

Cognición computacional

Computational Cognition

JAVIER BLANCO¹

Universidad Nacional de Córdoba, Argentina

javier.blanco@unc.edu.ar

Fecha de recepción: 29/04/2024

Fecha de aceptación: 27/06/2024

Resumen

En este trabajo sostendremos una interrogación a los sistemas computacionales y a la noción misma de computación desde perspectivas conceptuales, políticas y culturales. La ubicuidad y la insoslayable y creciente importancia que estos sistemas tienen en todos los ámbitos de la vida suele llevar a análisis apresurados y distópicos de sus alcances y límites, considerando algunas consecuencias propias de un estadio particular de su desarrollo como constitutivas o inevitables. Tratando de desenredar una vieja, pero renovada trama de confusiones, tomaremos algunas ideas y descubrimientos fundacionales de la computación como la efectividad, la programabilidad y la abstracción, los programas como datos y el interjuego entre manipulación humana y mecánica de símbolos propuesta por Dijkstra, como claves

¹ PhD. en ciencias de la computación, otorgado por la Universidad de Eindhoven, Países Bajos. Profesor Titular con dedicación exclusiva en la Facultad de Matemática, Astronomía, Física y Computación. Director de la Maestría en Tecnología, Políticas y Culturas, dependiente de la Facultad de Ciencias Sociales, de la Facultad de Filosofía y Humanidades y de la Facultad de Artes. Director Alterno del Doctorado en Ciencias de la Computación, Facultad de Matemática, Astronomía, Física y Computación. <https://orcid.org/0000-0003-4416-7892>

CÓMO CITAR ESTE ARTÍCULO:

En APA: Blanco, J. (2024). Cognición computacional. *Resonancias*, (17), 143-155. DOI: 10.5354/0719-790X.2024.74507

En MLA: Blanco, J. "Cognición computacional". *Resonancias*, n.º 17, julio de 2024, pp. 143-155. DOI: 10.5354/0719-790X.2024.74507

Palabras clave: computación, inteligencia artificial, metaprogramación, cognición, Coevolución

Keywords: Computation, Artificial, Intelligence, Metaprogramming, Cognition, Co-evolution

para construir una mirada que se sustraiga a los frecuentes prejuicios antropomórficos y que habilite pensar un horizonte de mejores entramados entre humanos y algoritmos. Para ello presentaremos tres niveles de análisis de sistemas computacionales que dan cuenta de propiedades diferentes en cada nivel de manera de poder sopesar mejor no solo la actualidad de la IA, sino también las posibilidades de evolución de las mediaciones computacionales en general.

Abstract

In this work, we interrogate computational systems and the very notion of computing from conceptual, political and cultural perspectives. The ubiquity and the unavoidable growing importance of these systems in all areas of life, usually lead to hasty and dystopian analyzes of their scope and limits. Some consequences derived from this particular stage of development are frequently considered ineluctable. In order to unravel an old but renewed web of confusions, we will consider some founding ideas and discoveries from computer science, such as effectiveness, programability and abstraction, programs as data and the interplay between human and mechanic symbol manipulation proposed by Dijkstra as keys to build a perspective that avoids anthropomorphic prejudices and enables us to think about an horizon with better couplings between humans and algorithms. To do this, we will present three levels of analysis of computational systems that account for different properties at each level in order to assess not only the current state of the art of AI but also the possibilities of evolution of computational mediations in general.

“For anyone wanting a precise analysis of what we refer to today as artificial intelligence, which seems now to have become the horizon of everyday life (and I will return to this question), it is necessary to begin from the following postulate: all noetic intelligence is artificial.” (Stiegler 2018)

