FRANCESCO MASERA

INTELLIGENZA ARTIFICIALE

COME FUNZIONA DAVVERO (GUIDA COMPLETA PER CHI PARTE DA ZERO)

Prefazione di

PAOLO SAVONA

Independently published

ISBN
979-82-8406-640-9

EDIZIONE
ANSEDONIA 6 APRILE 2026

Ai miei genitori,
che mi hanno saputo guidare nelle scelte della mia vita,
sperando di saper fare altrettanto con i miei figli

INDICE

PREFAZIONE DI PAOLO SAVONA

Questo lavoro di Francesco Masera sull'Intelligenza Artificiale (IA, per brevità) merita di essere usato come libro di testo per corsi di laurea su una materia che comincia ad apparire tra i corsi delle nostre Università. Tuttavia, data l'ormai larga diffusione delle tecniche di analisi IA della realtà, l'insegnamento dovrebbe estendersi almeno alle scuole medie superiori, fornendo ai professori lo strumento per aggiornarsi. Senza nulla togliere alla simpatica presentazione di una materia noiosa (termine mutuante dell'economia come "scienza triste"), gli studenti ai quali il testo è dedicato non sono solo quelli curiosi, che già conoscono gran parte del contenuto, magari non con la precisione offerta dall'Autore, ma tutti, che lo vogliano o meno. Perché non basta più saper leggere e scrivere e fare di conto, ma è ora necessario anche usare le tecniche IA, senza le quali ci si ritroverebbe ai margini della vita economica, che chiede queste conoscenze, se non proprio della vita sociale, dato che tutti avranno a che fare quotidianamente con esse. Una volta impossessatisi della logica e del linguaggio informatico non vi è per essi niente di così impegnativo da impedire una tale conoscenza.

Francesco Masera è una personificazione dell'evoluzione della conoscenza logico-matematica essendo la terza generazione di una Famiglia il cui Nonno era Consigliere economico della Banca d'Italia che faceva buon uso del linguaggio corrente sulla traccia dell'insegnamento di Luigi Einaudi e il cui Padre ha eccelso nell'uso del linguaggio matematico applicato all'economia, essendo allievo di John Richard Hicks, un grande pensatore le cui idee dovettero competere quelle di John Maynard Keynes. Francesco jr. maturò la sua preparazione in una università che allora ignorava l'IA, ma era curioso, come lui stesso afferma, di una curiosità che lo ha spinto a interessarsi del linguaggio informatico, alla cui diffusione ha già contribuito con un precedente lavoro.

Poiché la consultazione dell'indice di un libro è la prima guida, quello sotto gli occhi del lettore parte dall'esame dell'idea di "macchina pensante" – che aleggiava nell'aria del XX Secolo ed ebbe nella macchina di Alan Turing la prima manifestazione di successo decriptando la macchina Enigma tedesca, che contribuì in modo significativo alla vittoria delle Forze Alleate contro il nazismo – e passa a esaminare come funzionano i "cervelli digitali", per approdare ai "linguaggi macchina", sempre più ampi e raffinati, ossia i veicoli di trasmissione dei prodotti della prodigiosa capacità computazionale dell'incontro tra i progressi dei computer e quelli dell'IA, grazie a una crescente disponibilità di dati, che definisce il "carburante della macchina". Infine, si sofferma in dettaglio sul linguaggio macchina più diffuso, il Python, dalla cui conoscenza gli studenti curiosi e i prof volenterosi devono muovere se vogliono entrare nell'universo magico che il filosofo Luciano Floridi ha battezzato Infosfera.

Nella parte finale del lavoro di Masera irrompono giustamente i problemi etici, quelli che riguardano il comportamento dei grandi attori globali, gli effetti sul mondo del lavoro e i modi per capire i rischi sanciti da ripetute Dichiarazioni dei diritti dell'uomo, tutti temi che, secondo la mia valutazione, vanno tenuti presenti, ma non devono essere usati per ritardare la conoscenza e l'uso dell'IA che il manuale si è prefisso di raggiungere.

Un'ultima considerazione. L'Intelligenza Artificiale fornisce il meglio di sé stessa se lavora 24 ore su 24, alimentata da un carburante in forma di dati che affluiscono a milioni e milioni in continuazione, che vanno colti come ci ha insegnato Eraclito nel momento in cui "tutto scorre" (*Pantha rei*), capacità che i modelli econometrici non hanno. Poiché in parallelo migliaia di ricercatori perfezionano macchine e linguaggi, quando uno scritto è finito, gli argomenti in esso contenuti sono già in parte invecchiati e il manuale andrebbe rivisto in continuazione; non a caso le innovazioni tecnologiche si trovano più su siti web o su riviste che escono su base infra-annuale piuttosto che su libri, la cui manifattura, nonostante i progressi tipografici, chiede più tempo. Tra le obsolescenze causate dall'IA vi sono quindi anche i libri, ma ciò non significa che essi sono inutili, perché "marcano la storia" dell'evoluzione informatica, sempre utile da conoscere. Ad esempio, questo libro incorpora il progresso dell'IA auto-generativa, come la ChatGPT di Open AI, ma non che cosa accadrà quando si metteranno a punto i computer quantistici, il cui primo segno sono le tecniche di quantum *annealing* (in italiano "ricomposizione quantistica"), una scorciatoia efficace per affrontare problemi complessi di ottimizzazione. Ai miei più bravi allievi ho sempre consigliato di terminare un lavoro quando esso raggiungeva la

frontiera della conoscenza corrente, senza porsi il problema del suo spostamento, perché non raggiungerebbe mai il momento di pubblicarlo.

È il progresso, caro lettore, e occorre prendere atto che esso non si potrà mai fermare.

Paolo Savona

Professore emerito di Politica Economica
Luiss Guido Carli

COME USARE QUESTO LIBRO

Questo non è un manuale tecnico. È una mappa per esplorare l'intelligenza artificiale: capire cos'è, come funziona, e perché ci riguarda tutti.

Il libro è pensato per accompagnare il lettore in un viaggio tra storia, tecnologie, applicazioni, rischi e prospettive future dell'IA. Ogni capitolo è indipendente, ma segue un filo logico: puoi leggerli in ordine oppure partire da quello che ti incuriosisce di più.

Cosa troverai in ogni capitolo:

- una spiegazione chiara dei concetti principali, con esempi concreti e collegamenti al mondo reale;
- box di approfondimento per i più curiosi: brevi incursioni in temi avanzati o particolari. Sono facoltativi: puoi saltarli senza perdere il filo;
- una sezione finale con "Spunti per riflettere", domande aperte per stimolare il confronto;
- domande di comprensione, utili per verificare se hai colto i concetti chiave;
- in fondo al libro trovi tutte le soluzioni, per ripassare o per confrontarti con altri.
- nei capitoli più pratici (come il Capitolo 5 "LLM in pratica" e il Capitolo 12 "IA a scuola"), troverai anche attività guidate, *prompt* da provare e progetti da realizzare in autonomia o in gruppo.

Le appendici tecniche

Se hai interesse a capire meglio "cosa c'è sotto il cofano" dei modelli di intelligenza artificiale, in fondo al libro trovi una serie di Appendici tecniche pensate proprio per chi vuole approfondire gli aspetti più strutturali.

Non servono competenze avanzate di matematica o informatica: ogni spiegazione è pensata per essere accessibile, con esempi chiari e formule guidate passo passo.

Le appendici sono pensate come una cassetta degli attrezzi per i più curiosi, per chi vuole spingersi un po' oltre.

Questo è un "libro vivo"

Come osserva il professor Savona nella prefazione, la materia è in tumultuosa evoluzione. Un libro, nel momento in cui viene stampato, è già vecchio. Proprio per questa ragione è nato Libro Vivo IA (librovivoIA.it): un sito pensato come estensione naturale di questo libro. Lì troverai costanti aggiornamenti sui temi trattati e le principali novità. Un libro, per sua natura, fotografa un momento; il sito è il modo in cui quel momento continua a vivere e ad aggiornarsi. Ti consiglio di visitarlo regolarmente: è pensato per fare di questo testo non un punto d'arrivo, ma un punto di partenza.

PREMESSA: STORIE VERE E PERCORSI PER COMINCIARE

Nel mondo in cui viviamo, l'innovazione arriva sempre più spesso da giovani che, con coraggio e passione, decidono di non aspettare "il momento giusto", ma di agire subito. Alcuni iniziano da un'idea semplice, magari nata tra i banchi di scuola o durante una ricerca online, e riescono a trasformarla in qualcosa di straordinario. La tecnologia – e in particolare l'intelligenza artificiale – è uno degli strumenti più potenti che questi giovani hanno a disposizione. Ma da sola non basta: servono impegno, studio e curiosità.

Prendiamo il caso di Boyan Slat. Quando aveva solo 16 anni, durante una vacanza in Grecia, ha visto più plastica che pesci in mare. Due anni dopo, a 18 anni, ha fondato The Ocean Cleanup, un'organizzazione che oggi impiega oltre 120 persone, ha rimosso più di 50.000 tonnellate di plastica da oceani e fiumi (The Ocean Cleanup, 2026) e ha ricevuto finanziamenti per oltre 200 milioni di dollari, tra cui una donazione da 121 milioni nel 2025 da parte di The Audacious Project (The Ocean Cleanup, 2025). Boyan è stato nominato "European of the Year" nel 2017, ha parlato al World Economic Forum e Forbes lo ha inserito tra i "30 under 30" più influenti e innovativi del mondo, nella categoria scienza ed energia.

Un'altra storia incredibile è quella di Melanie Perkins, che ha fondato la startup Canva a 26 anni. Partita con un'idea nata mentre insegnava grafica all'università, ha costruito una piattaforma oggi usata da oltre 265 milioni di utenti attivi in più di 190 Paesi (Canva, 2026). Canva è diventata una delle aziende tech più grandi dell'emisfero sud, con una valutazione di 42 miliardi di dollari e un fatturato annuo di 4 miliardi (Gurman, 2026). Melanie è tra le donne più ricche d'Australia: secondo Forbes (2026), il suo patrimonio personale è stimato in circa 7,6 miliardi di dollari. Eppure, continua a vivere in modo semplice, a reinvestire nel progetto e a promuovere l'educazione digitale.

