even been a major theme in philosophy and economics for centuries ((Marx 1867)). In particular, division of labour should not be reduced to its technical dimension. Namely, the interdependence it generates between individuals within a society or on the scale of several has led to considering the division of labour as a important brick in the foundation of societies.

While the division and specialisation of labour within human societies can be dated back to the Neolithic period, they rather started being discussed in the 16th century. Their formalization in a systematic thinking is often associated with the writings of (Smith 1776), who explored the logic behind the fragmentation of work and the specialisation of tasks, as well as the links between division of labour and market competition. Smith used the example of a pin factory to illustrate the division of labour, with a work decomposed into eighteen operations required to produce one unit. He notes that each step of work fragmentation increases productivity. Unlike craftsmanship, where the craftsman controls the entire production process, manufacturing is superior in terms of productivity, due in particular to the simplification of tasks, the reduction of downtime, and control over the pace of work. He does, however, point out one limit to the division of labour: the size of the market. If the market is not big enough, there will be no outlet for the surplus production resulting from an ever-increasing division of labour.

While Smith focused on the specialisation of workers on a single production line, still within classical political economy, (Ricardo 1817) examined the division of labour on a global scale. With his theory of comparative advantage, Ricardo analysed the specialisation of national economies as they opened up to trade, using the famous example of England and Portugal and their production of wine and cloth. Ricardo's theory has later been further developed in numerous works on the international division of labour and international specialisation ((Samuelson 1948; Jones 1993)).

Note that, while Smith focused on human agents, Ricardo and others have already stressed the value of specialisation for non-human agents, by arguing that it increases economic competitiveness and global production. Indeed, the theory of comparative advantage asserts that, perhaps surprisingly, a less effective agent could still be helpful to a more effective agent, if the less effective agent specializes in tasks that they perform best, even in cases where the more effective agent performs better at these tasks.

Arguments from the Sociology of Work, Occupations, and Organisations

Specialisation enables intermediate bodies. Beyond the considerations of economic competitiveness, Durkheim put forward a thesis(Durkheim 1893) asserting that the division of labour is the source of the social bond in industrial societies. He even made the study of changes in the division of labour one of the foundations of sociology. The division of labour is, he says, a source of "organic solidarity" characterized by differentiation, cooperation and strong interdependencies between individuals, as opposed to mechanical forms of social organization - individuals grouped together in communities on the basis of proximity, authority or faith. Aware of working-class misery and social conflicts, Durkheim advocated in the preface to the second edition of his book (1902) for a greater application of the division of labour to social organization, through the creation of intermediate bodies constituted on a professional basis, so that they could play the role of moral authorities arbitrating social conflicts.

Such arguments apply not only to humans, but also to human organisations. By constituting intermediate bodies, such non-human agents can better defend their cause, define industry norms and share good practices to self improve. Specialisation thus helps similar non-human agents build more resilient networks.

In software development too, intermediate bodies have emerged and have been essential to define industry norms, and to increase the interoperability of various public and/or private software solutions. Some prominent examples include the *World Wide Web Consortium (W3C)*, the *Internet Engineering Task Force (IETF)* or the *Web Hypertext Application Technology Working Group (WHATWG)*, which have shaped Internet and Web protocol standards, thereby facilitating and securing the work of all companies that develop, exploit and depend on the Internet.

The (positive) social chain reaction of specialisation. By its psychological and moral dimensions (Hughes 1951), specialisation involves social interactions. Thus, specialised tasks arising from division of labour are part of extended and complex ensembles involving professionals, as well as non-professionals (Hughes 1956). Organized or not, they develop and defend (opposing) visions of what work should be. In this way, specialisation becomes part of social interactions during which legitimacy, monopolies, competition, and everything that delimits professional territories, are discussed. Specialisation is not fixed. It depends on interactions between stakeholders, and its limitations fluctuate. In the context of general AI, particularly generative AI promising to replace a considerable amount of professions and create new ones, adopting this vision of specialisation and division of labour which is not only technical, but also social, moral and psychological, would make it possible to anticipate and support emerging specialities.

A simplified lesson from company towns. In an era where AGI is increasingly becoming the proclaimed goal of AI development, and when the most powerful billionaire are talking about the "everything app"(Belanger 2024), it might be worth considering the 19th and 20th century use cases of company towns (Garner 1992). As defined in Wikipedia, "a company town is a place where all or most of the stores and housing in the town are owned by the same company that is also the main employer. Company towns are often planned with a suite of amenities such as stores, houses of worship, schools, markets, and recreation facilities". As a non-

specialised, generalist, non-human entity; or as a general intelligence to stay in the parlance of AI, company towns can be seen as dealing with needs which are not limited to the sphere of work (in this case, postal, school, health, food, etc. services). Linked to what is known as "industrial paternalism" (Noiriel 1988), this particular type of work management is associated with organisational problems. For example, the inability to provide all the services promised to workers, given that in fact, this system designed for blue-collar workers, mainly offered its advantages to white-collar workers. Another example is the difficulty of supervising workers, because it is impossible to be efficient by playing the roles of employer, priest, carer, policeman, etc., all at the same time, and therefore to bypass the division of labour and specialisation. Over and above the untenable nature of this strict social control of workers for reasons of labour profitability, this model of work organisation, based on task and power concentration, has not lasted. The reasons for the collapse of this system arguably include the replacement of the State by the management of the company town, the replacement of national law by the company practices, and most importantly, the replacement of the centuries-long process of specialisation of tasks and activities, by the one-company-do-it-all mindset of company-towns.