1. La inteligencia artificial, no (siempre) inteligente, no (solo) artificial

El sintagma “inteligencia artificial”, antes que una definición, tiene una historia no tan larga, pero sí muy intensa. Consecuentemente, no aspiramos aquí a formular una definición precisa, sino que abordaremos de manera crítica algunas de sus caracterizaciones, sobre todo, en los campos de las humanidades y de las ciencias sociales, en busca de formular indicaciones conceptuales y terminológicas que faciliten la comprensión del alcance de estas nuevas entidades en la vida social y cultural. Una de las dimensiones que consideraremos es entenderla como una nueva forma de desarrollo de software, con diferencias importantes con la mayoría de los programas que se desarrollaron en los últimos 70 años. Otra tiene que ver con las maneras de comprender los vínculos diversos entre

humanos y sistemas computacionales y sus posibilidades actuales de evolución. En este sentido, consideramos que las metáforas antropomórficas son confusas e inmotivadas y que obturan un análisis certero de estas dos dimensiones. Tratar de establecer paralelismos entre la inteligencia artificial y la humana (que como nos recuerda Stiegler siempre fue artificial, al menos en su versión noética) aparece como un camino poco promisorio. Por un lado, no es tan claro cómo funciona la inteligencia humana, por lo que parece poco propicio considerarla como parámetro con el cual contrastar los desarrollos de software de IA. En cualquier caso, es casi una obviedad que los funcionamientos son disímiles, al punto que los fundadores de la IA han sido muy cautos, hablando, por ejemplo, de programas que realizan tareas para las que un humano necesitaría inteligencia para poder realizar.

En el interesante libro *Atlas of AI*, Kate Crawford (2021) instala una celebrada frase: “la inteligencia artificial no es artificial ni es inteligente”. Crawford argumenta que los sistemas de IA no son inteligentes ya que carecen de autonomía, racionalidad y discernimiento sin que medie un entrenamiento computacional extensivo e intensivo con grandes bases de datos o reglas y recompensas predefinidas. Todos estos argumentos son válidos, sobre todo, si nos referimos a este momento de la IA, donde los programas producidos a través de alguna variante de *machine learning* son reconocedores estadísticos de patrones, más asociados a una extensión de la percepción (ver más abajo la idea del nooscopio) que a lo que en los humanos es considerado inteligente.

Todos estos argumentos son correctos y, en algún sentido, dan cuenta de una obviedad. Es también notorio cómo esta frase se repite en los debates sobre IA como si fuera una revelación y refleja cierto alivio. Aparentemente, reaparece aquí un recurrente prejuicio humanista que se asienta en suponer que la inteligencia humana es, en cierto sentido, paradigmática y que, por lo tanto, el grado de inteligencia de cualquier entidad se mediría por su cercanía o parecido con ella. No hay buenos argumentos para sustentar esto, y parece reflejar una mirada autocomplaciente de los humanos, poco crítica con los obvios límites y dificultades del pensamiento humano.

Para negar la artificialidad de la IA, Crawford afirma que esta es corporizada (*embodied*) y material. Se argumenta también (al igual que Pasquinelli (2023)) que gran parte del esfuerzo para producir los programas usando machine learning proviene de trabajo humano de clasificación y etiquetado de datos, en general mal remunerado, indispensable para llevar a cabo el proceso de entrenamiento. También eso es cierto y frecuente, aunque no es necesario, y hay muchos sistemas que se desarrollan con datos sintéticos. De cualquier manera, tanto el trabajo humano como la explotación de recursos materiales están, desde siempre, asociados a la producción de lo artificial.

Por diferentes motivos, William Rapaport (2023) también critica, por confuso, el sintagma “inteligencia artificial”. Por un lado, plantea que la idea de inteligencia no puede ser definida con precisión y que en muchos trabajos fundacionales (por ejemplo, de Minsky y Simon) se evitaba cualquier definición, asumiendo que el proyecto era hacer máquinas (o programas) que pudieran hacer cosas que requerirían inteligencia si fueran hechas por humanos. Estas tareas, argumenta Rapaport, quedan mejor enmarcadas bajo el término de ‘cognición’ en lugar de ‘inteligencia’, ya que incluyen estados mentales y procesos como creencias, conciencia, emociones, lenguaje, memoria, percepción, planificación y resolución de problemas.