E poi c'è Brittany Wenger, che a 17 anni – dopo aver studiato da autodidatta programmazione in Python – ha creato una rete neurale artificiale in grado

di diagnosticare il cancro al seno con un'accuratezza del 99,1%. Il suo sistema, chiamato Cloud4Cancer, è stato addestrato su oltre 7 milioni di righe di dati e successivamente esteso anche alla diagnostica della leucemia tramite profili di espressione genetica (Wenger, 2012). Brittany ha vinto la Google Science Fair nel 2012, sbaragliando concorrenti da tutto il mondo. La rivista Time l'ha inserita tra le "30 persone sotto i 30 che stanno cambiando il mondo", e il suo lavoro è stato presentato alla Casa Bianca durante l'amministrazione Obama. Oggi si specializza in oncologia pediatrica presso la Icahn School of Medicine del Mount Sinai a New York.

Tutte queste storie hanno qualcosa in comune. Non si tratta di "geni irraggiungibili", ma di giovani con la voglia di capire, di mettersi in gioco, di studiare. Nessuno di loro si è limitato a usare la tecnologia passivamente. Al contrario: l'hanno capita, l'hanno studiata, l'hanno usata per risolvere un problema reale.

Ecco il punto fondamentale: la tecnologia, e in particolare l'intelligenza artificiale, è un acceleratore. Se hai una buona idea e le competenze giuste, può portarti lontano. Ma è solo uno strumento. È la tua mente che fa la differenza. Ed è lo studio – a volte difficile, a volte noioso, ma sempre prezioso – che ti permette di trasformare quel potenziale in qualcosa di reale.

Imparare a programmare, anche solo le basi, è un modo per capire come funzionano gli algoritmi che oggi guidano gran parte del nostro mondo. Python, ad esempio, è un linguaggio semplice e potente, usato in tutto il mondo per sviluppare applicazioni di intelligenza artificiale. Ma non serve per forza diventare programmatori: si può studiare ingegneria, scienze ambientali, economia, filosofia, design, e trovare comunque il modo di usare l'IA per risolvere problemi reali. L'importante è avere la voglia di imparare e la curiosità di fare domande.

Perché alla fine, l'innovazione non è qualcosa che accade solo nei laboratori o nelle grandi aziende tecnologiche. Può cominciare anche nella tua stanza, davanti a un computer, con una domanda che nessuno ha ancora avuto il coraggio di porsi.

Questo libro nasce per accompagnarti in un viaggio: non per trasformarti in un esperto di tecnologia, ma per aiutarti a capire cos'è davvero l'intelligenza artificiale, come può essere usata in modo consapevole, e perché imparare a conoscerla – oggi più che mai – è il primo passo per costruire il tuo futuro con intelligenza, creatività e responsabilità.

DALL'IDEA DI MACCHINA PENSANTE ALL'IA CHE CREA

Nella metà del Novecento, in un'epoca attraversata da guerre, scoperte scientifiche e nuove frontiere tecnologiche, un gruppo di visionari iniziò a coltivare un'idea audace: creare macchine capaci di pensare.

Alan Turing, geniale matematico britannico, aveva già ipotizzato negli anni Quaranta che il pensiero umano potesse essere descritto come un processo meccanico. Nel suo celebre saggio del 1950, "*Computing Machinery and Intelligence*", pose una domanda provocatoria: "Le macchine possono pensare?" e propose un esperimento, oggi noto come Test di Turing, per stabilire se una macchina potesse simulare l'intelligenza umana in modo indistinguibile.

Poco dopo, nel 1956, un celebre seminario a Dartmouth, negli Stati Uniti, sancì ufficialmente la nascita dell'Intelligenza Artificiale (IA) come campo di ricerca autonomo.

I primi progetti si basavano su una visione simbolica dell'intelligenza: si pensava che bastasse definire regole logiche e rappresentare la conoscenza in modo esplicito perché una macchina potesse ragionare.

Nacquero i sistemi esperti, capaci di diagnosticare malattie o risolvere problemi matematici complessi. Allen Newell e Herbert Simon svilupparono i primi programmi in grado di dimostrare teoremi matematici, come il celebre Logic Theorist.

Il cammino non fu lineare. L'entusiasmo iniziale lasciò presto il posto a disillusioni: le limitazioni *hardware* e l'eccessivo ottimismo sulle capacità delle macchine provocarono i primi "inverni dell'IA", periodi di drastico calo degli investimenti e della fiducia.

Nonostante tutto, la ricerca non si fermò. Negli anni successivi, nuovi approcci, come il *Machine Learning* e le reti neurali, avrebbero permesso all'Intelligenza Artificiale di rinascere più forte che mai, avvicinandosi sempre di più a quello che Turing, McCarthy, Minsky e gli altri pionieri avevano solo intravisto: un'intelligenza artificiale capace di imparare, evolversi e trasformare il mondo.

Il punto di svolta per il grande pubblico arrivò il 30 novembre 2022, quando OpenAI rilasciò ChatGPT: in soli cinque giorni raggiunse un milione di utenti, e in due mesi cento milioni, rendendolo il servizio con la crescita più rapida della storia. Per la prima volta, chiunque poteva conversare con un'Intelligenza Artificiale in linguaggio naturale, e il mondo si accorse che qualcosa di profondo era cambiato.

1.1 I precursori dell'IA

L'Intelligenza artificiale è una disciplina relativamente giovane, che ha iniziato a prendere forma nel secolo scorso, ma che ha radici che affondano nella storia della filosofia, della matematica, della logica, della psicologia e dell'ingegneria.

L'idea di dotare le macchine di capacità di ragionamento affonda le sue radici nella logica formale, sviluppata da Aristotele oltre duemila anni fa. Il filosofo greco elaborò un metodo strutturato di ragionamento basato sull'induzione – il passaggio da osservazioni particolari a principi generali – e sulla deduzione – il passaggio da principi generali a conclusioni particolari.

Ad esempio, osservando che tutti gli uccelli visti finora volano, possiamo indurre la regola generale che "tutti gli uccelli volano"; applicando poi questa regola al caso specifico del cigno, dedurremmo che anche il cigno vola.

Al centro del metodo aristotelico vi erano le figure di sillogismo, schemi di argomentazione composti da tre proposizioni: una premessa maggiore, una premessa minore e una conclusione. Questo approccio non garantiva la verità dei contenuti – dipendenti dalla validità delle premesse – ma assicurava la correttezza della forma logica del ragionamento.

Questa visione strutturata e formale del pensiero è stata il primo grande passo verso la possibilità di rappresentare il ragionamento umano in modo meccanico: un'ispirazione diretta per la nascita dell'Intelligenza Artificiale, che si basa proprio su modelli di inferenza logica, apprendimento automatico e deduzione computazionale.

Approfondimento: i grandi pensatori: tra filosofi e matematici

Se Aristotele aveva gettato le basi della logica formale, altri grandi pensatori nei secoli successivi iniziarono a interrogarsi sulla possibilità di replicare il pensiero e le funzioni vitali attraverso macchine artificiali.

Nel 1637, il matematico e filosofo francese René Descartes, nel suo celebre "Discorso sul metodo", propose una visione radicale: gli animali, sosteneva, non sono altro che macchine complesse, governate da leggi meccaniche, e nulla impediva all'uomo di costruire artefatti capaci di imitare le loro funzioni vitali.

L'idea, pur lontana dai moderni concetti di intelligenza artificiale, introduceva una nozione fondamentale: il comportamento intelligente poteva, almeno in teoria, essere scomposto, analizzato e riprodotto (Descartes, 1993).

Pochi decenni dopo, un altro gigante del pensiero, il matematico tedesco Gottfried Wilhelm Leibniz, avanzò una proposta ancora più concreta. Nel suo saggio del 1684, "Dissertatio de Arte Combinatoria", Leibniz immaginò la creazione di una macchina calcolatrice universale, il "calculus ratiocinator", capace non solo di eseguire operazioni aritmetiche, ma anche di svolgere forme elementari di ragionamento logico. Basata su un sistema binario – quello stesso sistema che oggi regge l'informatica moderna – la macchina di Leibniz non fu mai realizzata completamente, ma rappresentò una straordinaria anticipazione dei concetti alla base dei computer e, più tardi, delle macchine intelligenti (Leibniz, 1684).

Se Aristotele, Descartes e Leibniz avevano tracciato le prime intuizioni sulla possibilità di meccanizzare il pensiero, furono i matematici, i logici e gli informatici tra il XIX e il XX secolo a gettare le basi teoriche e pratiche su cui si sarebbe costruita l'Intelligenza Artificiale moderna.

Tra questi pionieri spicca George Boole, matematico e filosofo inglese, che nel 1854 pubblicò "An Investigation of the Laws of Thought". In quest'opera, Boole introdusse l'algebra booleana, un sistema di calcolo basato su due valori, vero e falso, che sarebbe diventato il fondamento della logica digitale e dei circuiti elettronici, indispensabili per la nascita dei moderni computer (Boole, 1854).

Agli inizi del XX secolo, un altro gigante del pensiero logico, Bertrand Russell, insieme ad Alfred North Whitehead, sviluppò nel monumentale "Principia Mathematica" (1910) un linguaggio formale capace di rappresentare e manipolare proposizioni e predicati. La loro ambizione era creare una base matematica rigorosa per tutta la conoscenza, anticipando la necessità, che troverà piena espressione nell'IA, di formalizzare il sapere in modo sistematico e computabile (Russell et al., 1910-1913).

Poco dopo, Kurt Gödel, giovane matematico austriaco, con i suoi teoremi di incompletezza del 1931, dimostrò un limite fondamentale: esistono affermazioni vere che nessun sistema formale, per quanto potente, può dimostrare al suo interno. Gödel sollevava così interrogativi cruciali che ancora oggi accompagnano la riflessione sull'Intelligenza Artificiale: può una macchina, per quanto sofisticata, comprendere e risolvere ogni problema? (Gödel, 1931).

Non meno visionario fu Charles Babbage, che già nel XIX secolo progettò la Macchina Analitica, considerata il primo vero concetto di computer

programmabile. Sebbene la macchina non fu mai completata durante la sua vita, le intuizioni di Babbage aprirono la strada all'idea che dispositivi meccanici potessero eseguire istruzioni complesse in sequenza logica: il principio stesso del calcolo automatico (Menabrea, 1842).

Colui che sintetizzò secoli di idee e intuizioni in un progetto teorico e pratico concreto, fu Alan Turing. Matematico e informatico inglese, Turing è considerato il padre dell'Intelligenza Artificiale per aver formalizzato i principi del calcolo meccanico con la macchina di Turing, fornendo così il modello astratto di ciò che significa "computare". Turing pose la domanda che ancora oggi guida la ricerca nel campo: "Le macchine possono pensare?".