5 A Case for Specification

Closely related to specialisation, *specification* consists of precisely defining what a module excels at (and what it cannot perform safely). We dedicate a section to specification, as the value of specialisation is strongly tied to that of specification.

Documentations

The most straightforward step towards specification is *documentation*. To guarantee that any module or organisation is doing what it is supposed to be doing, and more importantly that it is used as it is meant to be used, it is essential to provide documents that describe the key features of the module or organisation. Most products sold in developed countries must in fact be accompanied with such a documentation.

In practice, documentation may be in conflict with the quest for ready-to-play products. In particular, viral adoption is often dependent on the ease with which the products can be used without relying on documentation. What fraction of the population has read ChatGPT's documentation? The flip side of this ease of use is unfortunately that such products are more likely to be misused, e.g. to be deployed in applications where they are ill-suited.

In the context of machine learning, increased calls for documentation have arisen, especially in the case of machine learning models through model cards (Mitchell et al. 2019) and in the case of datasets through data sheets (Gebru et al. 2021).

Type Systems and Proofs of Correctness

However, careful specification can yield much more significant improvements, especially with regards to the resilience and security of complex systems. In particular, modern programming languages provide sophisticated type systems (Cardelli 1996; Matsakis and Klock 2014; Gäher et al. 2024), which allow to directly encode the specifications of a module, and mathematically prove that the module correctly implements the specifications at compilation time. Crucially, this allows to catch bugs *before* deployment.

Moreover, type systems prevent misuses of a module, for instance by specifying the nature of the inputs that the module allows. Better yet, rich type systems can determine the guarantees that a module can provide, given some properties of the inputs that are fed to it.

Unfortunately, learning systems do not lend themselves well to such specifications, which hinders a thorough bug-catching procedure ahead of deployment. Arguably, this might be why bugs in learning systems have often been rebranded as *hallucinations* in the context of generative algorithms. But fundamentally, their ubiquity could be traced to the lack of specifications and of means to construct solutions that (provably) verify the specifications.

Now, it is noteworthy that type systems allow verification at compilation time, given the source code. In practice, an additional challenge is to verify the correctness of a program, given its compiled binary code. To meet this challenge, the fascinating domain of *verifiable computing* (Ahmad et al. 2018). In addition to multi-party computations (Garg and Srinivasan 2022), where carefully specified operations and communications enable more secure information processing, verifiable computing leverages powerful cryptographic primitives like *Succinct Non-Interactive Arguments of Knowledge* (SNARK) (Thaler et al. 2022), which enable a powerful computer to prove the soundness of its output to a weak verifier.

Recently, there has in fact been much progress towards verifiable machine-learning computing, at inference time (Fan et al. 2024a) (after deployment) and at training time (Fan et al. 2024b) (before deployment). In particular, these solutions allow to envisage the effortless enforcement of some existing laws (like EU's AI Act), e.g. by demanding that all commercialized AI systems prove that they were trained on legally obtained data (i.e. excluding copyrighted, sensitive or error-prone data). In fact, remarkably, the proofs may be constructed in *zero-knowledge*, i.e. without disclosing non-legally-binding proprietary data.

Nevertheless, verifiable computing for machine learning will remain necessarily restricted to the characteristics of a learning systems that can be humanly *specified*. While this includes important aspects (e.g. training or inference integrity), this inevitably excludes other considerations (e.g. "correct" hate speech moderation).

Hypertelia and the Pitfalls of Automated Task Specification

In biology, hypertelia (Brunner Von Wattenwyl 1874) designates an exaggerated growth of certain organs in relation to their function, to the point of making them annoying for the animal and its entire species (cf. canines of the saber-toothed tiger, too heavy antlers of the deer or disproportionate tail of the peacock). Taken up in philosophy by Simondon (Simondon 1958; Simondon, Mellamphy, and Hart 1980) for

his analysis of technical objects, the notion expresses in this context the idea of an exaggerated specialisation, and a functional over-adaptation of technical objects. Adapting an object too much to a particular purpose and context can result in its inability to function well. Tools adjusted to very specialized circumstances can lose autonomy outside of their specific technical environment. Specialisation must be guided by proven mechanisms and not be an end in itself. If guided by human choices, these can be inspired by the specialisation that has taken place in the professions. The challenge of good specification will arise more intensely when the granularity of specialisations is finer, which is enabled by automation. This is where a risk of hypertelia can occur, without being easily relieved. The challenge here is to determine who or what meta-algorithm defines the speciality of each sub-algorithm.

