

STEFANO PIETROPAOLI

*Ma di' soltanto una parola.
Contro la decisione algoritmica*

Abstract

The article provides a critical examination of the concept of “algorithmic decision-making” and its implications for the legal field, with particular attention to its impact on the foundational principles of the legal system. It foregrounds the uniquely human character of law as an artifact of linguistic and intellectual creation. Central to the legal system is the act of “articulating” the law – a practice inherently reliant on human judgment, interpretive reasoning, and the contextualization of facts and norms. While advanced technologies, such as natural language processing (NLP) and computational systems, enable machines to generate and process language, these systems lack the inherent capacity for genuine comprehension or autonomous decision-making. Their outputs are mere simulations produced through statistical models and pre-defined algorithms, devoid of intentionality and deeper understanding.

The article situates this discussion within a historical framework, tracing the origins of computational law to Leibniz’s vision of a *mathesis universalis*, thereby engaging with long-standing debates regarding the computability of legal reasoning. It further explores contemporary applications, such as smart contracts and autonomous vehicles, while critically interrogating whether the translation of legal norms into technical commands undermines the normative and ethical essence of the law.

In conclusion, the article argues that the delegation of judicial decision-making to algorithms risks eroding the human essence of the legal system. Genuine legal reasoning requires core human attributes, such as empathy, contextual discernment, and a commitment to justice – qualities inherently absent in algorithmic processes. The reduction of law to a mechanistic system threatens its integrity as a human construct, designed not merely to regulate behavior but to navigate the complexities of human society and strive toward the perpetually elusive ideal of justice.

Keywords

- Algorithmic decision-making
- Computational law
- Legal interpretation
- Artificial intelligence and law
- Human-centered justice.

Sommario

1. Il diritto si dice. E le macchine non dicono
2. Un evidente dono per l’imitazione. L’illusione dei sistemi di elaborazione del linguaggio naturale (NLP)
3. Calcolare il diritto?
4. Le macchine non decidono

1. Il diritto si dice. E le macchine non dicono

L'artificio umano che segna il diritto è la parola.

Le parole del diritto sono fatti – o meglio artefatti – frutto di un'arte, di una tecnica e di una scienza che sono opera dell'intelletto umano. I giuristi lavorano con le parole: elaborano concetti, inventano strategie argomentative, attribuiscono incessantemente significati a enunciati linguistici. Le parole non sono entità naturali, non si possono toccare, non hanno una corporeità. Eppure, una volta pronunciate diventano fatti capaci di incidere sulla vita di coloro ai quali sono rivolte.

Per sgombrare il passo da possibili fraintendimenti propongo un esempio. Ciò che chiamiamo “omicidio” non ha senso – perché *non esiste* – al di fuori del diritto. Può certo accadere che un essere umano provochi la morte di un suo simile, ma ciò che rileva sul piano giuridico è la *qualificazione* di quel particolare evento, capace di generare conseguenze che incidono concretamente nella vita dei consociati: conseguenze molto diverse, a seconda che quel determinato evento sia qualificato come, per l'appunto, un omicidio, oppure come esecuzione di una sentenza, come atto di legittima difesa, come esito di un'azione di guerra, e così via. Questa qualificazione, il cui esito sarà destinato a incidere sulla viva carne di un essere umano, avviene in una dimensione artificiale, altra dalla realtà naturale: dimensione artificiale *in quanto umana*, frutto dell'opera degli esseri umani.

Il diritto non ha nulla di naturale. Il diritto è una delle più alte espressioni dell'intelletto umano: non esiste *in natura*.

Il diritto, in quanto scienza giuridica, è giuris-dizione. Il diritto *si dice*.

Il diritto è *jus*, non *fas*: ha a che fare non con la giustizia divina, ma con la giustizia umana. Per riprendere una suggestione di Émile Benveniste¹: è espressione non di *thémis*, ma di *díke*, la cui radice *deik- si sviluppa in *deiknumi* (mostrare) in greco e in *dico* (dire) in latino. Il *judex* (colui che dice il *jus*) mostra non col gesto ma con la parola (non qualcosa che è, ma) ciò che *deve essere*.

Oggi il mondo dell'uomo, fatto di carne e di sangue, si sta fondendo con il mondo della macchina, fatto di bit e silicio. La vecchia umanità è diventata “antiquata”². Da una parte,

¹ É. Benveniste, *Il vocabolario delle istituzioni indoeuropee*, vol. II: *Potere, diritto, religione*, Einaudi, Torino, 1976, pp. 363-366.