Il lavoro di Turing e la sua vita sono stati così rilevanti che hanno ispirato anche un film, intitolato "*The Imitation Game"*, uscito nel 2014. Il film racconta le vicende di Turing durante la guerra, quando lavorava alla decifrazione dei codici nazisti. Una delle frasi più celebri del film è pronunciata da Turing, interpretato da Benedict Cumberbatch, quando dice: "*sometimes it is the people who no one imagines anything of who do the things that no one can imagine*".

Turing nel 1950, nel già menzionato articolo intitolato "*Computing Machinery and Intelligence"* (Turing, 1950), ha concepito quello che è rimasto famoso come il "test di Turing". Nell'articolo Turing pose la questione se le macchine possano pensare, sostenendo che non ha senso porsi la domanda, perché il concetto di pensiero è troppo vago e soggettivo, e dipende dalla definizione che se ne dà. Invece, secondo Turing, ha senso chiedersi se una macchina possa agire in modo intelligente, cioè se possa comportarsi in modo tale da ingannare un osservatore umano. Il test di Turing è stato quindi concepito come un esperimento mentale, per verificare se una macchina possa imitare con successo il linguaggio e il pensiero umani.

Il test di Turing, nella sua versione originale, prevede che un interrogatore umano si trovi in una stanza, dotato di una tastiera e di uno schermo. Da una parte, c'è una macchina, che può essere un computer o qualsiasi altro tipo di dispositivo capace di ricevere e inviare messaggi in linguaggio naturale. Dall'altra, c'è una persona, che funge da controllo. L'interrogatore non sa quale sia la macchina e quale la persona, e il suo compito è di scoprirlo, ponendo domande a entrambi attraverso la tastiera. La macchina e la persona possono rispondere alle domande come preferiscono, cercando di convincere l'interrogatore della propria identità. Il test si conclude quando l'interrogatore esprime il proprio giudizio, indicando quale dei due interlocutori crede sia la macchina e quale la persona. Se l'interrogatore sbaglia, cioè se indica la macchina come la persona e

viceversa, allora si dice che la macchina ha superato il test di Turing, dimostrando di avere una capacità di comunicazione paragonabile a quella umana.

Il test di Turing è ancora oggi considerato una pietra miliare nella storia dell'Intelligenza Artificiale, e un riferimento per la ricerca e lo sviluppo di sistemi intelligenti.

1.2 La nascita dell'IA

Il termine Intelligenza Artificiale è nato ufficialmente nel 1956, quando un gruppo di ricercatori, tra cui John McCarthy, Marvin Minsky, Claude Shannon e Herbert Simon, si riunì a Dartmouth, negli Stati Uniti, per discutere e studiare le possibilità di creare macchine intelligenti. In quell'occasione, McCarthy coniò il termine "Intelligenza Artificiale", definendola come "la scienza e l'ingegno di creare entità intelligenti" (McCarthy et al., 1955).

Dalla sua nascita ufficiale negli anni Cinquanta, l'Intelligenza Artificiale ha attraversato fasi alterne di grande entusiasmo e profonde crisi, alternate a momenti di riflessione e ripensamento. La sua storia è un susseguirsi di sogni audaci, scoperte rivoluzionarie e ostacoli apparentemente insormontabili.

La prima fase, nota come "età dell'oro" dell'IA, prese avvio con il celebre incontro di Dartmouth del 1956. In quel periodo i ricercatori concentrarono i loro sforzi sullo sviluppo di sistemi basati sulla logica simbolica: modelli in cui la conoscenza veniva rappresentata attraverso simboli, regole e fatti, e i problemi venivano risolti mediante inferenze formali. Nacquero così i programmi di gioco degli scacchi, i sistemi esperti in ambiti specifici e i programmi di dimostrazione automatica di teoremi.

Nonostante i successi iniziali, emersero presto i limiti di questi sistemi, incapaci di gestire efficacemente conoscenza comune, incertezze e problemi complessi e dinamici.

Le difficoltà furono analizzate in profondità nel rapporto di James Lighthill del 1973, commissionato dal governo britannico per valutare lo stato della ricerca nel campo della IA. Il giudizio fu severo: Lighthill concluse che "in nessun ambito dell'Intelligenza Artificiale si registravano scoperte concrete", sostenendo che gli obiettivi dell'IA fossero irraggiungibili, poiché la mente umana non opera secondo rigide regole logiche, ma attraverso processi biologici complessi e un linguaggio naturalmente ambiguo (Lighthill, 1973).

La pubblicazione del rapporto segnò l'inizio di un lungo periodo di disillusione noto come "inverno dell'IA", caratterizzato da un drastico calo degli investimenti e dell'interesse accademico. Nonostante la crisi, proprio in quegli anni nacquero nuove linee di ricerca, in particolare l'Intelligenza Artificiale Connessionista, basata sulle reti neurali artificiali: modelli ispirati alla struttura del

cervello umano, capaci di apprendere dai dati e di generalizzare, aprendo nuove prospettive.

La terza fase, la cosiddetta "rinascita", prese avvio nei primi anni Ottanta dal lancio del progetto giapponese della *Fifth Generation Computer Systems* (FGCS), annunciato nel 1981 dal Japan Information Processing Development Center (JIPDEC) (Tohru Moto-oka, 1982). L'obiettivo era ambizioso: costruire supercomputer intelligenti, dotati di parallelismo massiccio, comprensione del linguaggio naturale e capacità di ragionamento logico. Questo progetto non solo rilanciò la ricerca globale, stimolando la competizione internazionale, ma beneficiò anche dei progressi nell'informatica, nell'elettronica e nelle telecomunicazioni. Le nuove risorse tecnologiche permisero l'applicazione dell'IA in settori come robotica, visione artificiale, riconoscimento vocale, traduzione automatica, ricerca sul web, medicina, finanza e persino arti creative, gettando le basi per un'IA sempre più pervasiva.

La quarta fase, la "rivoluzione", quella che stiamo vivendo oggi, la possiamo far partire simbolicamente nel 2015, con la pubblicazione di un articolo fondamentale sul *Deep Learning*, che illustrava le straordinarie potenzialità delle reti neurali profonde (LeCun et al. 2015). Grazie a questa tecnica, l'IA raggiunse traguardi che pochi anni prima sembravano irraggiungibili. Tra le imprese più celebri si ricordano la vittoria del sistema IBM Watson nel quiz televisivo Jeopardy! (IBM Research, 2013), il trionfo di AlphaGo contro i migliori giocatori di Go (Google DeepMind, 2020), e il successo di AlphaZero, capace di apprendere da zero e dominare giochi complessi come scacchi, Go e shogi.

Parallelamente, l'IA iniziò a generare immagini, testi, musiche e *video* di qualità sorprendente, dimostrando capacità di percezione, produzione ed espressione sempre più raffinate.

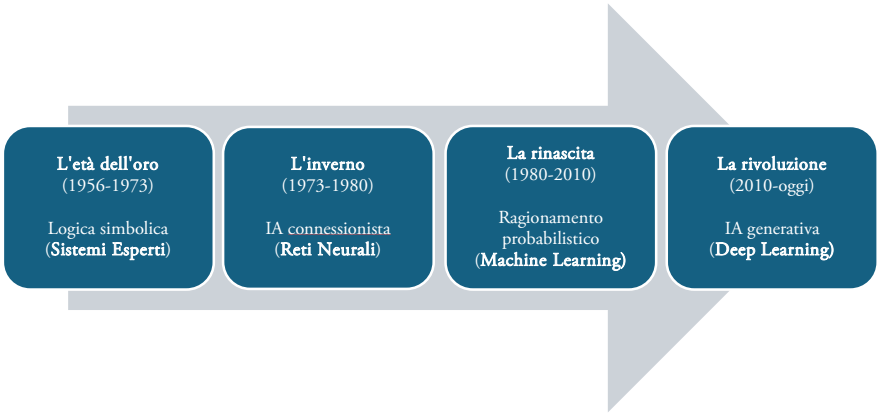

Figura 1: Fasi evolutive dello sviluppo dell'Intelligenza Artificiale

In pochi anni, l'Intelligenza Artificiale non solo ha superato molte delle aspettative iniziali, ma è diventata un motore di trasformazione globale, influenzando profondamente ogni aspetto della società contemporanea.

Nel prossimo capitolo riprenderemo l'analisi delle quattro fasi di sviluppo dell'Intelligenza Artificiale, soffermandoci in particolare sulle principali innovazioni che hanno caratterizzato ciascun periodo. Prima di addentrarci in questo approfondimento, però, è utile offrire una panoramica sulla portata delle trasformazioni introdotte e sui risultati concreti raggiunti dalle applicazioni della IA ad oggi.

1.3 Dove siamo oggi

Dopo decenni di ricerca e momenti di entusiasmo alternati a profonde delusioni, l'Intelligenza Artificiale è uscita dai laboratori. Oggi non solo sa "pensare" ma anche vedere, scrivere, creare, diagnosticare, perfino imparare da sola. In questa sezione esploreremo alcune delle sue applicazioni di frontiera più affascinanti e concrete.

Parallelamente, l'IA è diventata un asse strategico per governi, industrie e centri di ricerca: la competizione globale sull'innovazione in questo settore è intensa, con investimenti miliardari e politiche pubbliche mirate a sostenere lo sviluppo, la regolamentazione e l'adozione responsabile delle tecnologie emergenti.

In questo scenario dinamico e in continua evoluzione, l'Intelligenza Artificiale si configura non più soltanto come una tecnologia, ma come una forza trainante del cambiamento culturale, destinata a ridefinire il nostro modo di lavorare, di apprendere, di creare e di vivere.

Mustafa Suleyman, uno dei fondatori di DeepMind, la società che ha realizzato il sistema AlphaGo che ha superato il livello umano nel gioco del Go, nel suo libro "*The coming wave*" parla della IA come della quarta onda di innovazione che influenzerà l'evoluzione dell'umanità, dopo la scoperta del fuoco, la rivoluzione agricola e quella industriale. L'Intelligenza Artificiale, sostiene Suleyman, ha il potenziale di trasformare radicalmente il modo in cui viviamo, lavoriamo e interagiamo (Suleyman et al.,2023).

Per comprendere appieno la portata dell'Intelligenza Artificiale contemporanea, è utile osservare alcune delle sue applicazioni più avanzate. Nei prossimi paragrafi analizzeremo esempi concreti che mostrano come l'IA stia rivoluzionando il modo di comunicare, creare, percepire e programmare, offrendo uno spunto sulle straordinarie potenzialità già oggi a disposizione, e su quelle che ci attendono nel prossimo futuro.

L'Intelligenza Artificiale applicata alla comprensione e alla generazione del linguaggio naturale (*Natural Language Processing* – NLP e *Natural Language Generation* – NLG) rappresenta una delle aree di sviluppo più incredibili.