Por otro lado, la idea de “artificial” tiene también para él varios problemas. El que enfatiza es que la palabra remite a algo poco legítimo, sin realidad propia, a un simulacro de, en este caso, la inteligencia real. En cualquier caso, también puede considerarse que la inteligencia humana es esencialmente artificial, constituida además gracias a la multiplicidad de andamiajes externos que la hacen posible, siendo el lenguaje uno de los fundamentales. Esta artificialidad constitutiva de la inteligencia humana habilitaría, siguiendo a Stiegler, también la producción de estupidez, a partir de la industrialización de órganos exosomáticos constitutivos de los andamiajes cognitivos. Estamos hoy en medio de ese proceso cuya futura evolución es aún incierta y en disputa.

Ahora bien, cuando nos referimos a los sistemas computacionales que hoy constituyen la llamada IA, Rapaport prefiere hablar de “inteligencia (o cognición) sintética” o, más específicamente indicando la naturaleza de esos procesos de síntesis, “cognición computacional” (Rapaport 1354). La mayor generalidad de la palabra “cognición” y la mayor especificidad de “computacional” parecen delimitar con mucha más precisión esta noción elusiva. En lo que sigue, seguiremos usando el acrónimo IA, pero lo entenderemos como refiriendo a la cognición computacional. También esta especificidad permitiría entender mejor la crítica de Crawford esgrimida contra la “artificialidad”; tiene más sentido si se quiere indicar que al IA no es solo computacional.

2. Multiescalaridad de las tecnologías computacionales

“From a bit to a few hundred megabytes, from a microsecond to a half an hour of computing confronts us with completely baffling ratio of 10^9 ! The programmer is in the unique position that his is the only discipline and profession in which such a gigantic ratio, which totally baffles our imagination, has to be bridged by a single technology. He has to be able to think in terms of conceptual hierarchies that are much deeper than a single mind ever needed to face before. Compared to that number of semantic levels, the average mathematical theory is almost flat. By evoking the need for deep conceptual hierarchies, the automatic computer confronts us with a radically new intellectual challenge that has no precedent in our history.” (Dijkstra 1988)

Entender las características de los sistemas de lo que se llama hoy IA requiere distinguir, al menos, tres tipos de registros con grados de generalidad decreciente que suelen confundirse en algunos análisis. En esta sección abordaremos el registro más general desde el que daremos cuenta, sucintamente, de toda mediación computacional.

En el registro más general nos encontramos con las propiedades compartidas por todos los sistemas computacionales, que son los que han transformado drásticamente, durante ya casi un siglo, la escena intelectual, presentando desafíos inéditos, nuevas formas de conocimiento, y han hecho posible formas de acción diferida de complejidad creciente. En los últimos años, los indiscutibles logros de la tecnología de machine learning han revitalizado el rol de las mediaciones computacionales en la estructuración del mundo. Sin embargo, en la mayoría de los trabajos y proyectos actuales suele asumirse que es el desarrollo de la IA la causa principal de esa renovada relevancia, y que los programas generados por machine learning representan el apogeo de las tecnologías computacionales. Contrario a esto, consideramos que la profunda transformación en curso de la condición humana está impulsada por la novedad radical asociada a la capacidad de programar comportamientos de manera precisa y efectiva, y por la -literalmente- infinita variedad y versatilidad del mundo algorítmico, variedad que no necesariamente tienen los programas producidos por machine learning.

Como bien indica Benjamin Bratton (2015), el mundo actual está constituido por una pila de capas digitales que habilitan una multiplicidad de operaciones. Extrapolando ideas de la cibernética, consideramos estas capas como nuevas dimensiones de la materialidad del mundo, que permiten múltiples formas de percepción, de cognición y de acción. La llamada “digitalización” no es una reducción o simplificación del inabarcable flujo de la realidad, sino que constituye una serie de capas superpuestas sobre este *continuum*, y cuya razón de exis-

tencia es habilitar nuevas formas de operación computacional, que, desafiando ciertas intuiciones, multiplican las capacidades y posibilidades. Es una obviedad que las mediaciones computacionales están transformando drásticamente la existencia, aumentando su complejidad, viendo a diario la aparición de nuevas herramientas o procesos de transformación en todas las esferas del mundo de la vida. Pese a esto, muchos autores siguen insistiendo en que lo digital implica una reducción de la realidad o de las personas a “meros datos”, que pierden su singularidad convirtiéndose en objetos de cálculo. Como veremos más adelante, es mucho más fértil conceptualmente asumir esta “datificación” y las consecuentes transformaciones de lo real como una enorme oportunidad evolutiva.