² Il riferimento d'obbligo è ovviamente G. Anders, *L'uomo è antiquato*, vol. I: *Considerazioni sull'anima nell'epoca della seconda rivoluzione industriale*, Bollati Boringhieri, Milano, 2005, sul quale si veda adesso il recente contributo di S. Vantin, *L'uomo è antiquato? Responsabilità, tecnica e norma nella riflessione di Günther Anders*, in *Diacronia*,

l'uomo si avvale sempre più frequentemente di protesi artificiali di ogni genere, che innestate sul proprio corpo gli consentono di sviluppare capacità del tutto nuove. Dall'altra, le macchine stanno acquisendo capacità e qualità ritenute fino ad oggi intimamente umane: una razionalità e un'autonomia che mimano quelle umane³ (se non una vera e propria "intelligenza"), ma anche una diversa fisicità, basata sulla possibilità di avvalersi di tessuti organici. Organico e sintetico stanno saldandosi, in una nuova simbiosi che non può lasciare indifferenti. L'uomo viene macchinizzato, la macchina viene umanizzata.

Che cosa sarà dunque del diritto, in un'epoca in cui i confini dell'umano sembrano ormai destinati a non essere più tracciabili con la sicurezza che ci eravamo illusi di trovare nel corso del Novecento?

Se vogliamo salvare il diritto, e con esso la (sua) umanità, occorre riflettere ancora una volta sulla differenza tra essere umano e artefatto, recuperando il senso originario dell'artificiale come espressione e non come alternativa all'umano (in questa prospettiva nessuna espressione ha mai ingenerato fraintendimenti più di "intelligenza artificiale")⁴⁵.

Da giurista, ciò che istintivamente vorrei affermare è che il diritto esisterà fino a quando resterà *parola*, e in quanto tale espressione esclusiva dell'umanità.

I viventi diversi dall'uomo comunicano tra loro, anche in maniera estremamente articolata ed efficace, attraverso suoni e vocalizzazioni (l'ululato del lupo o il canto della balena), gesti (il cavallo che batte lo zoccolo sul terreno, la danza delle api), posture (il gatto che inarca la schiena) e segnali chimici (i feromoni delle formiche per tracciare un percorso, o più banalmente l'urina con cui i cani marcano un territorio). Possono, inoltre, organizzarsi in comunità con un certo grado di complessità, in cui i membri collaborano, comunicano e svolgono ruoli specifici, osservando determinate regole (dalle formiche agli elefanti, dai delfini alle termiti). Tuttavia, essi *non dicono*. E *non conoscono il diritto*.

2023, 1, pp. 189-217. Ma va ricordato anche Arnold Gehlen (*L'uomo nell'era della tecnica*, Armando, Roma, 2003), la cui antropologia filosofica può essere considerata come anticipatrice del pensiero *post-human*. Cfr. anche M. Farisco, *Ancora uomo. Natura umana e postumanesimo*, Vita&Pensiero, Milano, 2011; Id., *Uomo, natura, tecnica. Il modello postumanistico*, Zikkurat, Roma-Teramo 2008. Per quanto riguarda un'analisi critica della visione gehleniana della tecnica, cfr. U. Fadini, *Sviluppo tecnologico e identità personale. Linee di antropologia della tecnica*, Dedalo, Bari, 2000.

³ Cfr. G. Tamburrini, *Autonomia delle macchine e filosofia dell'intelligenza artificiale*, in *Rivista di filosofia*, 2017, 2, pp. 263-275.

⁴ R. Bodei, *Dominio e sottomissione: schiavi, animali, macchine, intelligenza artificiale*, il Mulino, Bologna, 2019.

⁵ E. Mazzarella, *Contro metaverso: salvare la presenza*, Mimesis, Milano, 2022.

Se questo vale per i viventi diversi dall'uomo, che cosa possiamo dire invece in riferimento ai non viventi? In altre parole, che cosa possiamo dire delle macchine?

Le macchine – nella forma di computer e intelligenze artificiali – possono elaborare automaticamente il linguaggio umano. Ma occorre sottolineare che le macchine non comprendono le parole, esattamente come non vedono le immagini e non ascoltano la musica, ma le elaborano dopo averle trasformate in bit: le cifre binarie sono, in definitiva, l'unica cosa che le macchine possono “capire”.

I sistemi di dettatura e i comandi vocali per smartphone o per dispositivi domestici intelligenti trasformano i suoni prodotti dalla voce umana – le parole – in codice binario. Grazie alla sintesi vocale, viceversa, è possibile trasformare il codice in voce parlata. Il risultato è che assistenti vocali come Siri, Alexa e Google Assistant riproducono suoni che imitano il tono, l'accento e il ritmo della voce umana, in modo tale che chi li ascolta li possa interpretare come parole. Questi assistenti vocali emettono suoni, certo, ma *non dicono*.