Modelli come GPT di OpenAI, Claude di Anthropic, Gemini di Google e Llama di Meta sono oggi in grado di generare testi fluidi, coerenti e contestualmente rilevanti, raggiungendo livelli di comprensione linguistica che fino a pochi anni fa sembravano appannaggio esclusivo degli esseri umani.

Uno studio del 2024 di Cameron R. Jones e Benjamin K. Bergen ha sottoposto diversi modelli di linguaggio a una versione randomizzata del test di Turing. In quell'esperimento GPT-4 è stato giudicato "umano" nel 54% delle conversazioni, superando nettamente *chatbot* precedenti ma rimanendo leggermente al di sotto del livello degli interlocutori umani. Questo lavoro è stato considerato una delle prime dimostrazioni empiriche robuste della capacità dei moderni modelli linguistici di avvicinarsi alla soglia di indistinguibilità conversazionale rispetto agli esseri umani (Jones & Bergen, 2024).

Nel 2025 gli stessi autori hanno replicato l'esperimento con un protocollo ancora più rigoroso basato sul test di Turing a tre partecipanti, più vicino al gioco dell'imitazione originale descritto da Alan Turing. In questa configurazione un giudice umano conversa simultaneamente con un umano e con una macchina e deve stabilire quale dei due sia la persona reale. In queste condizioni, il modello GPT-4.5 è stato identificato come umano nel 73% delle conversazioni, addirittura più spesso del vero interlocutore umano. Gli autori interpretano questi risultati come una delle prime evidenze sperimentali di superamento del test di Turing in un contesto interattivo controllato (Jones & Bergen, 2025).

Tuttavia, diversi studi recenti invitano alla cautela nell'interpretazione di questi risultati. Alcune ricerche mostrano che il successo dei modelli linguistici nel test dipende in larga misura dalla loro capacità di imitare lo stile linguistico e le dinamiche socio-emotive della conversazione, più che da una reale comprensione dei contenuti. In esperimenti basati su trascrizioni di dialoghi, ad esempio, sia gli esseri umani sia i modelli linguistici tendono a identificare erroneamente i testi prodotti da GPT-4 come umani più spesso di quelli scritti da persone reali (Shin et al., 2024).

Altri lavori propongono di affiancare o sostituire il Turing test con metodi più quantitativi – talvolta definiti *computational Turing tests* – che confrontano sistematicamente le caratteristiche linguistiche dei testi generati dalle macchine con quelle dei testi umani su larga scala. Queste analisi mostrano che, nonostante l'elevata plausibilità conversazionale, le produzioni dei modelli linguistici rimangono spesso distinguibili da quelle umane quando vengono analizzate con strumenti statistici e computazionali (Wang & Li, 2025).

Per questo motivo molti ricercatori ritengono oggi che il superamento del test di Turing rappresenti soprattutto un traguardo nella simulazione linguistica e sociale, più che una prova definitiva di intelligenza comparabile a quella umana.

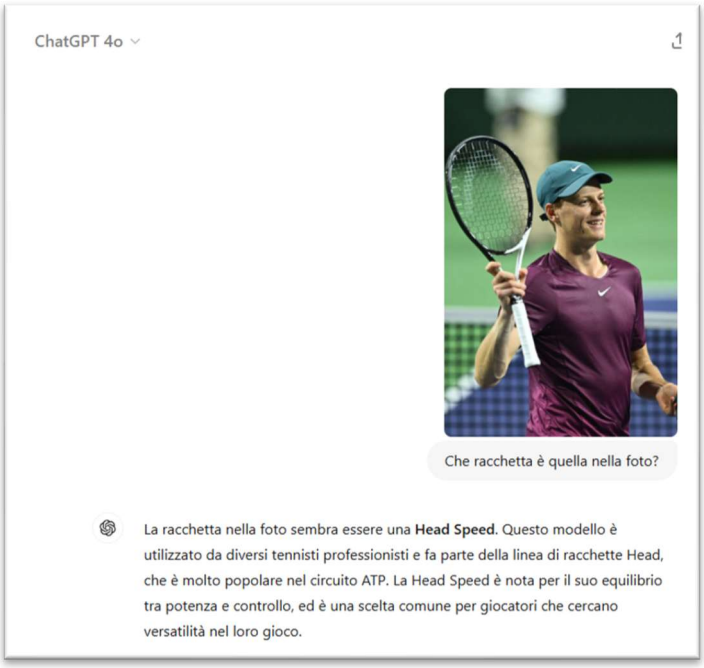

Figura 2: Esempio di query "multimodale" a ChatGPT.

I modelli più recenti sono capaci di gestire contemporaneamente diversi formati di *input* – testo, immagini, *audio* e persino *video* – aprendo la strada a interazioni sempre più naturali e multimodali (Figura 2). Nel capitolo 3 approfondiremo come funzionano questi modelli di linguaggio.

L'IA è oggi anche capace di creare *video* realistici partendo da semplici descrizioni testuali, immagini o tracce *audio*. Negli ultimi anni sono stati sviluppati diversi sistemi in grado di trasformare *prompt* testuali in sequenze animate Sora di OpenAI, Veo 2 di Google, Kling di Kuaishou e Runway Gen-3. Questi sistemi riescono a mantenere una buona coerenza temporale tra i fotogrammi e a simulare movimenti complessi, mostrando progressi molto rapidi nel realismo delle scene e nella qualità delle animazioni (Figura 3).

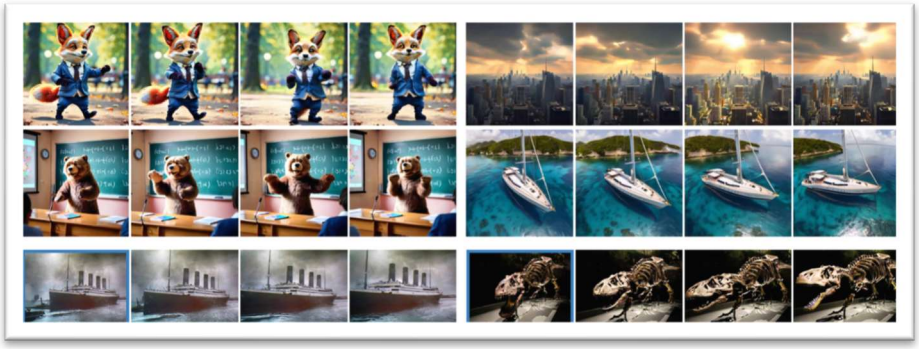

Figura 3: Esempi di video generati dall'IA.

La IA è oggi anche in grado di generare codice di programmazione. Partendo da richieste in linguaggio naturale i moderni sistemi di IA sono in grado di generare interi programmi funzionanti. Chiedendo ad esempio a Gemini "scrivi un programma in Python per creare un semplice gioco", ecco comparire il codice funzionante, corredato da una spiegazione sintetica degli obiettivi e delle regole operative (Figura 4).

```
import random

# Impostazioni del Gioco
numero_segreto = random.randint(1, 100)  # Genera un numero casuale tra 1 e 100
numero_tentativi = 5  # Numero massimo di tentativi

# Ciclo di gioco
for tentativo in range(1, numero_tentativi + 1):
  scommessa = int(input(f"Tentativo {tentativo} di {numero_tentativi}: Inserisci un

  if scommessa == numero_segreto:
    print(f"Hai indovinato! Il numero segreto era {numero_segreto}")
    break
  elif scommessa < numero_segreto:
    print("Il numero è più alto.")
  else:
    print("Il numero è più basso.")

if tentativo == numero_tentativi:
  print(f"Hai perso! Il numero segreto era {numero_segreto}")
```

Figura 4: Generazione di codice in Python da Gemini

Nel 2025 sono emersi anche i cosiddetti "agenti di codice": sistemi come Claude Code di Anthropic, Codex di OpenAI e Jules di Google che non si limitano a scrivere singole funzioni, ma possono autonomamente esplorare un progetto, pianificare modifiche, scrivere codice e verificarlo con i test. Ne parleremo nel Capitolo 4 dedicato all'*Agentic AI*.

Le applicazioni della IA alla ricerca scientifica hanno portato a risultati straordinari. AlphaFold, il sistema sviluppato da DeepMind, ha risolto un problema aperto da oltre cinquant'anni: prevedere con precisione la struttura tridimensionale delle proteine a partire dalla loro sequenza di aminoacidi. Durante la pandemia di COVID-19, AlphaFold ha dimostrato il suo potenziale pratico contribuendo a predire rapidamente la struttura di proteine chiave del virus SARS-CoV-2, accelerando la comprensione dei meccanismi d'infezione e supportando lo sviluppo di terapie e vaccini.

AlphaFold ha predetto la struttura di oltre 200 milioni di proteine, praticamente ogni proteina conosciuta dalla scienza, e i suoi dati sono liberamente accessibili alla comunità scientifica (DeepMind, 2024). Nel 2024, Demis Hassabis e John Jumper di DeepMind hanno ricevuto il Premio Nobel per la Chimica proprio per questo lavoro, la prima volta che un risultato dell'Intelligenza Artificiale viene riconosciuto con il massimo onore scientifico. Nello stesso anno, AlphaFold 3 ha esteso le predizioni a complessi multi-molecolari, incluse interazioni tra proteine, DNA, RNA e piccole molecole (Abramson et al., 2024), e a novembre il codice e i pesi del modello sono stati rilasciati per uso accademico. A fine 2025, l'articolo scientifico di AlphaFold 3 aveva già superato le 9.000 citazioni, e la sua adozione nella ricerca farmaceutica continua ad ampliarsi.

In ambito medico, i sistemi di IA raggiungono un'accuratezza diagnostica su scansioni mediche comparabile o superiore a quella dei medici specialisti in diverse aree (radiologia, dermatologia, oftalmologia). A fine 2025, la FDA americana ha autorizzato complessivamente 1.451 dispositivi medici basati su IA, di cui il 76% in radiologia (IntuitionLabs, 2026).

E che dire dei veicoli a guida autonoma. Sembra una cosa lontana nel tempo, eppure è già realtà. Google ha una società che si chiama Waymo, che ha avviato servizi commerciali di robo-taxi in dieci città americane: Phoenix, San Francisco, Los Angeles, Austin, Atlanta, Miami, Dallas, Houston, San Antonio e Orlando (TechCrunch, 2026). Queste auto completamente senza guidatore effettuano oltre 400.000 corse a settimana (Waymo, 2025), con l'obiettivo di superare il milione entro fine 2026. Nel febbraio 2026, Waymo ha raccolto un round di investimento da 16 miliardi di dollari e annunciato l'espansione in oltre 20 nuove città, incluse Tokyo e Londra – i primi mercati internazionali (Waymo, 2026). Tesla ha avviato nel 2025 ad Austin, in Texas, un servizio pilota di trasporto con veicoli dotati del sistema Full Self-Driving, inizialmente con un operatore di sicurezza a bordo. A inizio 2026, Tesla ha iniziato a integrare corse senza conducente nel servizio, con l'obiettivo di espandersi in altre sette città americane entro metà anno (InsideEVs, 2026).