Es importante poner en foco este registro general de análisis de las mediaciones computacionales para evitar una frecuente falacia de composición que consiste en tomar propiedades específicas de algún subconjunto determinado; en particular a los sistemas producidos por machine learning, como aplicables a cualquier sistema computacional. Esta falacia tiene consecuencias conceptuales bastantes insidiosas, ya que este tipo de tecnología produce programas muy diferentes a los usuales, al punto que algunos autores no los reconocen *a priori* como tales (Janlert 2008), que no tienen nada que ver con la mayoría de los sistemas existentes o posibles. Para sumar a la confusión, la palabra “algoritmo” ha tomado cierto uso social asociado a los programas producidos por machine learning que, en esencia, son clasificadores (o generadores) estadísticos, muy diferentes de lo que se considera paradigmáticamente un algoritmo. Vamos a remarcar esto con cierta insistencia en aras de escapar a lo que termina siendo una mirada determinista o incluso tecnofóbica frente a la acelerada evolución de las mediaciones algorítmicas.

Otra característica fundamental de los sistemas computacionales se deriva de un hecho conocido desde los trabajos fundacionales de Turing y que fue consistentemente desarrollado y explotado por la teoría de la computación: los programas son también datos. Esta propiedad tiene muchísimas consecuencias, siendo fundamental también en el desarrollo de la práctica de la programación. Existen múltiples tipos de metaprogramas (es decir, programas que toman a otros programas como datos) indispensables para el desarrollo de las tecnologías computacionales. Compiladores, intérpretes, máquinas virtuales, sistemas operativos, *shells* de comandos, macroexpansiones, super-compiladores, son algunos de los ejemplos de metaprogramas de uso cotidiano. También los clasificadores estadísticos producidos por la tecnología de machine learning son el *output* de un meta-programa de entrenamiento. La meta-programación es la regla, no la excepción.

Esto permite comprender a la tecnología computacional como una metatecnología, es decir, como un espacio de producción de diversas tecnologías, cada una

con sus especificidades. De hecho, casi todas las formas técnicas previamente existentes han sido transformadas por la inclusión de subsistemas computacionales como parte de los objetos o ambientes técnicos. Un auto, un televisor, una lavadora o una afeitadora eléctrica, cada vez más, funcionan gracias a crecientes cantidades de software integrado en ellos.

Otra de las características salientes de la tecnología computacional es la marcada por Dijkstra en el epígrafe: la cantidad, profundidad y complejidad de los niveles de abstracción involucrados. Las relaciones intranivel e internivel son múltiples y generalmente se realizan también como programas. Desde esta perspectiva puede apreciarse otra de las características novedosas de los sistemas computacionales: la capacidad de vincular diferentes órdenes de magnitud.

3. ¿Qué podemos aprender de las máquinas que aprenden?

“The Fathers of the field had been pretty confusing: John von Neumann speculated about computers and the human brain in analogies sufficiently wild to be worthy of a medieval thinker and Alan M. Turing thought about criteria to settle the question of whether Machines Can Think, a question of which we now know that it is about as relevant as the question of whether Submarines Can Swim.” (Dijkstra 1984)

En un segundo término, nos encontramos con los sistemas de machine learning, una tecnología que adquirió un desarrollo acelerado, sobre todo, en las últimas dos décadas. Esta tecnología involucra una nueva forma de programación, basada en el reconocimiento automático de patrones en los más diversos espacios de datos, generando clasificadores estadísticos capaces de comprimir y operativizar la información —considerada en un sentido laxo— contenida en esas grandes masas de datos.