2. Un evidente dono per l'imitazione. L'illusione dei sistemi di elaborazione del linguaggio naturale (NLP)

Non è necessario, credo, scomodare la stanza cinese di John Searle⁶, per mettere a fuoco il tema della afasia delle macchine. La mia professoressa di greco al liceo aveva sicuramente ben chiaro che, mentre dettava le versioni da tradurre, quei “bestioni” dei suoi allievi non avevano la benché minima idea di che cosa quell'insieme di lettere, spiriti e accenti significasse. La parte degli “insensati bestioni” di vichiana memoria la recitano qui le macchine: con la differenza, rispetto ai giovani liceali, che esse vengono acriticamente etichettate come “intelligenti”.

Mi viene in mente la scena del film del 1968 *Il pianeta delle scimmie* in cui il Dottor Zaius (un orango che riveste la carica di Ministro per la ricerca scientifica), di fronte all'astronauta terrestre impersonato da Charlton Heston e che – contrariamente agli esseri umani presenti sul suo pianeta – “sembra” poter parlare, osserva: «Divertente. Un uomo che si

⁶ Rinvio almeno a J. R. Searle, *Menti cervelli e programmi: un dibattito sull'intelligenza artificiale*, CLUED, Milano, 1984. Si veda adesso anche il recente J. R. Searle, *Intelligenza artificiale e pensiero umano: filosofia per un tempo nuovo*, Castelvechi, Roma, 2023.

comporta come una scimmia». In risposta alle rimostranze dell'uomo, Zaius si rivolge ai colleghi rimarcando che l'animale che hanno davanti dimostra «un evidente dono per l'imitazione (*He shows a definite gift for mimicry*)», per concludere, infine: «L'uomo non ha comprensione. Può essere addestrato a fare qualche semplice trucco. Nient'altro (*Man has no understanding. He can be taught a few simple tricks. Nothing more*)».

L'atteggiamento del Dottor Zaius, ovviamente errato nella finzione cinematografica, è quello che invece dovremmo avere di fronte alle moderne macchine basate su sistemi di elaborazione del linguaggio naturale (NLP). Gli algoritmi di NLP – su cui si basano oggi chatbot, traduttori automatici, motori di ricerca – analizzano enormi quantità di dati per poter poi generare delle “risposte”. Ma le macchine non hanno una comprensione intrinseca del significato delle domande che vengono loro sottoposte, e tantomeno delle risposte che forniscono agli utenti. Esse, semplicemente, processano dati e generano risultati basati su modelli statistici.

La capacità delle macchine di simulare la parola umana dipende dalla quantità e qualità dei dati su cui sono state addestrate. Se i dati sono incompleti o distorti, anche le risposte delle macchine saranno imperfette. Se i dati sono moltissimi e affidabili, il trucco resta ma non si vede.

Il recente successo dei sistemi di intelligenza artificiale generativa è legato proprio alla grande quantità di dati di addestramento, oltre che alla potenza computazionale impiegata per generare linguaggio artificiale. Nella fase di addestramento di ChatGPT, per esempio, gli algoritmi di apprendimento automatico hanno analizzato enormi quantità di testo per individuare regole, modelli e sfumature del linguaggio umano, migliorando le loro prestazioni nel tempo senza essere esplicitamente programmate per ogni singola operazione, così da generarne imitazioni sempre migliori. Anche se ChatGPT può generare testi altamente complessi, la sua abilità rimane tuttavia limitata e vincolata ai modelli e ai dati su cui è stato addestrato. L'algoritmo può essere supervisionato o meno, può sostanzarsi in reti neurali, ma il senso non cambia: viene addestrato su dataset di testo in base ai quali calcola la probabilità che determinate sequenze di parole abbiano – per gli umani – un significato.

In sostanza, le macchine individuano pattern sulla base dei quali formulano previsioni, col risultato di simulare risposte che sembrano intelligenti, senza tuttavia avere una

comprensione intrinseca del loro significato. Sono macchine combinatorie: calcolano con simboli, risolvendo problemi complessi, ma non capiscono il significato di quei segni e tantomeno perché il problema debba essere risolto. Anche se una macchina è in grado di superare il Test di Turing, ciò non implica che essa pensi come un essere umano: può solo imitare il comportamento umano in modo più o meno convincente. Ma rimane l'attore inumano di un *imitation game*.