Two Limits of Specifications

Unfortunately, many critical information processing tasks (e.g. content recommendation, email drafting, language translation) are extremely hard to specify. We highlight more generally two reasons why not all systems can be fully specified.

First, some tasks are too complex to specify, in the sense of the Kolomogorov-Solomonoff complexity (Solomonoff 1960; Kolmogorov 1963). Formally, a formalized specification is a program that, given any program, returns whether the program verifies the specification. The Kolomogorov-Solomonoff complexity is defined as the shortest program that implements this formalized specification. Unfortunately, it is conjectured that many tasks require extremely complex specifications, in the sense that they cannot be formally described in less than a million lines of codes. As an example, EU's AI Act is 144 pages long. Yet, it is clearly far from formalized and has been argued to be very incomplete (Laux, Wachter, and Mittelstadt 2024). Another example is today's web standards, which correspond to very long documents that specify languages like HTML, CSS and ECMAScript, among others.

Second, some specifications are extremely hard to agree on. This is typically the case of content moderation, but it also holds for text autocompletion, biasless image generation or content recommendation. Given additionally the lack of knowledge about the human population's distribution of specification preferences, and perhaps even about one's own preferred specification, a choice of specification may seem premature and inappropriate. This is one heart of the challenges surrounding "AI alignment" (Hoang and El Mhamdi 2019; El-Mhamdi and Hoang 2024; Majka and El-Mhamdi 2025).

In the absence of concise and clearly consensual specifications, should systems still aim to be specified?

Specifying Unspecifiable Tasks Through Specified Governance

We stress that the specification challenge is an old problem. Throughout centuries, the correct punishment of a convicted felon has been extremely hard to specify. Nevertheless, this does not mean that we should give up on the specification effort and, e.g., abandon the choice of punishment to an all-powerful judge or dictator.

Instead of specifying the correct punishment, our societies have worked hard on specifying *how* to specify the correct punishment. This may be called the problem of the specification of the *governance* of specifications. Democracies have typically solved it by writing constitutions. which specified how any modification of the law could be enforced. Similar governance structures are arguably needed for organizations and algorithms whose goals are partially open-ended (and thus under-specified).

Remarkably, a growing line of research has provided new solutions for *algorithmic governance*. In particular, WeBuildAI (Lee et al. 2019) proposed a software which allow a number of stakeholders to collaboratively select the recipients of food donations. This software was itself subject to both informal and formal specifications, e.g. both donors, recipients, volunteers and the organising nonprofit association should all have some voting power on the decision, or they may either use the software's learning-based system to construct a model of their preferences, or write themselves a model of their preferences. This system has subsequently inspired other algorithmic governance systems, e.g. for trolley dilemmas (Noothigattu et al. 2018), kidney donation (Freedman et al. 2020) content recommendation (Hoang et al. 2021), proposal prioritization (Small et al. 2021) and contextual note selection (Righes et al. 2023). More recently, (Hoang et al. 2024) aims to clarify the challenges of the specification of the algorithmic collaborative governance of the scoring of any set of alternatives. The list of subtasks to better specify includes participant verification, trust propagation through a web of trust, preference generalization, and secure preference aggregation, among others.

6 Conclusion

In this paper, we showed the limits of the arguments against specialisation, when applied to non-human information-processing entities. We highlighted the industrial, democratic and security values of specialisation, especially when it is accompanied with careful specification. In particular, we articulated how the state of knowledge in adversarial machine learning, complex system engineering, economics, and the sociology of work, occupations and organisations all point to the numerous issues of generality. Finally, after acknowledging the limits of specifications in general, we emphasised on the importance of specifying the governance of under-specified tasks, especially when these tasks are complex or when they do not lend themselves to consensual specifications across populations and time.

We hope that the improved understanding of the value of specialisation will help researchers, developers, managers, organisations, regulators and politicians better orient the construction of a more prosperous, more secure and more sovereign information space.

Acknowledgements

The authors thank Peva Blanchard for fruitful comments.

References

Acharya, D. B.; Kuppan, K.; and Divya, B. 2025. Agentic AI: Autonomous Intelligence for Complex Goals–A Comprehensive Survey. *IEEE Access*.

Ahmad, H.; Wang, L.; Hong, H.; Li, J.; Dawood, H.; Ahmed, M.; and Yang, Y. 2018. Primitives towards verifiable computation: a survey. *Frontiers of Computer Science*, 12: 451–478.

American Medical Association. 1871. *Code of Ethics of the American Medical Association Adopted May, 1847*. Turner, Hamilton.

Anil, C.; Durmus, E.; Rimsky, N.; Sharma, M.; Benton, J.; Kundu, S.; Batson, J.; Tong, M.; Mu, J.; Ford, D. J.; et al. 2024. Many-shot jailbreaking. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Arrow, K. J. 1975. Vertical integration and communication. *The Bell Journal of Economics*, 173–183.