En la actualidad, las tecnologías de machine learning plantean problemas específicos que hace falta enfocar. En un trabajo pionero (Joler y Pasquinelli 2021), los autores muestran a este tipo de tecnología como un nooscopio; es decir, un instrumento de magnificación del conocimiento, análoga a los microscopios o telescopios. Como aquellos, el nooscopio permite ver lo invisible a costa de producir ciertas distorsiones. En general, los sistemas producidos por machine learning permiten establecer correlaciones que serían imposibles de encontrar por métodos “humanos”. Los programas que emergen del proceso de entrenamiento son clasificadores estadísticos capaces de determinar con una alta probabilidad

si algún dato nuevo satisface el mismo criterio implícito que los elementos del conjunto de datos de entrenamiento.

Además de los clasificadores, y dependiendo de estos, en el presente hay una proliferación de sistemas generativos capaces de crear diferentes expresiones cognitivas y culturales. Entre estos programas son ejemplares, por su multiplicidad de uso, los grandes modelos de lenguaje, los cuales son capaces de producir textos de gran complejidad sobre un tema específico, entablar conversaciones amigables o incluso construir programas que resuelvan algún problema propuesto. Cabe preguntarse si estamos ante un verdadero salto cualitativo en las capacidades de los programas o, por ejemplo, si esta capacidad de programar permitirá a estos sistemas el automejoramiento -anhelado o temido- de las IA.

Hay una opacidad intrínseca en los clasificadores estadísticos. El código no es comprensible en su totalidad y la forma de representación de los datos queda diseminada en los pesos de una red, no es reconocible ni siquiera para quienes programaron el sistema. Los programas desarrollados por técnicas de programación usuales son artefactos con propiedades lógicas deducibles *a priori*; por lo tanto, analizables y demostrables matemáticamente. En cambio, los clasificadores estadísticos, sobre todo los producidos por *deep learning*, no tienen propiedades lógicas reconocibles y, en general, solo se enuncian propiedades empíricas y con credenciales epistémicas más dudosas. Se conoce el proceso de entrenamiento, y pueden calcularse ciertos parámetros de adecuación o robustez del modelo producido, aunque son usualmente *ad hoc*, por lo tanto, no es siempre claro el significado del valor de dichos parámetros.

Los sistemas de IA generativa aprenden a producir expresiones que son reconocidas como válidas por el clasificador asociado. La producción es aleatoria. Por ejemplo, en los sistemas de producción de lenguaje, las palabras se van produciendo secuencialmente usando funciones probabilísticas ponderadas por la subred semántica asociada al prompt introducido. No son grandes productores de novedades y no tienen ninguna aspiración de verdad. Producen textos probables basados en un entrenamiento con billones de entradas lingüísticas. Son capaces de cometer errores garrafales, pero, en general, darán respuestas estadísticamente aceptables. No producirán nunca grandes novedades y no hay ninguna razón para confiar en su adecuación a los hechos. A veces lo harán, otras no, y no hay ni habrá ningún criterio intrínseco para distinguirlas. Esto es válido también para la extraña habilidad que tienen de “programar”. Los programas escritos en lenguajes de programación usuales son también textos, asociados semánticamente a su contexto de producción. Si se formula con claridad el tipo de problema a resolver, los generadores pueden producir programas asociados estadísticamente a ese problema. Es difícil o, literalmente, improbable que se generen programas que incluyan alguna novedad interesante, por lo que parece

igualmente improbable que vayan a producir un automejoramiento, más aún dado que el programa generado mismo no posee un código comprensible. A riesgo de remarcar lo obvio, esta improbabilidad solo se aplica a los sistemas producidos por la actual tecnología de machine learning; esto no invalida la posibilidad de otro tipo de programa que se automejore, como ya presuponía Turing en 1947 (Turing 1947).