Le prestazioni raggiunte dall'intelligenza artificiale sono ormai in diversi ambiti paragonabili a quelle umane e, in taluni casi, persino superiori. La Figura 5 illustra l'evoluzione di queste capacità negli ultimi dieci anni, attraverso benchmark che misurano competenze specifiche. Secondo lo Stanford AI Index Report (2025), il divario tra IA e prestazione umana su benchmark come MMLU, MATH e HumanEval si è ridotto a frazioni di punto percentuale, anche se su test più recenti e complessi – come Humanity's Last Exam e FrontierMath – le prestazioni dell'IA restano ancora molto al di sotto del livello umano (Stanford Institute for Human-Centered Artificial Intelligence, 2024).

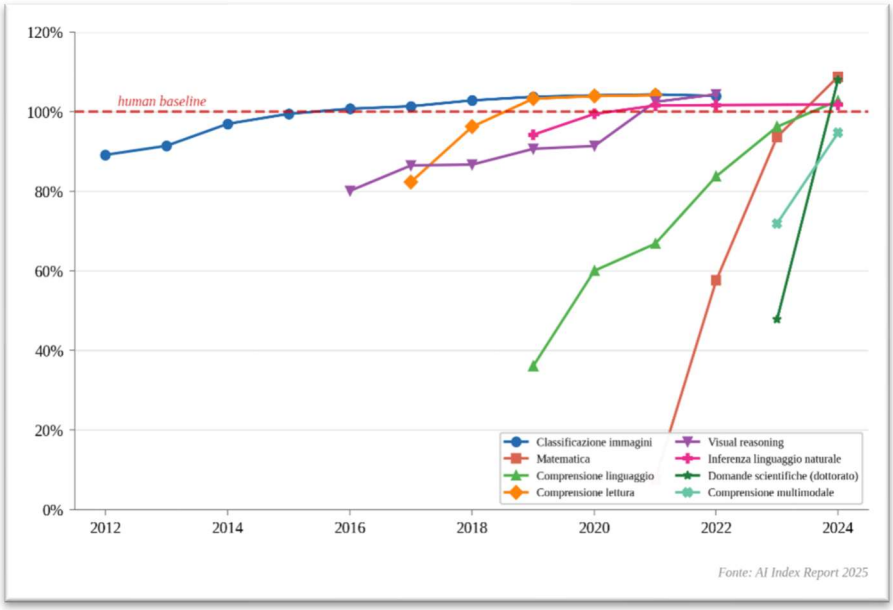

Figura 5: Benchmark prestazioni IA vs. prestazioni umane

Il problema, paradossalmente, è che i test esistenti non bastano più a misurare le capacità dei modelli più avanzati. La comunità scientifica sta sviluppando nuovi *benchmark* più sofisticati – come ARC-AGI, GPQA e FrontierMath – per continuare a valutare i progressi in aree come il ragionamento astratto, la conoscenza specialistica e la risoluzione di problemi matematici complessi. Una spinta decisiva in questa direzione è arrivata dai cosiddetti *reasoning models* – come o1 e o3 di OpenAI e DeepSeek-R1 – che nel 2024–2025 hanno introdotto tecniche di ragionamento passo-passo a tempo di inferenza, ottenendo risultati notevoli proprio su questi *benchmark* avanzati.

§§§

In questo capitolo abbiamo ripercorso il lungo cammino dell'Intelligenza Artificiale – dalle prime intuizioni di Turing e dei pionieri di Dartmouth, attraverso gli inverni e le rinascite, fino alle straordinarie applicazioni che oggi trasformano il nostro quotidiano. Abbiamo visto come l'IA sia passata da un'idea visionaria a una forza concreta, capace di superare il test di Turing, prevedere la struttura delle proteine, guidare veicoli senza conducente e generare testi, immagini e video con una qualità impensabile solo pochi anni fa.

Ma sapere *cosa* l'Intelligenza Artificiale è in grado di fare è solo il primo passo. La domanda che ora si impone è: *come* riesce a farlo? Che cosa si nasconde dietro queste capacità apparentemente prodigiose? Nel prossimo capitolo apriremo la "scatola nera" dell'IA per esplorarne i meccanismi interni – dagli algoritmi alle reti neurali, dai dati alle logiche di apprendimento – e scoprire cosa rende davvero possibile tutto questo.

Domande di comprensione

1. Chi è Alan Turing e quale fu il suo contributo fondamentale per lo sviluppo dell'IA?
2. Cosa si intende con "inverno dell'intelligenza artificiale"? Quali furono le cause?
3. Cosa caratterizzava i primi sistemi di IA simbolica e perché suscitarono tanto interesse?
4. Quali sono alcuni degli ambiti applicativi più innovativi dell'IA attuale descritti nel capitolo?
5. Che differenza c'è tra "comprendere" e "generare" per un'intelligenza artificiale?
6. Che cosa rappresenta il lancio di ChatGPT nel novembre 2022 nella storia dell'Intelligenza Artificiale? Perché è considerato un momento di svolta?

Trovi tutte le risposte nella sezione "Soluzioni" in fondo al libro.

DENTRO L'IA: COME FUNZIONANO I CERVELLI DIGITALI

Finora abbiamo guardato l'IA da fuori. Adesso è tempo di aprirla, come si fa con un vecchio computer o un robot, per vedere cosa la muove davvero: algoritmi, reti, dati e logiche di apprendimento. Siete pronti a entrare nella black box?

In questo capitolo proveremo a rispondere ad alcune domande fondamentali: cosa rende possibile l'IA? Quali strumenti permettono alle macchine di apprendere, ragionare, comunicare e persino creare? Cosa si nasconde all'interno della *black box* dell'Intelligenza Artificiale Generativa?

Per farlo, introdurremo i concetti di base che governano gli algoritmi, ovvero quelle sequenze di istruzioni che guidano il comportamento di ogni sistema di IA. Scopriremo come gli algoritmi possano essere classificati in base alla loro autonomia, alle modalità di apprendimento, al tipo di dati che elaborano e ai problemi che sono in grado di risolvere.

Esploreremo poi le principali tecniche utilizzate nel campo dell'IA: dai Sistemi Esperti alle reti neurali, dal *Machine Learning* al *Deep Learning*, fino ad arrivare alle applicazioni di *Natural Language Processing*.

Pur introducendo alcuni concetti tecnici, questo capitolo sarà essenziale per acquisire una comprensione chiara delle logiche di funzionamento dell'Intelligenza Artificiale: una consapevolezza fondamentale per saperne cogliere appieno le opportunità e riconoscerne, al tempo stesso, i limiti strutturali.

2.1 IA simbolica: i sistemi esperti

Dopo che venne coniato il termine Intelligenza Artificiale nel 1956, durante il celebre workshop di Dartmouth, lo sviluppo di soluzioni *software* capaci di creare "macchine intelligenti" come abbiamo detto ha attraversato fasi alterne: momenti di entusiasmo e periodi di disillusione.

Negli anni Cinquanta, i computer avevano capacità computazionali estremamente limitate. La loro "intelligenza" si riduceva all'imitazione della capacità umana di manipolare simboli, ad esempio, operatori aritmetici, per compiere

semplici calcoli, come somme o moltiplicazioni. Solo successivamente vennero introdotte funzioni più complesse, come il confronto tra valori numerici. Ma per funzionare, questi sistemi richiedevano ancora una forte supervisione umana: erano gli esseri umani a creare gli algoritmi, a fornire i dati nel formato corretto e a interpretare i risultati ottenuti.

Tra gli anni Sessanta e Settanta presero forma i cosiddetti sistemi esperti. Questi programmi erano progettati per emulare il processo decisionale di esperti umani all'interno di domini (ambiti di applicazione) specifici, utilizzando basi di conoscenza dettagliate e motori inferenziali per analizzare dati e proporre soluzioni.

I sistemi esperti rappresentarono il primo tentativo concreto di superare i limiti dei *software* costruiti su algoritmi rigidi, cercando di rendere le macchine più flessibili e capaci di "ragionare" su informazioni complesse. È importante ricordare che, all'epoca, la gestione di grandi quantità di dati era ancora molto costosa: basti pensare che, nel 1970, alcuni scienziati dovettero dimostrare la possibilità di costruire modelli di linguaggio basati su appena venti parole, a causa dei limiti di memoria dei computer disponibili (Quillian, 1970).

Ma cosa rende "esperti" questi sistemi?

Il termine si riferisce alla loro capacità di eseguire procedure di inferenza, processi logici induttivi o deduttivi, per giungere a conclusioni partendo da un insieme di dati iniziali, anche se incompleti.

Il "cuore logico" di un sistema di intelligenza artificiale tradizionale, soprattutto nei sistemi esperti è il motore inferenziale. Si tratta del componente che applica regole (già programmate) per dedurre nuove informazioni o prendere decisioni.

In parole semplici, funziona come un investigatore che, avendo una lista di regole e alcune informazioni di partenza, cerca di arrivare a una conclusione logica.

Come funziona?

- Prende in *input* fatti (es. "il paziente ha la febbre")
- Applica regole del tipo "Se A e B, allora C" (es. "Se febbre e tosse, allora possibile influenza")
- Genera un *output*: una conclusione o azione (es. "Suggerire visita medica")

Spesso, per lavorare su dati qualitativi, veniva utilizzata la logica *fuzzy*, che consentiva di gestire informazioni incerte o imprecise, una capacità indispensabile in molti contesti reali.

Approfondimento: cosa significa logica "fuzzy"?

La logica fuzzy (o "logica sfumata") è un modo per ragionare senza usare solo il bianco e nero, cioè vero o falso, ma accettando anche tutte le sfumature intermedie.

Invece di dire che qualcosa è assolutamente vero (1) o assolutamente falso (0), la logica fuzzy permette di dire che un'affermazione è parzialmente vera, con un numero compreso tra "0" e "1".

Un esempio semplice:

Immagina di voler stabilire se una persona è alta.