Autor, D. 2014. Polanyi's paradox and the shape of employment growth. Technical report, National Bureau of Economic Research.

Baack, S. 2024. A Critical Analysis of the Largest Source for Generative AI Training Data: Common Crawl. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, 2199–2208.

Belanger, A. 2024. Elon Musk's improbable path to making X an "everything app". *Ars Technica*.

Bellamy, R. 2017. *The rule of law and the separation of powers*. Routledge.

Ben-Porat, O.; Mansour, Y.; Moshkovitz, M.; and Taitler, B. 2024. Principal-Agent Reward Shaping in MDPs. In Wooldridge, M. J.; Dy, J. G.; and Natarajan, S., eds., *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, 9502–9510. AAAI Press.

Benkler, Y. 2006. *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. Yale University Press.

Bianchi, T. 2024. Google: annual advertising revenue 2001-2023. *Statista*.

Biggio, B.; Nelson, B.; and Laskov, P. 2012. Poisoning Attacks against Support Vector Machines. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress.

Blili-Hamelin, B.; Graziul, C.; Hancox-Li, L.; Hazan, H.; El-Mhamdi, E.-M.; Ghosh, A.; Heller, K. A.; Metcalf, J.; Murai, F.; Salvaggio, E.; Smart, A. J.; Snider, T.; Tighanimine, M.; Ringer, T.; Mitchell, M.; and Dori-Hacohen, S. 2025. Position: Stop treating 'AGI' as the north-star goal of AI research. In *Forty-second International Conference on Machine Learning Position Paper Track*.

Bookwalter Drury, H. 1918. *Scientific management: A history and criticism*. Columbia University Press.

Borghoff, U. M.; Bottoni, P.; and Pareschi, R. 2025. Human-artificial interaction in the age of agentic AI: a system-theoretical approach. *Frontiers in Human Dynamics*, 7: 1579166.

Boullier, D.; and El Mhamdi, E. M. 2020. Le machine learning et les sciences sociales à l'épreuve des échelles de complexité algorithmique. *Revue d'anthropologie des connaissances*, 14(14-1).

Bradford, A. 2020. *The Brussels Effect: How the European Union Rules the World*. Oxford University Press.

Bradford, A. 2024. "The Brussels Effect" and European Sovereignty. In *European Sovereignty: The Legal Dimension–A Union in Control of its own Destiny*, 191–200. Springer.

Brennan, T. J. 2025. US government antitrust Google and Facebook cases: three neglected questions. In *Research Handbook On Competition And Technology*, 212–231. Edward Elgar Publishing.

Bresnahan, T. F.; and Levin, J. D. 2012. *Vertical integration and market structure*. National Bureau of Economic Research Cambridge (MA).

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Brunner Von Wattenwyl, K. 1874. *Ueber die Hypertelie in der Natur.* Wien.

Brynjolfsson, E.; and Mitchell, T. 2017. What can machine learning do? Workforce implications. *Science*, 358(6370): 1530–1534.

Cardelli, L. 1996. Type systems. *ACM Computing Surveys (CSUR)*, 28(1): 263–264.

Carlton, D. W. 1979. Vertical integration in competitive markets under uncertainty. *The Journal of Industrial Economics*, 189–209.

Casilli, A. A. 2020. From the virtual class to the click workers: the transformation of work into service in the era of digital platforms. *MATRIZes*, 14(1): 13–21.

Church, A. 1936. A note on the Entscheidungsproblem. *The journal of symbolic logic*, 1(1): 40–41.

Coase, R. H. 1937. The Nature of the Firm. *Economica*.

Cohen, S.; Bitton, R.; and Nassi, B. 2024. Here Comes The AI Worm: Unleashing Zero-click Worms that Target GenAI-Powered Applications. *CoRR*, abs/2403.02817.

Davies, T.; and Georgieva, Z. 2024. Google AdTech: Break Up or Break Out? *Utrecht Law Journal Special Issue on Modern Bigness*.

Diaz, F.; and Madaio, M. 2024. Scaling laws do not scale. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 7, 341–357.

Dohse, K.; Jürgens, U.; and Nialsch, T. 1985. From" Fordism" to" Toyotism"? The social organization of the labor process in the Japanese automobile industry. *Politics & Society*, 14(2): 115–146.