Es importante conocer todas estas limitaciones para saber qué esperar y qué no de estos sistemas y de sus posibles evoluciones. Seguramente seguirán mejorando en base a entrenamientos más exhaustivos y correcciones de algunos errores comunes. De hecho, los grandes procesadores de lenguaje más populares pasaron por una cantidad de arreglos y ajustes realizadas por miles de trabajadores para reducir comportamientos lingüísticos ofensivos o meramente para mejorar las interacciones. De cualquier manera, a menos que haya algún nuevo salto conceptual, las limitaciones enunciadas seguirán estando.

Si bien no hay ninguna pretensión de verdad en los generadores de lenguaje, son sistemas de mucha utilidad. Y aunque no puedan revelar de manera confiable conocimiento del mundo, sí constituyen un acceso privilegiado a lo que en la teoría política se conoce como “ideología”; es decir, las creencias, ideas, significados que estructuran la vida social. No sabemos si los textos producidos son verdaderos, pero sí sabemos que reflejan lo que se ha dicho o escrito sobre un tema dado (Weatherby 2023).

4. Ensamblajes computacionales

Para poder analizar en contexto las formas de existencia de los objetos digitales o computacionales, son necesarias herramientas conceptuales que den cuenta de su pluralidad y sus ambivalencias. John Johnston introduce el concepto de ensamblaje computacional para referir a la conjunción entre un mecanismo computacional dado y sus correlatos discursivos. Es decir, el conjunto de los programas, aplicaciones y sistemas computacionales y los marcos conceptuales, nociones y enunciados que conjuntamente construyen sentidos y condiciones de inteligibilidad de dichos sistemas

Este registro que refiere a lo que podríamos llamar la computación “*in the wild*”, permite analizar las características que provienen tanto del tipo de programación como del contexto de desarrollo y de aplicación de los sistemas en cuestión. En el presente, abundan ensamblajes computacionales basados en sistemas de machine learning, que son, en general, quienes detentan (o padecen) las

características que se discuten en muchos trabajos filosóficos, políticos o culturales. Sesgos, reproducción del pasado, predicción o alteración de las preferencias o las conductas, apropiación de la atención, desinformación y *fake news* son algunos de los males que se pretenden conjurar con la urgencia de los tiempos políticos. La buena noticia es que estas características no son inherentes a los sistemas algorítmicos en general y, en cierta medida, ni siquiera a los sistemas de machine learning, sino a los contingentes ensamblajes computacionales. Como ejemplo de esto, la noción de *gubernamentalidad algorítmica* planteada por Rouvroy y Berns (2013) cae en este registro, pero el uso de la palabra “algorítmica” sugiere una generalidad que no tiene. Para ser precisos habría que hablar de “gubernamentalidad basada en machine learning en el capitaloceno”, pero claramente el sintagma propuesto es mucho más atractivo y memorable.

Volviendo al nooscopio, el análisis realizado por los autores, y continuado en (Pasquinelli 2023), considera el contexto de producción de diferentes sistemas de machine learning mostrando que, en esos casos, los programas se construyen esencialmente a través de trabajo humano. Desde esta perspectiva buscan, también, dilucidar las profundas transformaciones en el mundo del trabajo.

Quizá una de las diferencias actuales esenciales entre cognición computacional y cognición humana está dada por la “falta de juicio” del lado computacional. Brian Cantwell Smith (2019) plantea la idea de “juicio” (*judgement*) como capacidad que, en el presente, sería distintivamente humana y que da cuenta de un compromiso existencial con el mundo. Más allá de la pregunta acerca de qué sería necesario para que sistemas de cognición computacional adquieran capacidad de juicio, cabe asumir una nueva paradoja reflexiva y preguntarse cómo se establece hoy un compromiso existencial con un mundo poblado de clasificadores estadísticos, cómo estos clasificadores integrados ya a la cotidianidad, a la toma de decisiones, a la cultura afectan y transforman la capacidad ¿humana? de juicio y sus alcances.