- *Con la logica tradizionale:*
  - *Se è alta più di 1,80 m → Vero (1)*
  - *Se è più bassa → Falso (0)*
- *Con la logica fuzzy:*
  - *Una persona alta 1,75 m potrebbe avere un "grado di altezza" di 0,8*
  - *Una persona alta 1,60 m potrebbe avere un valore di 0,3*

In pratica: la logica fuzzy è utile quando le categorie sono vaghe o sfumate, come "caldo", "veloce", "giovane", "economico"…

È usata, ad esempio, nei termostati intelligenti, nelle auto a guida assistita e nei sistemi di raccomandazione, dove serve gestire situazioni incerte o non precise.

Il funzionamento di un sistema esperto si articola generalmente in diverse fasi. Si parte dall'acquisizione della conoscenza: gli esperti di un determinato ambito trasferiscono il proprio sapere in una base di conoscenza strutturata, che rappresenta la memoria operativa del sistema. In seguito, il motore inferenziale analizza i dati e le situazioni presentate, applicando regole logiche per giungere a conclusioni o soluzioni. Gli utilizzatori interagiscono con il sistema attraverso un'interfaccia utente che consente loro di inserire dati, porre domande e ricevere risposte o consigli. Infine, un elemento distintivo dei sistemi esperti è la capacità di giustificare le proprie decisioni: il sistema fornisce spiegazioni sulle conclusioni raggiunte, contribuendo così a instaurare un rapporto di fiducia con gli utenti.

Un esempio rimasto celebre di sistema esperto è il progetto DENDRAL, sviluppato nel 1965 da Edward Feigenbaum e Joshua Lederberg presso l'Università di Stanford (Feigenbaum & Lederberg, 1965).

DENDRAL fu progettato per aiutare i chimici a capire la struttura delle molecole. In pratica, funzionava così: i ricercatori inserivano i dati di laboratorio (ottenuti da strumenti come lo spettrometro), e DENDRAL cercava di

indovinare com'era fatta la molecola. Il sistema riusciva a elaborare e proporre soluzioni con un livello di accuratezza comparabile a quello degli esperti umani.

Il successo di DENDRAL aprì la strada a numerose applicazioni dei sistemi esperti in vari settori: dalla medicina, per la diagnosi e il trattamento di malattie, alla finanza, fino alla meteorologia.

Tuttavia, a partire dalla fine degli anni Ottanta, l'interesse verso i sistemi esperti cominciò a diminuire.

I limiti della logica simbolica emersero con chiarezza: man mano che i problemi da affrontare diventavano più complessi, i sistemi risultavano sempre più intricati, difficili da aggiornare e mantenere, perdendo così il vantaggio della semplicità e dell'economicità iniziale.

Parallelamente, la disponibilità di enormi quantità di dati liberamente accessibili rese più conveniente ricavare la conoscenza dall'analisi dei dati stessi, piuttosto che intervistare esperti per ogni singola materia.

I ricercatori cominciarono a dedicarsi sempre più ad un altro alternativo filone di studi per cercare di costruire macchine in grado di emulare la intelligenza umana, quello del cosiddetto "*Machine Learning*".

2.2 *Machine Learning*: l'apprendimento dai dati

La prima comparsa del termine *Machine Learning* (ML) risale al 1959 quando Arthur Samuel ha introdotto questa definizione nell'ambito dei suoi lavori sui programmi di gioco automatico (Samuel, 1959).

L'idea alla base del ML, o apprendimento automatico, è sorprendentemente semplice, ma rivoluzionaria: è possibile rappresentare la realtà attraverso funzioni matematiche che l'algoritmo non conosce a priori, ma che può individuare autonomamente osservando i dati. In altre parole, anziché definire esplicitamente ogni passaggio necessario per risolvere un problema, si lascia che il sistema impari da solo a costruire modelli e regole a partire dall'esperienza.

Questo approccio si differenzia significativamente dal concetto tradizionale di algoritmo. I *software* classici (come i sistemi esperti) risolvono problemi seguendo una sequenza precisa e predeterminata di istruzioni (una serie di *what / if)*. Al contrario, il *Machine Learning* affronta problemi che non possono essere codificati facilmente in una serie rigida di passaggi, ma che l'essere umano riesce a gestire in modo intuitivo sulla base dell'esperienza.

Un esempio pratico può rendere chiara questa differenza: immaginiamo di voler costruire un sistema che preveda se domani pioverà.

Seguendo un approccio tradizionale, dovremmo fornire al computer una lunga lista di regole esplicite, come:

- se la pressione atmosferica scende sotto un certo valore, allora probabile pioggia;
- se l'umidità supera una certa soglia e la temperatura è bassa, allora possibilità di pioggia.

Ma il tempo atmosferico è influenzato da un insieme complesso di variabili: pressione, temperatura, vento, umidità, stagionalità, condizioni locali… Ogni nuova combinazione di questi fattori potrebbe richiedere di scrivere regole aggiuntive.

E non è tutto: ogni nuova regola non si scrive da sola. Deve essere individuata, compresa e formalizzata da un essere umano. Serve un esperto che osservi il fenomeno, riconosca un *pattern*, lo traduca in linguaggio logico e lo integri nel sistema. Questo processo richiede tempo, competenze e risorse.

Alla fine, mantenere e aggiornare manualmente tutte queste istruzioni diventerebbe eccessivamente complicato se non impossibile: troppe eccezioni, troppi casi particolari, troppe interazioni imprevedibili. Un sistema basato solo su regole statiche finirebbe per essere fragile, rigido e poco scalabile di fronte alla complessità dinamica del mondo reale.

Con il *Machine Learning*, invece, il metodo cambia radicalmente: non si scrivono regole. Si raccolgono grandi quantità di dati storici, condizioni atmosferiche passate e corrispondenti osservazioni di pioggia o bel tempo, e si lascia che il sistema impari da solo a individuare le correlazioni.

Ad esempio, il modello potrebbe scoprire, senza che nessuno glielo abbia detto, che un rapido calo della pressione combinato a un certo tasso di umidità aumenta la probabilità di pioggia, oppure che un certo tipo di vento in una specifica area geografica anticipa l'arrivo di precipitazioni.

Il sistema non si limita a "ricordare" i dati passati, ma costruisce una propria rappresentazione della realtà, imparando a prevedere il futuro anche su combinazioni di dati mai viste prima.

Così, quando riceve i dati aggiornati di oggi, pressione, umidità, temperatura, può prevedere se domani pioverà, basandosi sull'esperienza accumulata dall'analisi di migliaia di situazioni precedenti.

Un aspetto fondamentale che distingue il *Machine Learning* è la sua capacità di analizzare contemporaneamente un numero enorme di dati, anche apparentemente non correlati all'oggetto dell'analisi, e di scoprire pattern nascosti tra di essi. Nel caso delle previsioni del tempo, un sistema potrebbe rilevare che particolari variazioni minime nella velocità del vento, combinate a impercettibili cambiamenti nella pressione atmosferica, sono predittive di fenomeni meteorologici specifici. Questi schemi, troppo complessi o controintuitivi per essere

individuati manualmente dall'uomo, vengono invece colti dall'algoritmo, che costruisce modelli predittivi più efficaci e adattabili.

Questa stessa logica si applica in molti altri ambiti, come ad esempio quello delle scommesse sportive. Immaginiamo di voler prevedere chi vincerà una partita di tennis. Un approccio semplice potrebbe basarsi su pochi elementi: la classifica dei due giocatori, il numero di partite vinte negli ultimi giorni e forse il risultato del loro ultimo incontro. Ma in realtà, il risultato di una partita può dipendere da molte più cose. Per esempio, il tipo di campo – erba, terra o cemento – può favorire uno stile di gioco rispetto a un altro. Anche le condizioni del tempo possono influenzare la prestazione, così come eventuali infortuni, la stanchezza accumulata da viaggi recenti, il fuso orario o lo stato d'animo del giocatore. Perfino la presenza del pubblico o del proprio allenatore può fare la differenza.

Un sistema di intelligenza artificiale può prendere in considerazione tutte queste variabili contemporaneamente e individuare schemi che non sono immediatamente visibili. In questo modo, riesce a formulare una previsione più precisa e affidabile rispetto a chi si basa solo su pochi dati superficiali.

Come fa, dunque, il *Machine Learning* a "imparare da solo"? Quale meccanismo gli consente di riconoscere schemi così complessi e trarre conclusioni affidabili da enormi quantità di dati?

Per scoprirlo, dobbiamo entrare nel cuore pulsante di molte applicazioni di IA: le reti neurali artificiali, sistemi ispirati al funzionamento del cervello umano.

Approfondimento: come funziona il cervello umano?

Il cervello umano è formato da circa 86 miliardi di neuroni, cellule specializzate che comunicano tra loro attraverso segnali elettrici e chimici. Ogni neurone riceve informazioni da altri neuroni tramite connessioni chiamate sinapsi, le elabora e decide se trasmettere un segnale a sua volta.

Quando impariamo qualcosa – come andare in bicicletta o riconoscere un volto – il cervello rafforza certe connessioni e ne indebolisce altre, rendendo più facile e veloce rispondere agli stimoli simili in futuro. Questo processo si chiama plasticità neurale e rappresenta la base dell'apprendimento biologico.

Le reti neurali artificiali cercano di imitare questo meccanismo: anche loro sono fatte da "neuroni" collegati tra loro, che modificano le proprie "connessioni" (i pesi) per migliorare le prestazioni nel tempo.

Le reti neurali sono composte da unità di calcolo, chiamate neuroni artificiali, che elaborano dati e trasmettono segnali attraverso connessioni pesate. L'apprendimento avviene modificando questi pesi, in base all'errore commesso nel confronto tra il risultato previsto e quello effettivo.

Le prime reti neurali, sviluppate a partire dagli anni Cinquanta e Sessanta, erano relativamente semplici e limitate nella loro capacità di apprendere funzioni complesse. Affrontavano principalmente problemi lineari, e non riuscivano a catturare tutta la ricchezza e la variabilità del mondo reale.

Queste difficoltà hanno aperto la strada allo sviluppo di un metodo più sofisticato e capace di affrontare problemi complessi: il *Deep Learning*.

Approfondimento: cosa sono le relazioni lineari e non lineari?

Una relazione lineare è un tipo di legame semplice tra due variabili: se una cresce, anche l'altra cresce (o diminuisce) in modo proporzionale. È come dire: "più studi, più alto sarà il voto", seguendo una linea retta. Se rappresentiamo questa relazione su un grafico, otteniamo infatti una retta.

Le relazioni non lineari, invece, sono più complesse: l'effetto di una variabile sull'altra può cambiare in base alla situazione. Per esempio, "un po' di sonno in più migliora la concentrazione, ma dormire troppo può renderci più stanchi." In un grafico, queste relazioni non formano una retta, ma curve, onde, salti… insomma, comportamenti molto più variabili.