Dou, Y.; Niculescu, M. F.; and Wu, D. 2013. Engineering optimal network effects via social media features and seeding in markets for digital goods and services. *Information Systems Research*, 24(1): 164–185.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; Goyal, A.; Hartshorn, A.; Yang, A.; Mitra, A.; Sravankumar, A.; Korenev, A.; Hinsvark, A.; Rao, A.; Zhang, A.; Rodriguez, A.; Gregerson, A.; Spataru, A.; Rozière, B.; Biron, B.; Tang, B.; Chern, B.; Caucheteux, C.; Nayak, C.; Bi, C.; Marra, C.; McConnell, C.; Keller, C.; Touret, C.; Wu, C.; Wong, C.; Ferrer, C. C.; Nikolaidis, C.; Allonsius, D.; Song, D.; Pintz, D.; Livshits, D.; Esiobu, D.; Choudhary, D.; Mahajan, D.; Garcia-Olano, D.; Perino, D.; Hupkes, D.; Lakomkin, E.; AlBadawy, E.; Lobanova, E.; Dinan, E.; Smith, E. M.; Radenovic, F.; Zhang, F.; Synnaeve, G.; Lee, G.; Anderson, G. L.; Nail, G.; Mialon, G.; Pang, G.; Cucurell, G.; Nguyen, H.; Korevaar, H.; Xu, H.; Touvron, H.; Zarov, I.; Ibarra, I. A.; Kloumann, I. M.; Misra, I.; Evtimov, I.; Copet, J.; Lee, J.; Geffert, J.; Vranes, J.; Park, J.; Mahadeokar, J.; Shah, J.; van der Linde, J.; Billock, J.; Hong, J.; Lee, J.; Fu, J.; Chi, J.; Huang, J.; Liu, J.; Wang, J.; Yu, J.; Bitton, J.; Spisak, J.; Park, J.; Rocca, J.; Johnstun, J.; Saxe, J.; Jia, J.; Alwala, K. V.; Upasani, K.; Plawiak, K.; Li, K.; Heafield, K.; Stone, K.; and et al. 2024. The Llama 3 Herd of Models. *CoRR*, abs/2407.21783.

Durkheim, É. 1893. *De la division du travail*. FB Editions.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. D. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography, Third Theory of Cryptography Conference, TCC*.

Economides, N. 2001. The Microsoft antitrust case. *Journal of Industry, Competition and Trade*, 1: 7–39.

Efron, B.; and Morris, C. 1977. Stein's paradox in statistics. *Scientific American*, 236(5): 119–127.

Ekstrand, M. D.; Riedl, J. T.; Konstan, J. A.; et al. 2011. Collaborative filtering recommender systems. *Foundations and Trends® in Human–Computer Interaction*, 4(2): 81–173.

El-Mhamdi, E. M.; Farhadkhani, S.; Guerraoui, R.; Guirguis, A.; Hoang, L.-N.; and Rouault, S. 2021. Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning). *Advances in neural information processing systems*, 34: 25044–25057.

El-Mhamdi, E.-M.; Farhadkhani, S.; Guerraoui, R.; Gupta, N.; Hoang, L.-N.; Pinot, R.; Rouault, S.; and Stephan, J. 2022. On the impossible safety of large AI models. *arXiv preprint arXiv:2209.15259*.

El-Mhamdi, E.-M.; Guerraoui, R.; Rouault, S.; et al. 2017. On The Robustness of a Neural Network. In *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*, 84–93. IEEE.

El-Mhamdi, E.-M.; Guerraoui, R.; Rouault, S.; et al. 2018. The Hidden Vulnerability of Distributed Learning in Byzantium. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*.

El-Mhamdi, E.-M.; and Hoang, L.-N. 2024. On Goodhart's law, with an application to value alignment. *arXiv preprint arXiv:2410.09638*.

Evans, M.; and Zimmermann, A. 2014. The global relevance of subsidiarity: An overview. *Global perspectives on subsidiarity*, 1–7.

Fan, Y.; Ma, K.; Zhang, L.; Lei, X.; Xu, G.; and Tan, G. 2024a. ValidCNN: A large-scale CNN predictive integrity verification scheme based on zk-SNARK. *IEEE Transactions on Dependable and Secure Computing*.

Fan, Y.; Ma, K.; Zhang, L.; Liu, J.; Xiong, N.; and Yu, S. 2024b. VeriCNN: Integrity verification of large-scale CNN training process based on zk-SNARK. *Expert Systems with Applications*, 124531.

Farhadkhani, S.; Guerraoui, R.; Hoang, L. N.; and Villemaud, O. 2022. An Equivalence Between Data Poisoning and Byzantine Gradient Attacks. In *International Conference on Machine Learning, ICML*.

Fetz, A.; and Filippini, M. 2010. Economies of vertical integration in the Swiss electricity sector. *Energy economics*, 32(6): 1325–1330.

Freedman, R.; Borg, J. S.; Sinnott-Armstrong, W.; Dickerson, J. P.; and Conitzer, V. 2020. Adapting a kidney exchange algorithm to align with human values. *Artificial Intelligence*, 283: 103261.

Friedman, G. 1961. *The anatomy of work: labor, leisure and the implications of automation*. Transaction Publishers.

Friedmann, G. 1955. Le Travail en miettes. *Esprit (1940-)*, 1725–1747.