3. Calcolare il diritto?

È nel Seicento che si afferma nella sua pienezza l'idea che si possa “calcolare con i concetti” nello stesso modo in cui è possibile calcolare con i numeri. Ed è sempre nel Seicento che si afferma per la prima volta l'idea della computabilità del diritto⁷. Per elaborarla occorre una mente versata in egual modo per la filosofia, il diritto, la logica e la matematica.

Gottfried Wilhelm Leibniz è stato il primo a elaborare compiutamente l'idea che anche il diritto – tradizionalmente basato su principi interpretativi e di natura argomentativa – possa essere concepito come un sistema computabile, cioè regolato da meccanismi logici che permettano di dedurre le conclusioni a partire da premesse definite⁸. La sua ambizione era creare una *mathesis universalis*, una scienza universale del ragionamento, che potesse formalizzare tutti i campi della conoscenza attraverso un linguaggio simbolico universale⁹. Il suo progetto più noto in questo senso è il *calculus ratiocinator*, una sorta di linguaggio logico-matematico che avrebbe permesso di risolvere dispute concettuali in modo oggettivo, proprio come si risolvono equazioni aritmetiche. Nella visione di Leibniz, si trattava di trovare un linguaggio che rendesse possibile “calcolare” con le idee e i concetti giuridici, riducendo la complessità della giurisprudenza a un insieme di regole e operazioni deduttive.

⁷ Non mi riferisco qui al diverso seppur connesso tema della calcolabilità del diritto come prevedibilità, sul quale non posso che rinviare al fondamentale N. Irti, *Un diritto incalcolabile*, Giappichelli, Torino, 2016.

⁸ M. R. Antognazza, *The Oxford Handbook of Leibniz*, Oxford University Press, New York, 2018. Per un'introduzione, del medesima autrice, cfr. M. R. Antognazza, *Leibniz: An Intellectual Biography*, Cambridge University Press, Cambridge, 2009.

⁹ P. P. Portinaro, *Leibniz, la logica e la giurisprudenza*, in *Materiali per una storia della cultura giuridica*, fasc. XLVI, 1, 2016, pp. 239-252.

Questa idea della computabilità del diritto, per quanto utopica, precorre in modo sorprendente le moderne teorie dell'informatica giuridica e dell'intelligenza artificiale applicata al diritto. Leibniz immaginava che una volta formalizzato il sistema giuridico, ogni disputa giuridica potesse essere risolta semplicemente inserendo le premesse legali all'interno di un algoritmo, che avrebbe prodotto una sentenza logicamente corretta e imparziale.

Quello che oggi chiamiamo diritto computazionale, dunque, proviene dal secolo barocco. La potenza di calcolo che Leibniz più di tre secoli fa poteva soltanto immaginare è diventata realtà. Lo straordinario sviluppo tecnologico impresso dalla rivoluzione informatica ha ingrossato la schiera di chi – in modi e per scopi assai diversi da quelli che avevano animato il razionalismo seicentesco – immagina un diritto integralmente formalizzato, espresso in simboli computabili da una macchina.¹⁰

Le norme giuridiche vengono così tradotte in linguaggi di programmazione, in modo tale che le macchine possano compiere operazioni con esso¹¹. Il *codex* diventa *code*. La parola si fa bit.

Il diritto computazionale prevede l'uso di tecnologie informatiche avanzate per automatizzare, interpretare e applicare il diritto¹². L'obiettivo è rappresentare il diritto in formato leggibile da una macchina al fine di creare sistemi che possano “comprendere” le norme giuridiche e prendere “decisioni” giuridicamente rilevanti. Attraverso una serie di tecniche avanzate di intelligenza artificiale, una disposizione normativa viene divisa in unità più piccole (ad esempio, parole o frasi) chiamate *token*, ognuna delle quali viene sottoposta a un'analisi strutturale (*parsing*) e conseguentemente “marcata”. Le parole diventano vettori numerici.

Per comprendere come questo diritto computazionale sia già realtà – peraltro già disciplinata normativamente anche in Italia – e non una futuribile chimera, possiamo

¹⁰ Cfr. W. Lenzen, *Leibniz on alethic and deontic modal logic*, in D. Berlioz, F. Nef (a cura di), *Leibniz et les puissances du langage*, Vrin, Paris, 2005, pp. 341-362; D. M. Katz, *Quantitative Legal Prediction – or – How I Learned to Stop Worrying and Start Preparing for the Data Driven Future of the Legal Services Industry*, in *Emory Law Journal*, fasc. 62, 2013.