5. El interjuego: Machine learning como método de (meta)programación

“Hence, computing science is --and will always be-- concerned with the interplay between mechanized and human symbol manipulation, usually referred to as “computing” and “programming” respectively.” (Dijkstra 1988)

Programar es un proceso sofisticado de manipulación humana en determinados sistemas formales, basados en propiedades lógicas de los lenguajes de

programación y de las especificaciones del problema a resolver. Muchas herramientas conceptuales y lógicas están involucradas en este proceso. Se espera que quien programa pueda proveer no solo el programa funcionando, sino buenos argumentos de por qué resuelve el problema propuesto.

Los procesos de entrenamiento por machine learning no son tan generosos. Si bien, al final del proceso se dispone de un clasificador o un generador, la justificación de corrección o, al menos, de adecuación de la solución rara vez puede producirse. A lo sumo se puede dar cuenta del proceso de producción. Es por ello que estos clasificadores no poseen las certificaciones de calidad que sí tienen otro tipo de programas. Janlert (2008) duda de que pueda llamarse programas a estos sistemas, ya que estos deberían no solo establecer un sistema causal del proceso de ejecución y una descripción del proceso mismo, sino también una serie de justificaciones, de explicaciones racionales que permitan confiar en el sistema producido. Sin duda, estos sistemas son capaces de causar los procesos requeridos, pero ya su descripción es poco comprensible y la justificación suele ser inexistente (e imposible de construir). Estamos frente a lo que Janlert llama “*dark programming*”. Esta manera de programar da lugar a sistemas opacos en los que la forma de representación suele no ser explícita.

Desde sus inicios, la computación se basó en un mito fundacional: la posibilidad de liberar la mente de las tareas repetitivas. Paradójicamente, el vínculo que suele establecerse hoy con los sistemas de IA suele fomentar la repetición de conductas; disminuir la capacidad de atención necesaria para desarrollar nuevas ideas; aumentar, en resumen, la entropía cognitiva. Creemos que ese estado de cosas no es necesario. Las múltiples formas de programación, los interjuegos presentes y posibles entre programación y computación, abren un paisaje de oportunidades evolutivas que pueden renovar los acoplamientos cognitivos. Como intentó Hilbert a inicios del siglo pasado con la matemática, también podemos reformular las bases de la programación de sistemas de IA hoy, sin que nadie nos expulse del paraíso que Turing ha creado para nosotros.

6. La coevolución

“In the long run I expect computing science to transcend its parent disciplines, mathematics and logic, by effectively realizing a significant part of Leibniz’s Dream of providing symbolic calculation as an alternative to human reasoning. (Please note the difference between “mimicking” and “providing an alternative to”: alternatives are allowed to be better.)” (Dijkstra 1988)

Periódicamente vuelve a enunciarse la cuestión del “peligro existencial” planteado por la evolución de la IA. El riesgo de extinción de la humanidad estaría dado cuando la IA “supere” la inteligencia humana, alcance lo que se denomina “inteligencia artificial general” o super-inteligencia que sería incontrolable para humanos y que tomaría el control del mundo. Si bien eso está lejos de los actuales desarrollos, otro factor de preocupación es que las mismas IA sean capaces de automejorarse, adelantando el momento de la “singularidad”.

No abordaremos aquí esta pregunta, en particular, porque parece estar basada en suposiciones conceptuales erróneas, más allá de lo sugestivas que puedan resultar para la imaginación. Las diferentes formas de cognición computacional tienen características propias que no necesariamente coinciden con las formas cognitivas usuales en humanos. Y por supuesto que los sistemas de IA actuales no incorporan deseos ni emociones humanas, y no hay ninguna razón para que los tengan.

La mirada antropomórfica sobre la IA -y los usos lingüísticos asociados- es de alguna manera comprensible, pero oculta más de lo que muestra. Una de las maneras más insidiosas de este sesgo conceptual se manifiesta cuando se critica la IA para repetir que los sistemas computacionales no poseen real inteligencia en comparación con la humana, volviendo a situarse, anacrónicamente, en cierto dualismo cartesiano. Por el contrario, como afirma Dijkstra, la posibilidad de construir programas que permitan formas cada vez más sofisticadas de procesamiento simbólico es un camino para mejorar las capacidades de razonamiento humano o, lo que desde nuestra perspectiva sería lo mismo, de razonamiento de los acoplamientos cognitivos entre mentes y programas. Y no hace falta para esto tener sistemas computacionales autónomos (tampoco los humanos lo son), sino mejores integraciones entre dispositivos cognitivos, humanos o no. La autonomía siempre es relativa y gradual, y ese grado es mayor, de acuerdo con Simondon, cuando incrementa las posibilidades de individuación.