Gäher, L.; Sammler, M.; Jung, R.; Krebbers, R.; and Dreyer, D. 2024. Refinedrust: A type system for high-assurance verification of Rust programs. *Proceedings of the ACM on Programming Languages*, 8(PLDI): 1115–1139.

Garg, S.; and Srinivasan, A. 2022. Two-round multiparty secure computation from minimal assumptions. *Journal of the ACM*, 69(5): 1–30.

Garner, J. 1992. *The company town: architecture and society in the early industrial age*. Oxford University Press.

Gebru, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J. W.; Wallach, H.; Iii, H. D.; and Crawford, K. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92.

Gibert, M.; and Martin, D. 2022. In search of the moral status of AI: why sentience is a strong argument. *AI & SOCIETY*, 37(1): 319–330.

Grossman, S. J.; and Hart, O. D. 1986. The costs and benefits of ownership: A theory of vertical and lateral integration. *Journal of political economy*, 94(4): 691–719.

Gunst, S.; and De Ville, F. 2021. The Brussels effect: how the GDPR conquered Silicon Valley. *European Foreign Affairs Review*, 26(3).

Guo, X.; Yu, F.; Zhang, H.; Qin, L.; and Hu, B. 2024. COLD-Attack: Jailbreaking LLMs with Stealthiness and Controllability. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Hackman, J. R. 1969. NATURE OF THE TASK AS A DETERMINER OF JOB BEHAVIOR. *Personnel Psychology*, 22(4).

Hoang, L.-N. 2024. The poison of dimensionality. *arXiv preprint arXiv:2409.17328*.

Hoang, L. N.; Beylerian, R.; Colbois, B.; Fageot, J.; Faucon, L.; Jungo, A.; Noac'h, A. L.; Matissart, A.; and Villemaud, O. 2024. Solidago: A Modular Collaborative Scoring Pipeline. *CoRR*, abs/2211.01179.

Hoang, L. N.; and El Mhamdi, E. M. 2019. *Le fabuleux chantier: Rendre l'intelligence artificielle robustement bénéfique*. edp Sciences.

Hoang, L.-N.; Faucon, L.; Jungo, A.; Volodin, S.; Papuc, D.; Liossatos, O.; Crulis, B.; Tighanimine, M.; Constantin, I.; Kucherenko, A.; et al. 2021. Tournesol: A quest for a large, secure and trustworthy database of reliable human judgments. *arXiv preprint arXiv:2107.07334*.

Hou, X.; Zhao, Y.; Wang, S.; and Wang, H. 2025. Model context protocol (mcp): Landscape, security threats, and future research directions. *arXiv preprint arXiv:2503.23278*.

Hughes, E. C. 1951. Mistakes at work. *Canadian Journal of Economics and Political Science/Revue canadienne de economiques et science politique*, 17(3): 320–327.

Hughes, E. C. 1956. Social role and the division of labor. *The Midwest Sociologist*, 18(2): 3–7.

Hutter, M. 2003. A gentle introduction to the universal algorithmic agent AIXI.

Jones, R. 1993. Heckscher-Ohlin trade theory: Harry Flam and M. June Flanders, eds., (The MIT Press, Cambridge, MA, 1991) pp. x + 222. *Journal of International Economics*, 35(1-2): 197–199.

Kakabadse, A.; and Kakabadse, N. 2005. Outsourcing: current and future trends. *Thunderbird international business review*, 47(2): 183–204.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020a. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020b. Scaling Laws for Neural Language Models. *CoRR*, abs/2001.08361.

Kattis, A.; and Nikolov, A. 2017. Lower Bounds for Differential Privacy from Gaussian Width. In *33rd International Symposium on Computational Geometry, SoCG 2017, July 4-7, 2017, Brisbane, Australia*, volume 77 of *LIPIcs*, 45:1–45:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Kierans, A.; Ghosh, A.; Hazan, H.; and Dori-Hacohen, S. 2025. Quantifying Misalignment Between Agents: Towards a Sociotechnical Understanding of Alignment. In Walsh, T.; Shah, J.; and Kolter, Z., eds., *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, 27365–27373. AAAI Press.

Kifer, D.; and Machanavajjhala, A. 2011. No free lunch in data privacy. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*, 193–204. ACM.

Kleppmann, M.; Frazee, P.; Gold, J.; Graber, J.; Holmgren, D.; Ivy, D.; Johnson, J.; Newbold, B.; and Volpert, J. 2024. Bluesky and the at protocol: Usable decentralized social media. In *Proceedings of the ACM Conext-2024 Workshop on the Decentralization of the Internet*, 1–7.

Kolmogorov, A. N. 1963. On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A*, 369–376.

Landes, W. M.; and Posner, R. A. 1997. Market power in antitrust cases. *J. Reprints Antitrust L. & Econ.*, 27: 493.

Laux, J.; Wachter, S.; and Mittelstadt, B. 2024. Trustworthy artificial intelligence and the European Union AI act: On the conflation of trustworthiness and acceptability of risk. *Regulation & Governance*, 18(1): 3–32.