¹¹ Sul tema, imprescindibile L. Lessig, *Code and other laws of cyberspace*, Basic Books, New York, 2000. Si veda del medesimo autore adesso anche L. Lessig, *Code: version 2.0*, Basic Books, New York, NY, 2006. Cfr., inoltre L. E. Di-ver, *Digisprudence: code as law rebooted*, Edinburgh University Press Ltd, Edinburgh, 2022.

¹² Per un'introduzione cfr. A. Andhov, *Computational Law*, Karnov, Copenhagen, 2022. Si veda anche J. Weidong, *The domain of computational law*, in *Peking University Law Journal*, fasc. 10, 2, 2022, pp. 109-130.

prendere come esempio il cosiddetto *smart contract*¹³. Questo termine si riferisce a un programma informatico che esegue automaticamente i termini di un contratto quando sono soddisfatte determinate condizioni predefinite. I contratti sono memorizzati su una *blockchain*, una tecnologia di registro distribuito che garantisce la trasparenza e l'immutabilità delle transazioni.

Il meccanismo di base può essere riassunto in maniera elementare: le condizioni e i termini del contratto vengono tradotti in codice informatico; una volta che le condizioni predefinite sono soddisfatte, lo *smart contract* esegue automaticamente l'azione specificata, senza la necessità di intervento umano; poiché gli *smart contract* sono eseguiti sulla *blockchain*, ogni operazione è visibile a tutte le parti interessate e non può essere modificata o falsificata.

Ad esempio, un semplice *smart contract* potrebbe stabilire che se l'acquirente invia una certa somma di criptovaluta all'indirizzo del venditore, allora il venditore deve trasferire l'oggetto digitale all'acquirente. Una volta che il pagamento è stato effettuato, lo *smart contract* esegue automaticamente l'invio, senza la necessità di intervento umano.

Se usiamo un linguaggio di programmazione molto noto come Solidity, l'esempio che abbiamo fatto viene tradotto come segue¹⁴:

```
``solidity
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract SimpleSale {
    address public seller;
    uint public price;
    string public item;
    bool public itemTransferred;

    event ItemPurchased(address buyer, uint amount);
```

¹³ Sul tema cfr. F. Murino, *Dalla firma elettronica agli smart contract*, in T. Casadei, S. Pietropaoli (a cura di), *Diritto e tecnologie informatiche: questioni di informatica giuridica, prospettive istituzionali e sfide sociali*, Wolters Kluwer CEDAM, Milano, 2024, pp. 145-166.

¹⁴ Alcuni termini indicano le variabili: *address public seller* è l'indirizzo del venditore che distribuisce l'oggetto; *address buyer* è l'indirizzo dell'acquirente; *uint public price* è il prezzo dell'oggetto; *item* è la descrizione o l'identificativo dell'oggetto che viene venduto. Altri termini indicano invece gli eventi: *ItemPurchased* indica che l'oggetto è stato acquistato; *ItemTransferred* che l'oggetto è stato trasferito all'acquirente. Altri termini ancora indicano le funzioni: *ItemPurchased* permette all'acquirente di acquistare l'oggetto inviando il pagamento corretto, e se l'importo è esatto e l'oggetto non è già stato trasferito, la funzione registra l'acquirente; *itemTransferred* permette invece al venditore di trasferire l'oggetto all'acquirente e di ricevere il pagamento, una volta verificato che solo il venditore possa eseguire questa operazione e che l'oggetto non sia già stato trasferito.

```

event ItemTransferred(address seller, address buyer, string item);

constructor(uint _price, string memory _item) {
  seller = msg.sender;
  price = _price;
  item = _item;
  itemTransferred = false;
}
function purchaseItem() public payable {
  require(msg.value == price, "Incorrect payment amount.");
  require(!itemTransferred, "Item has already been transferred.");

  itemTransferred = true;
  emit ItemTransferred(seller, msg.sender, item);

  emit ItemPurchased(msg.sender, msg.value);
}
}

```

Il venditore (*seller*) crea il contratto specificando il prezzo (*price*) dell'oggetto e una sua descrizione (*item*). Un acquirente (*buyer*) innesca la funzione *purchaseItem* inviando esattamente la somma richiesta: l'oggetto viene automaticamente inviato all'acquirente, e i fondi vengono inviati al venditore.