Le reti neurali più semplici erano in grado di gestire solo relazioni lineari o poco più. Ma il mondo reale – dal linguaggio umano al riconoscimento di immagini – è fatto di dinamiche non lineari, molto più difficili da descrivere.

2.3 *Machine Learning* tradizionale: la cassetta degli attrezzi

Abbiamo visto che il *Machine Learning* si distingue dalla programmazione classica perché il sistema impara dai dati anziché seguire regole scritte a mano. Ma prima che il *Deep Learning* diventasse dominante, i ricercatori hanno sviluppato una serie di tecniche di apprendimento automatico che, pur non basandosi su reti neurali, si sono rivelate estremamente efficaci – e che sono ancora oggi ampiamente utilizzate.

Chiamerò queste tecniche "*Machine Learning* tradizionale", per distinguerle dal *Deep Learning*. Conoscerle è importante per almeno due ragioni: primo, perché rappresentano le fondamenta su cui si è costruita l'IA moderna; secondo, perché per molti problemi reali – soprattutto quando i dati sono strutturati in

tabelle con righe e colonne – restano la soluzione migliore, più veloce e più interpretabile.

Immaginiamo di lavorare al pronto soccorso e di dover decidere rapidamente la priorità di ogni paziente che arriva. Un medico esperto, dopo anni di esperienza, ha sviluppato un intuito: guarda la pressione arteriosa, la frequenza cardiaca, la temperatura, e in pochi secondi capisce se il caso è urgente o può attendere.

Un albero decisionale fa qualcosa di simile, ma partendo dai dati. Prende i dati storici di centinaia di pazienti – le loro caratteristiche e l'esito – e costruisce una serie di domande in sequenza: "La pressione è sotto 91?" Se sì, bassa urgenza. Se no, "L'età è superiore ai 60 anni?" E così via, ramificandosi come un albero. Ad ogni domanda, l'algoritmo sceglie la caratteristica che meglio divide i pazienti in gruppi omogenei, riducendo progressivamente l'errore di classificazione.

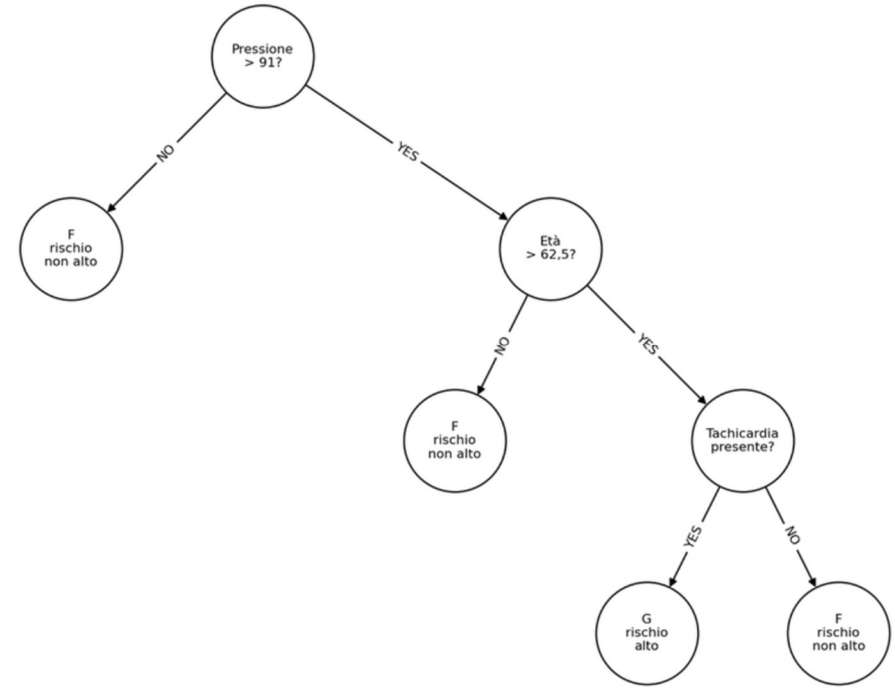

Figura 6: Albero decisionale

Ma come fa l'algoritmo a scegliere quale domanda fare per prima? Immaginiamo di avere i dati di 100 pazienti arrivati al pronto soccorso, di cui 30 sono stati classificati come urgenti e 70 come non urgenti. L'algoritmo deve trovare la caratteristica che divide meglio questi due gruppi.

Supponiamo di avere tre informazioni per ogni paziente: la pressione sistolica, la frequenza cardiaca e la temperatura corporea. L'algoritmo prova tutte le

possibili soglie per ciascuna caratteristica. Ad esempio, prova a dividere i pazienti in base a "pressione sotto 91" e verifica: dei 20 pazienti con pressione sotto 90, quanti sono urgenti? Se sono 18 su 20 (il 90%), la divisione è molto buona – ha creato un gruppo quasi puro. Poi prova "frequenza cardiaca sopra 110": dei 35 pazienti che soddisfano questa condizione, 15 sono urgenti (il 43%) – una divisione meno netta.

Il criterio matematico usato si chiama impurità: un nodo è "puro" quando contiene pazienti tutti dello stesso tipo, ed è "impuro" quando li mescola. L'algoritmo sceglie, ad ogni passo, la domanda che riduce di più l'impurità – in pratica, quella che crea i sottogruppi più omogenei. Nel nostro esempio, "pressione sotto 91" vince e diventa la prima domanda dell'albero, la radice. Poi il processo si ripete per ciascun ramo: tra i pazienti con pressione sopra 91, quale domanda li separa meglio? Forse "frequenza cardiaca sopra 110". E così via, fino a quando ogni foglia dell'albero contiene un gruppo sufficientemente omogeneo, oppure si raggiunge una profondità massima decisa in anticipo per evitare che l'albero diventi troppo complesso e memorizzi i dati anziché imparare da essi – un fenomeno noto come *overfitting*.

Il grande vantaggio degli alberi decisionali è la loro trasparenza: a differenza di una rete neurale profonda, dove i meccanismi decisionali sono nascosti in milioni di pesi, un albero decisionale ti mostra esattamente perché ha preso una certa decisione. È possibile seguire il percorso dall'alto verso il basso e capire quale combinazione di fattori ha determinato l'*output*. In ambiti come la medicina o la finanza, dove le decisioni devono essere spiegabili, questa proprietà è fondamentale.

Gli alberi decisionali non sono l'unica tecnica del ML tradizionale. Eccone alcune altre:

La regressione lineare e logistica permette di prevedere un valore numerico (ad esempio la pressione arteriosa di domani) o di classificare un dato in due categorie (ad esempio, rischio alto o basso). È una delle tecniche più semplici e interpretabili. Per esempio, un modello di regressione lineare potrebbe imparare che per ogni anno di età in più, la pressione sistolica media aumenta di 0,6 mmHg: una relazione chiara, espressa da una formula che chiunque può leggere e verificare. La regressione logistica, invece, potrebbe prendere età, colesterolo e abitudine al fumo di un paziente e calcolare che la sua probabilità di evento cardiovascolare nei prossimi dieci anni è del 14%.

Le *Support Vector Machines* (SVM) cercano di trovare la linea (o il piano, in più dimensioni) che meglio separa due gruppi di dati. Sono particolarmente efficaci quando i dati non sono facilmente separabili con metodi lineari, grazie a un trucco matematico chiamato "kernel" che proietta i dati in uno spazio con

più dimensioni. Immagina di avere su un tavolo delle biglie rosse e blu mescolate in cerchi concentrici: guardando il tavolo dall'alto, nessuna linea retta può separarle. Ma se sollevi le biglie in base alla loro distanza dal centro – proiettandole in uno spazio tridimensionale – improvvisamente un piano orizzontale le divide perfettamente. Questo è, in sostanza, ciò che fa il kernel. Le SVM sono state tra le tecniche più usate per il riconoscimento della scrittura a mano, come quello dei codici di avviamento postale sulle buste.

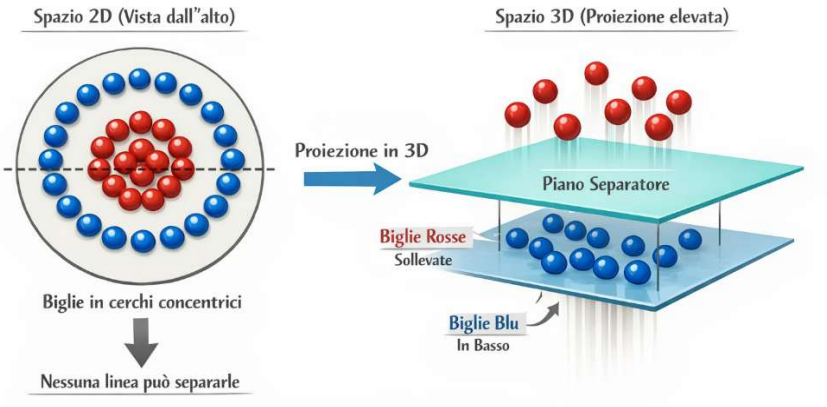

Figura 7: Support Vector Machines

Il *K-Nearest Neighbors* (K-NN) classifica un nuovo dato guardando i K dati più simili nel dataset di addestramento: se la maggior parte dei "vicini" appartiene alla categoria A, il nuovo dato sarà classificato come A. È intuitivo, come chiedere consiglio ai tuoi vicini di banco. Se un servizio di streaming deve decidere se consigliarti un film, può cercare i cinque utenti con gusti più simili ai tuoi e verificare: se quattro di loro hanno apprezzato quel film, probabilmente piacerà anche a te. La "distanza" tra utenti si calcola confrontando i voti dati agli stessi film – più sono simili, più siete "vicini" nello spazio dei dati.

Il *Naïve Bayes* usa la probabilità per classificare: dato un insieme di caratteristiche, calcola la probabilità che il dato appartenga a ciascuna categoria. È il cuore dei primi filtri anti-spam delle email. Funziona così: il modello apprende che parole come "gratis", "offerta imperdibile" e "clicca qui" compaiono molto più spesso nelle email di spam che in quelle legittime. Quando arriva un nuovo messaggio, calcola la probabilità che sia spam in base alle parole che contiene. Se il messaggio include "hai vinto" e "clicca subito", la probabilità combinata di spam schizza verso il 99% – e il messaggio finisce nella cartella indesiderata. Si chiama "naïve" (ingenuo) perché assume che ogni parola contribuisca in modo indipendente, un'approssimazione che nella realtà non è sempre vera, ma che funziona sorprendentemente bene.