Lee, M. K.; Kusbit, D.; Kahng, A.; Kim, J. T.; Yuan, X.; Chan, A.; See, D.; Noothigattu, R.; Lee, S.; Psomas, A.; et al. 2019. WeBuildAI: Participatory framework for algorithmic governance. *Proceedings of the ACM on human-computer interaction*, 3(CSCW): 1–35.

Li, Z.; and Agarwal, A. 2017. Platform integration and demand spillovers in complementary markets: Evidence from Facebook's integration of Instagram. *Management Science*, 63(10): 3438–3458.

Lian, X.; Yuan, B.; Zhu, X.; Wang, Y.; He, Y.; Wu, H.; Sun, L.; Lyu, H.; Liu, C.; Dong, X.; Liao, Y.; Luo, M.; Zhang, C.; Xie, J.; Li, H.; Chen, L.; Huang, R.; Lin, J.; Shu, C.; Qiu, X.; Liu, Z.; Kong, D.; Yuan, L.; Yu, H.; Yang, S.; Zhang, C.; and Liu, J. 2022. Persia: An Open, Hybrid System Scaling Deep Learning-based Recommenders up to 100 Trillion Parameters. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, 3288–3298. ACM.

Lieberman, M. B. 1991. Determinants of vertical integration: An empirical test. In *Academy of Management Proceedings*, 31–35. Academy of Management Briarcliff Manor, NY 10510.

Littler, C. R. 1978. Understanding taylorism. *British Journal of Sociology*, 185–202.

Locke, J. 1690. *Two Treatises of Government*. Awnsham Churchill.

Loukidou, L.; Loan-Clarke, J.; and Daniels, K. 2009. Boredom in the workplace: More than monotonous tasks. *International Journal of Management Reviews*.

Majka, A.; and El-Mhamdi, E.-M. 2025. The Strong, Weak and Benign Goodhart's law. An independence-free and paradigm-agnostic formalisation. *arXiv preprint arXiv:2505.23445*.

Marx, K. 1847. *Misère de la philosophie*. C.G. Vogler, Bruxelles.

Marx, K. 1867. *Das Kapital*. Hamburg: Otto Meissner, 1867; New-York: L. W. Schmidt.

Matsakis, N. D.; and Klock, F. S. 2014. The rust language. *ACM SIGAda Ada Letters*, 34(3): 103–104.

Maxwell, T. 2024. Leaked Documents Show OpenAI Has a Very Clear Definition of 'AGI'. *Gizmodo*.

McCauley, C. D.; Ruderman, M. N.; Ohlott, P. J.; and Morrow, J. E. 1994. Assessing the developmental components of managerial jobs. *Journal of applied psychology*, 79(4): 544.

Ministère de la santé. 2004 (initial version 1953). Article R4127-70. *Code de la santé publique*.

Mitchell, M.; Wu, S.; Zaldivar, A.; Barnes, P.; Vasserman, L.; Hutchinson, B.; Spitzer, E.; Raji, I. D.; and Gebru, T. 2019. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, 220–229.

Montesquieu. 1748. *De l'esprit des lois*. Barrillot & Fils.

Munir, S.; Kollnig, K.; Shuba, A.; and Shafiq, Z. 2024. Google's Chrome Antitrust Paradox. *arXiv preprint arXiv:2406.11856*.

Murugesan, S. 2025. The Rise of Agentic AI: Implications, Concerns, and the Path Forward. *IEEE Intelligent Systems*, 40(2): 8–14.

Noiriel, G. 1988. Du" patronage" au" paternalisme": la restructuration des formes de domination de la main-d'œuvre ouvrière dans l'industrie métallurgique française. *Le mouvement social*, 17–35.

Noothigattu, R.; Gaikwad, S. N. S.; Awad, E.; Dsouza, S.; Rahwan, I.; Ravikumar, P.; and Procaccia, A. D. 2018. A Voting-Based System for Ethical Decision Making. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, 1587–1594. AAAI Press.

Oprea, A.; and Vassilev, A. 2023. Adversarial machine learning: A taxonomy and terminology of attacks and mitigations. Technical report, National Institute of Standards and Technology.

Özdemir, G.; and Tuna, G. 2024. Privilege Escalation: Threats, Prevention, and a Case Study. In *Cases on Forensic and Criminological Science for Criminal Detection and Avoidance*, 151–187. IGI Global.

Parnas, D. L. 1972. On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15(12): 1053–1058.

Penard, W.; and Van Werkhoven, T. 2008. On the secure hash algorithm family. *Cryptography in context*, 1–18.

Perrigo, B. 2023. Exclusive: OpenAI Lobbied the EU to Water Down AI Regulation. *Time*.

Polanyi, M. 2009. The tacit dimension. In *Knowledge in organisations*, 135–146. Routledge.

Radosevich, B.; and Halloran, J. 2025. MCP Safety Audit: LLMs with the Model Context Protocol Allow Major Security Exploits. *arXiv preprint arXiv:2504.03767*.