La logica di trasferimento (`itemTransferred = true`) e il pagamento (`payable(seller).transfer(price)`) vengono eseguiti subito dopo aver verificato che il pagamento è corretto e che l'oggetto non è già stato trasferito. L'evento `ItemTransferred` si realizza immediatamente dopo che l'oggetto viene trasferito, seguito dall'evento `ItemPurchased`. In altre parole, una volta che l'acquirente invia la somma richiesta, l'intera transazione avviene senza che il venditore debba fare nulla dopo la configurazione iniziale del contratto.

Questo esempio banale esprime uno dei possibili esiti del diritto computazionale: migliorare la velocità e l'efficienza dei processi e, soprattutto, ridurre gli errori umani nel prendere decisioni giuridicamente rilevanti.

Un altro esempio può spiegare meglio di molte parole le attuali prospettive del diritto computazionale. Com'è noto, tutte le maggiori aziende automobilistiche stanno investendo nello sviluppo di veicoli a guida autonoma. Tali mezzi di trasporto non soltanto sapranno individuare gli ostacoli (un pedone, un marciapiede, e così via) ed evitarli, scegliere l'itinerario più breve o più panoramico, attivare i tergicristalli alla prima goccia di pioggia, frenare in caso di coda e così via, ma potranno essere impostati per individuare e interpretare specifici segnali stradali (disposizioni normative rese computabili) e di con-

sequenza rispettare i limiti di velocità previsti per la tipologia di strada che si sta percorrendo, mantenere automaticamente la distanza di sicurezza in autostrada, fermarsi al semaforo rosso, e molto altro ancora.

Una conseguenza teoricamente relevantissima di questo *encoding* normativo è la sostanziale impossibilità per il viaggiatore umano di violare le norme del Codice della strada. Il veicolo *non potrà* superare i limiti di velocità, violare la distanza di sicurezza o passare col rosso. Ma, se la violazione della norma non sarà più concretamente possibile, saremo ancora di fronte a una norma giuridica? La norma – perché certo di norma ancora si tratta – non sarebbe forse inquadrabile come una norma meramente tecnica (e *non giuridica*), destinata dunque a stabilire il funzionamento di un sistema piuttosto che a orientare il comportamento umano? Il Codice originariamente rivolto agli esseri umani non verrebbe così sostituito da un “manuale di istruzioni” destinato alla macchina, scritto esclusivamente in un linguaggio a lei comprensibile?

Non è questa la sede per tentare di inquadrare in maniera più precisa la questione. Tuttavia, mi sia consentito accennare a una distinzione che può forse contribuire a una più efficace riflessione sul tema. La formalizzazione necessaria a rendere “comprensibile” alla macchina la conoscenza giuridica ha per oggetto soltanto i testi normativamente rilevanti (leggi o sentenze che siano) oppure il diritto nella sua complessità? In altri termini, la traduzione dal linguaggio naturale al linguaggio-macchina riguarda la sola disposizione normativa o anche la norma in senso proprio, cioè il significato riferibile a quella disposizione?

Se, infatti, non mi pare che si possa negare la possibilità di rappresentare in maniera formale anche la più complessa delle disposizioni normative, tutt'altro grado di difficoltà riguarda due altri aspetti fondamentali dell'esperienza giuridica: l'interpretazione e l'applicazione del diritto. Arriviamo così al cuore del problema: il diritto computazionale può sostituire la decisione umana?

4. Le macchine non decidono

Nel lessico informatico si suole parlare di “algoritmi di decisione” per indicare una classe di algoritmi progettati per prendere decisioni o risolvere problemi attraverso

l'analisi di dati e l'esecuzione di regole predefinite. Esistono diversi tipi di algoritmi di decisione, ciascuno con caratteristiche specifiche in base al contesto di utilizzo: alberi di decisione, *random forests*, macchine a vettori di supporto, *Rule-Based Systems*. Il loro impiego è ormai diffuso in svariati ambiti, dalla medicina (dove sono usati per diagnosticare malattie o scegliere i trattamenti più appropriati sulla base dei dati dei pazienti), alla finanza (dove vengono utilizzati per l'analisi dei rischi, la previsione dei mercati e la valutazione dei prestiti).

Non è mia intenzione esaminare in questa sede gli indubbi vantaggi in termini di efficienza di queste tecniche, e neppure insistere sui potenziali rischi connessi al loro impiego. Il mio bersaglio polemico è un altro: l'espressione "decisione algoritmica".

Le decisioni umane implicano volontà, consapevolezza e comprensione del contesto. Un essere umano riflette sulle informazioni disponibili, considera le possibili conseguenze e sceglie intenzionalmente un'opzione, spesso considerando anche fattori etici, politici e giuridici nelle loro decisioni. Inoltre, le decisioni umane sono influenzate dalle esperienze passate, dalle emozioni e da quelli che potremmo chiamare istinto o intuizione¹⁵.