Parafraseando a Hansen, vemos como la distribución de la cognición, la percepción y la acción en el mundo se encuentra en un proceso de rápida evolución. La IA actual, en tanto reconocimiento estadístico de patrones, es una parte

importante de este proceso de redistribución, pero sus límites son ineluctables. Dos caminos posibles y compatibles entre sí se siguen de acá. Uno consiste en mejorar los ensamblajes computacionales basados en este momento tecnológico. Es posible (y necesario) constituir ensamblajes virtuosos que combinen tecnologías libres, vínculos cognitivos más transparentes y valores políticos progresivos. El otro camino, en la línea de la coevolución propuesta por Ashford Lee (2020), es más complejo, pero más transformador, y tiene que ver con encontrar nuevas bases conceptuales para el desarrollo de mediaciones computacionales (Marcus y Davis). Sistemas que superen la opacidad de los clasificadores estadísticos, que puedan acceder a su propio código y que habiliten ensamblajes donde, tanto humanos como ambientes tecnológicos se integren en procesos mutuamente reflexivos, entendiéndose por esto la posibilidad de transformar el ambiente mismo “desde dentro”. Quizá haya que reconstruir muchos sistemas actuales basados en una tecnología más transparente y confiable, aunque esto implique cierta ralentización del desarrollo. Es un camino difícil, pero, como decía Rilke, que algo sea difícil es una razón más para elegirlo.



7. Bibliografía

- Ashford Lee, Edward. *The Co-evolution. The Entwined Futures of Human and Machines*. The MIT Press, 2020.
- Bratton, Benjamin. *The Stack*. MIT Press, 2015.
- Crawford, Kate. *Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence*. Yale University Press, 2021.
- Dijkstra, Edsger. “The threats to computing science.” EWD 898 Texas: Universidad de Austin, 1982.
- . “On the cruelty of really teaching computing science.” EWD 1036 Texas: Universidad de Austin, 1988.
- Hansen, Mark. “Engineering Pre-individual Potentiality: Technics, Transindividuation, and 21st-Century Media”. *Sub-Stance*, vol. 41, no. 3, 2012, pp. 32-59.
- Janlert, Lars-Erik. “Dark programming and the case for the rationality of programs”. *Journal of Applied Logic*, vol. 6, 2008, pp. 545-552.
- Joler, Vlado y Matteo Pasquinelli. “El Nooscopio de manifiesto”. *La Fuga*, no. 25, 2021. Disponible en <https://lafuga.cl/el-nooscopio-de-manifiesto/1053>.
- Marcus, Gary y Davis, Ernest. *Rebooting AI: Building Artificial Intelligence we can trust*. Pantheon Books, 2019.
- Pasquinelli, Matteo. *The Eye of the Master: A Social History of Artificial Intelligence*. Verso, 2023.
- Rapaport, William. *Philosophy of Computer Science: An Introduction to the Issues and the Literature*. John Wiley and Sons, Inc. 2023.
- Smith, Brian Cantwell. *The Promise of Artificial Intelligence. Reckoning and Judgment*. MIT Press, 2019.
- Stiegler, Bernard. “Artificial Stupidity and Artificial Intelligence in the Anthropocene”. Conferencia del 23 de noviembre de 2018, Institute of Ereignis, Shanghai.
- Turing, A.: “The Automatic Computing Engine”. Conferencia dictada en la London Mathematical Society, 20 February 1947.
- Weatherby, Leif. “ChatGPT Is an Ideology Machine”. *Jacobin*. 17 de abril de 2023. Science and Technology / Theory.