Ray, P. P. 2025. A survey on model context protocol: Architecture, state-of-the-art, challenges and future directions. *Authorea Preprints*.

Ricardo, D. 1817. *On the Principles of Political Economy and Taxation*. McMaster University Archive for the History of Economic Thought, 3 edition.

Righes, L.; Saeed, M.; Demartini, G.; and Papotti, P. 2023. The Community Notes Observatory: Can Crowdsourced Fact-Checking be Trusted in Practice? In *Companion Proceedings of the ACM Web Conference 2023*, 172–175.

Saltzer, J. H.; and Schroeder, M. D. 1975. The protection of information in computer systems. *Proceedings of the IEEE*, 63(9): 1278–1308.

Samuelson, P. A. 1948. International trade and the equalisation of factor prices. *The Economic Journal*, 58(230): 163–184.

Schaul, K.; Chen, S. Y.; and Tiku, N. 2023. Inside the secret list of websites that make AI like ChatGPT sound smart. *Washington Post*, 19.

Shapiro, C. 1999. *Information rules: A strategic guide to the network economy*. Harvard Business School Press.

Shaw, M. 1989. Larger scale systems require higher-level abstractions. In *Proceedings of the 5th International Workshop on Software Specification and Design, IWSSD 1989, Pittsburgh, Pennsylvania, USA, 1989*, 143–146. ACM.

Siegmann, C.; and Anderljung, M. 2022. The Brussels effect and artificial intelligence: How EU regulation will impact the global AI market. *arXiv preprint arXiv:2208.12645*.

Silvers, R.; and Alperovitch, D. 2024. Review of the Summer 2023 Microsoft Exchange Online Intrusion. Technical report, Cybersecurity and Infrastructure Security Agency.

Simondon, G. 1958. *Du mode d'existence des objets techniques*. Éditions Aubier-Montaigne.

Simondon, G.; Mellamphy, N.; and Hart, J. 1980. *On the mode of existence of technical objects*. University of Western Ontario London.

Skendzic, A.; and Kovacic, B. 2012. Microsoft office 365-cloud in business environment. In *2012 Proceedings of the 35th International Convention MIPRO*, 1434–1439. IEEE.

Small, C.; Bjorkegren, M.; Erkkilä, T.; Shaw, L.; and Megill, C. 2021. Polis: Scaling deliberation by mapping high dimensional opinion spaces. *Recerca: revista de pensament i anàlisi*, 26(2).

Smith, A. 1776. *An Inquiry into the Nature and Causes of the Wealth of Nations*. McMaster University Archive for the History of Economic Thought.

Solomonoff, R. J. 1960. A preliminary report on a general theory of inductive inference.

Suya, F.; Mahloujifar, S.; Suri, A.; Evans, D.; and Tian, Y. 2021. Model-Targeted Poisoning Attacks with Provable Convergence. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July*

2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, 10000–10010. PMLR.

Tarr, P.; Ossher, H.; Harrison, W.; and Sutton Jr, S. M. 1999. N degrees of separation: Multi-dimensional separation of concerns. In *Proceedings of the 21st international conference on Software engineering*, 107–119.

Thaler, J.; et al. 2022. Proofs, arguments, and zero-knowledge. *Foundations and Trends® in Privacy and Security*, 4(2–4): 117–660.

Tighanimine, M. 2025. Le travail journalistique entre priorisation de l'information et réaction aux algorithmes de recommandation des réseaux sociaux. *Socio. La nouvelle revue des sciences sociales*, (20): 127–158.

Treisman, D. 2007. *The Architecture of Government: Rethinking Political Decentralization*. Cambridge University Press.

Turing, A. M. 1936. On computable numbers, with an application to the Entscheidungsproblem. *J. of Math*, 58(345-363): 5.

Ungarino, R. 2020. Here are 9 fascinating facts to know about BlackRock, the world's largest asset manager. *Business Insider*.

Veness, J.; Ng, K. S.; Hutter, M.; and Silver, D. 2010. Reinforcement learning via AIXI approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, 605–611.

Vergara, R. A. G.; and Del Rosario, R. 2012. Samsung Electronics and Apple, Inc.: A Study in Contrast in Vertical Integration in the 21 st Century. *American International Journal of Contemporary Research*, 2(9): 77–81.

Watson, D. 2019. Fordism: A review essay. *Labor History*, 60(2): 144–159.

Williams, A.; Miceli, M.; and Gebru, T. 2022. The exploited labor behind artificial intelligence. *Noema Magazine*, 22.

Wörsdörfer, M. 2024. Apple's antitrust paradox. *European Competition Journal*, 20(1): 113–146.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Zhu, T.; Xiong, P.; Li, G.; and Zhou, W. 2015. Correlated Differential Privacy: Hiding Information in Non-IID Data Set. *IEEE Trans. Inf. Forensics Secur.*, 10(2): 229–242.