Le macchine si basano su algoritmi che elaborano dati e applicano regole predefinite. Se evitiamo ingenui antropomorfismi, è del tutto evidente che non ha senso alcuno parlare di "decisione" in riferimento a una macchina. La macchina esegue, in modo coerente e ripetibile, senza emozioni o esperienze personali. L'algoritmo è solo uno strumento: non prende decisioni. Esso esegue una serie di calcoli e operazioni per risolvere un determinato problema, ma non ha la capacità di scegliere in senso stretto.

Pensiamo al cubo di Rubik: lo si risolve seguendo una sequenza di mosse, in altre parole eseguendo algoritmi. Questo processo non implica alcuna decisione. Il discorso non cambia se affrontiamo problemi più complessi. Riprendiamo anche qui il caso dei veicoli a guida autonoma. Dal punto di vista tecnico, le automobili utilizzano gli algoritmi di decisione per "scegliere" tra diverse opzioni basate su criteri predefiniti e dati di input. Queste opzioni sono predeterminate dai programmatori e limitate dalle capacità del sistema di riconoscere e rispondere ai dati ambientali. Il sistema gestisce il percorso in base ai dati acquisiti in tempo reale dai sensori, a modelli di guida programmati e parametri

¹⁵ Sul tema della rilevanza giuridica della decisione, fondamentale ancora oggi A. Catania, *Decisione e norma*, Jovene, Napoli, 1979 (meritoriamente riedito nel 2023 da Castelvechi Editore, a cura di Valeria Giordano e Francesco Mancuso).

predefiniti, effettuando “scelte” su itinerario, velocità, distanza di sicurezza, e così via. Queste “scelte”, pur se addestrate con dati storici e modelli predittivi, sono limitate agli scenari previsti dai programmatori. Le macchine non scelgono veramente, ma arrivano a un risultato in base ai dati e alle regole che seguono, in altre parole eseguono comandi determinati da come sono progettate e programmate.

Qualcuno potrebbe obiettare che questa considerazione riguarda soltanto gli algoritmi cosiddetti “deterministici”, ossia quelli che, dato un certo input, producono sempre il medesimo output (per intenderci: un algoritmo che deve organizzare una lista di numeri in ordine crescente, ogni volta che gli viene riproposto il medesimo insieme di numeri di partenza restituirà sempre la medesima lista come risultato). A mio avviso, l'assenza di qualsiasi aspetto “volitivo” riguarda invece anche gli algoritmi non deterministici (quelli che possono produrre output diversi a partire da un medesimo input, includendo elementi di “casualità” o criteri variabili). Infatti, anche se un algoritmo può produrre output diversi basati su input variabili, il suo comportamento rimane predeterminato dalle regole e dai modelli matematici con cui è stato progettato. Su questo tema è opportuno soffermarci.

Se chiedo a un sistema informatico di “scegliere” un numero casuale da 1 a 100, qual è la ragione della opzione apparentemente effettuata dalla macchina? Non si tratta dell'espressione di una libera volontà di scelta, o di una scelta “casuale”. Le macchine possono generare risultati (e non decisioni) “casuali”, ma non nel senso umano del termine. Tornando al nostro esempio: i numeri generati in maniera apparentemente casuale dai computer, devono essere più correttamente definiti come “pseudo-casuali”. Ciò in quanto sono prodotti da algoritmi (PRNG: Pseudo-Random Number Generators) il cui risultato è prevedibile se si conosce lo stato iniziale del sistema. Si tratta di algoritmi impiegati frequentemente nelle simulazioni di possibili scenari in ogni campo del sapere. Le simulazioni c.d. Monte Carlo, per esempio, utilizzano numeri pseudo-casuali per simulare fenomeni fisici, per valutare la resilienza di una certa struttura, oppure per modellare l'andamento dei prezzi delle azioni e valutare il rischio di portafogli di investimento.

Gli algoritmi PRNG utilizzano una funzione matematica che prende il “seme” (*seed*) di partenza e genera una sequenza di numeri seguendo una formula ricorsiva: la sequenza generata è periodica, vale a dire che dopo un certo numero di iterazioni si ripete. La se-

quenza di numeri generata *sembra* casuale, ma non lo è. In molti casi la generazione del numero avviene tramite l'esecuzione di un semplice comando in Python, in cui il seme è l'orario di invio della richiesta (dall'anno, mese, giorno, ora, e minuto, fino ad arrivare magari al millesimo di secondo e oltre). Se il seme e l'algoritmo sono noti, la sequenza è certa e il risultato prevedibile.

Questa considerazione mi pare perfettamente replicabile, ai fini del nostro discorso, anche ai casi di cosiddetta “generazione di numeri casuali veri”. Si tratta di casi in cui i semi sono determinati da sorgenti di entropia fisica, e dunque da fenomeni che sono casuali e imprevedibili come il rumore termico (il movimento casuale degli elettroni in un conduttore a causa dell'energia termica) o fenomeni quantistici (come il decadimento radioattivo o la fluttuazione dei fotoni, intrinsecamente casuali secondo la meccanica quantistica).

La generazione di numeri casuali veri attraverso fenomeni fisici esterni implica che la macchina stessa non stia scegliendo in senso proprio. Piuttosto, la macchina sta misurando o rilevando eventi casuali che avvengono indipendentemente da essa. Ciò solleva una distinzione importante tra il ruolo passivo e attivo della macchina nella generazione della casualità. Le macchine utilizzano sensori e dispositivi per misurare fenomeni fisici esterni che sono casuali per natura. In questo processo, la macchina non “sceglie” il fenomeno, ma lo rileva e lo traduce in dati utilizzabili. Un dispositivo che utilizza il rumore termico per generare numeri casuali misura le fluttuazioni degli elettroni in un conduttore, fenomeno che avviene naturalmente e indipendentemente dal dispositivo stesso. Dopo aver rilevato i fenomeni esterni, la macchina converte questi segnali in numeri utilizzabili attraverso algoritmi e processi matematici. Questa conversione parte da un input casuale. Ma rimane di per sé deterministica.

Anche quando utilizzano input casuali, gli algoritmi rimangono deterministici nelle loro operazioni. Se le macchine possono produrre risultati che sembrano casuali, questi sono il risultato di regole predefinite e input esterni, piuttosto che di una scelta autonoma o di libero arbitrio.

Le macchine *non* prendono decisioni, dunque. E ben strani effetti sarebbero causati da una decisione a loro pienamente imputabile, a partire dal piano della responsabilità.

Le macchine non avrebbero comunque consapevolezza, intenzionalità o capacità di comprendere il contesto (giuridico e morale) delle loro decisioni. Esse funzionano se-

guendo regole predefinite e algoritmi, dando esecuzione a istruzioni programmate e producendo risultati basati sui dati e sui modelli matematici che sono stati loro forniti.

Per questi motivi, la responsabilità ricade e ricadrà sempre sugli esseri umani e sulle organizzazioni che progettano, sviluppano, implementano e supervisionano questi sistemi. Parlare di decisioni algoritmiche, dunque, rischia di attribuire alla tecnologia una capacità che essa non possiede davvero.

Come già detto, gli algoritmi conducono a *risultati*, non a *decisioni*. La differenza è fondamentale. Voi direste mai che un semaforo, quando diventa rosso, “decide” di fermarvi? Si tratta di un atto automatico, vincolato da un meccanismo predefinito, come ebbe modo di notare già Carl Schmitt¹⁶. Lo stesso vale quando usiamo una calcolatrice: se moltiplichiamo 3x3 e otteniamo 9, chi direbbe che la calcolatrice ha preso una “decisione”? Essa non fa altro che seguire una sequenza di operazioni e produrre un risultato.

Non possiamo cadere nella trappola della ingenua antropomorfizzazione delle tecnologie. Immaginare un giudice automa che “prende decisioni” è fuorviante, perché ciò che la tecnologia produce non sono *decisioni*, ma *risultati* derivanti da processi algoritmici. E qui risiede il vero problema: un giudice automa non garantirebbe affatto l'imparzialità che alcuni pretendono possa raggiungere.

Le decisioni giudiziarie richiedono interpretazione, empatia e valutazione del contesto: elementi che un algoritmo, per quanto avanzato, non potrà mai cogliere. Delegare le “decisioni” alla tecnologia significherebbe privare il diritto della sua componente umana e trasformare un processo complesso in una mera esecuzione meccanica di regole. Ma, soprattutto, significherebbe svuotare il diritto del suo stesso significato, negandone la sua più vera e profonda essenza: un artificio creato *dagli* esseri umani *per* gli esseri umani, *necessariamente imperfetto* perché sempre teso al conseguimento di quell'irraggiungibile fine che chiamiamo *giustizia*.

¹⁶ C. Schmitt, *Il compimento della Riforma*, in *Scritti su Thomas Hobbes*, Giuffrè, Milano, 1986, pp. 159-190.