Infine, i metodi *Ensemble* combinano più modelli per ottenere previsioni migliori. Le *Random Forest* costruiscono decine o centinaia di alberi decisionali e li fanno "votare"; il *Gradient Boosting* costruisce alberi in sequenza, dove ogni nuovo albero si concentra sugli errori del precedente. L'idea è simile a quella di un consiglio medico: un singolo medico può sbagliare diagnosi, ma se consulti cento specialisti e segui l'opinione della maggioranza, la probabilità di errore diminuisce drasticamente. Nella pratica, gli algoritmi di *Gradient Boosting* come XGBoost e LightGBM dominano le competizioni di data science su dati tabulari – quelli organizzati in righe e colonne, come fogli di calcolo o database – superando spesso anche le reti neurali profonde in questo tipo di problemi.

Potrebbe venire naturale chiedersi: se il *Deep Learning* è così potente, perché usare ancora queste tecniche "tradizionali"? La risposta è che ogni strumento ha il suo ambito ideale. Le reti neurali profonde eccellono con dati non strutturati – immagini, testi, *audio* – dove le relazioni tra i dati sono complesse e difficili da codificare esplicitamente. Ma per i dati strutturati – le classiche tabelle con righe e colonne che si trovano nelle aziende, negli ospedali, nelle banche – il ML tradizionale resta spesso la scelta migliore: più veloce da addestrare, più facile da interpretare, meno avido di dati e di potenza di calcolo.

Nella pratica, i due approcci convivono e si completano. Un sistema di IA moderno potrebbe usare una rete neurale per analizzare le immagini radiografiche e un albero decisionale per combinare i risultati con i dati clinici del paziente. Capire entrambi gli approcci è fondamentale per chi vuole davvero comprendere l'Intelligenza Artificiale.

2.4 *Deep Learning*: il salto di qualità

Il *Deep Learning* (DL) è una branca avanzata dell'apprendimento automatico (*Machine Learning*) che si distingue per l'uso di reti neurali profonde, ovvero modelli composti da numerosi strati di neuroni artificiali. Per capire appieno il *Deep Learning*, è utile chiarire prima la distinzione tra *Machine Learning* e *Deep Learning*.

Il *Machine Learning* comprende tutte quelle tecniche in cui un sistema impara a compiere previsioni o decisioni a partire dai dati, migliorando la propria *performance* nel tempo senza essere stato esplicitamente programmato per ogni singolo compito. Nelle sue forme tradizionali, il *Machine Learning* richiede che siano gli esseri umani a selezionare le caratteristiche più rilevanti dei dati, in un processo chiamato *feature engineering*. Per esempio, se volessimo addestrare un algoritmo a riconoscere le automobili in una foto, dovremmo indicare al sistema quali elementi osservare: il numero di ruote, la forma dei fari, la presenza di una carrozzeria, ecc.

Il *Deep Learning* si differenzia proprio in questo punto: le reti profonde imparano da sole a estrarre e combinare le caratteristiche più significative dei dati, senza che sia necessario intervenire manualmente. Questo approccio consente di affrontare problemi molto più complessi, come il riconoscimento di immagini, la traduzione automatica di testi, o la generazione di contenuti realistici.

Oggi, grazie alla disponibilità di grandi quantità di dati, alla potenza dei computer e ai progressi negli algoritmi di IA, il *Deep Learning* è diventato la tecnologia alla base di molte delle innovazioni che stiamo vivendo, dai sistemi di raccomandazione ai veicoli autonomi, fino ai modelli di intelligenza artificiale generativa.

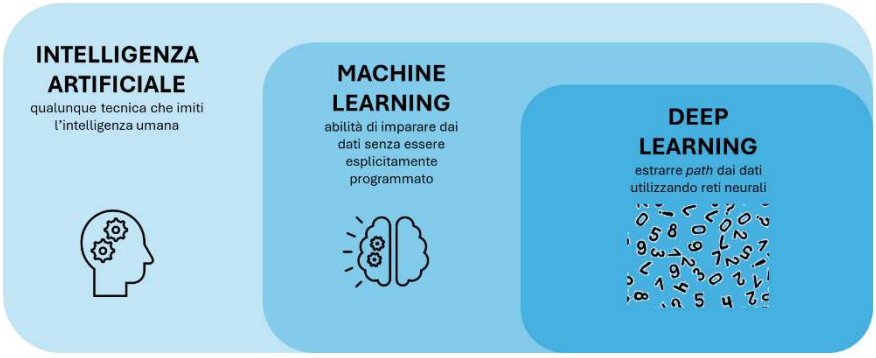

Figura 8: Machine Learning e Deep Learning nell'ambito della IA

Finora abbiamo introdotto alcuni concetti che, a prima vista, possono sembrare un po' astratti o confusi – ed è del tutto normale. Per comprendere davvero come funziona il *Deep Learning*, è fondamentale partire dalle reti neurali, che ne sono il fondamento. Nei prossimi paragrafi le esploreremo passo dopo passo, e vedrai che, man mano che ne scopriremo il funzionamento, tutto comincerà a diventare molto più chiaro.

2.5 Reti neurali artificiali: architetture di base

Le "reti neurali artificiali" (*Artificial Neural Network* – ANN) rappresentano uno dei tentativi più affascinanti della scienza di imitare la complessità del cervello umano. L'idea di fondo è semplice: così come il nostro cervello è formato da miliardi di neuroni collegati tra loro, una rete neurale artificiale è composta da unità elementari – i neuroni artificiali – che ricevono segnali, li elaborano e li trasmettono ad altri neuroni. È dalla cooperazione di migliaia o milioni di queste unità che emergono le capacità sorprendenti dell'IA moderna.

Tutto parte dal Perceptron, inventato nel 1958 da Frank Rosenblatt (Rosenblatt, 1958). Si tratta del prototipo del neurone artificiale: un'unità che riceve

dati in ingresso, assegna a ciascuno un "peso" – un numero che ne indica l'importanza relativa – e produce un risultato in uscita. L'idea di un neurone artificiale, tuttavia, risale ancora più indietro, agli anni Quaranta, quando Warren McCulloch e Walter Pitts (McCulloch & Pitts, 1943) proposero per primi una rappresentazione matematica del comportamento neuronale. Fu però Rosenblatt a trasformare quell'intuizione in qualcosa di concreto e operativo: il suo Mark I Perceptron era una macchina fisica, composta da 400 fotocellule e 512 unità di output, capace di apprendere a riconoscere lettere e immagini. La notizia fece scalpore: il New York Times titolò "Machine Can Learn by Itself", e Rosenblatt predisse che la sua macchina avrebbe presto riconosciuto volti, tradotto lingue e composto musica.

Il funzionamento di un neurone artificiale si può riassumere in tre passaggi: il neurone moltiplica ciascun dato di ingresso per il suo peso, somma tutti i risultati (aggiungendo un valore costante chiamato "bias", che serve a rendere il modello più flessibile), e infine applica una "funzione di attivazione" – una trasformazione matematica che introduce la capacità di gestire relazioni complesse e non lineari tra i dati. Senza questa funzione, anche una rete con migliaia di neuroni si comporterebbe come un modello banale, capace solo di tracciare linee rette. Con essa, la rete può "curvare" lo spazio dei dati e imparare a separare categorie che non seguono confini rettilinei – come distinguere mele da banane quando sono mescolate in modo intricato. Per un approfondimento sulle principali funzioni di attivazione e sul loro ruolo matematico, si rimanda all'Appendice sulle funzioni di attivazione.

Un singolo neurone può fare poco da solo. La vera potenza emerge quando molti neuroni vengono combinati in strati successivi. Una rete neurale tipica ha tre componenti: uno strato di ingresso (che riceve i dati), uno o più strati nascosti (dove avviene l'elaborazione), e uno strato di uscita (che produce il risultato). Quando gli strati nascosti sono molti – decine, centinaia o anche migliaia – parliamo di "reti neurali profonde" (*Deep Neural Network* – DNN), ed è da qui che viene il termine Deep Learning.

La profondità è ciò che permette a queste reti di affrontare problemi complessi. Ogni strato impara a riconoscere caratteristiche sempre più astratte: in una rete che analizza immagini, ad esempio, i primi strati potrebbero rilevare bordi e colori, quelli intermedi forme geometriche, e quelli finali oggetti completi come volti o automobili. È un po' come salire una scala di astrazione: dal dettaglio al concetto.

Per capire concretamente come funziona, immaginiamo di voler prevedere se uno studente supererà un esame, conoscendo le sue ore di studio e il voto

all'esame intermedio. Una rete con due ingressi, tre neuroni nello strato nascosto e un'uscita può imparare questa relazione. Ma come fa a "imparare"?

All'inizio, i pesi della rete sono assegnati a caso – il che significa che le sue previsioni saranno sostanzialmente inutili. L'apprendimento avviene attraverso un processo iterativo: la rete riceve dei dati di cui conosce già il risultato corretto (i dati di "addestramento"), produce una previsione, la confronta con la realtà e misura l'errore commesso. Poi, e qui sta il cuore del meccanismo, corregge i propri pesi per ridurre quell'errore. Il comportamento di una rete neurale è paragonabile a quello di un bambino: prima dell'istruzione non conosce le regole del mondo, e solo attraverso l'esperienza e il feedback può migliorare le sue capacità.

Il metodo con cui la rete corregge i pesi si chiama Backpropagation (retropropagazione dell'errore), sviluppato nella sua forma moderna negli anni Ottanta da David Rumelhart, Geoffrey Hinton e Ronald Williams (Rumelhart, Hinton & Williams, 1986). L'idea è elegante: l'errore calcolato sull'uscita viene "propagato all'indietro" attraverso la rete, strato per strato, fino all'ingresso. Ogni neurone riceve un'indicazione su quanto ha contribuito all'errore complessivo, e il suo peso viene aggiornato di conseguenza. Per decidere in quale direzione correggere i pesi, si utilizza la "discesa del gradiente" – un algoritmo che, ad ogni passo, cerca la direzione in cui l'errore diminuisce più rapidamente, un po' come uno sciatore che cerca la via più ripida per scendere a valle.

Ripetendo questo processo per migliaia o milioni di cicli (chiamati "epoche"), la rete affina progressivamente i propri pesi fino a raggiungere previsioni accurate. Nel nostro esempio dello studente, dopo l'addestramento la rete potrebbe assegnare pesi elevati sia alle ore di studio sia al voto intermedio, "capendo" che entrambi sono buoni predittori del successo all'esame. Ogni neurone dello strato nascosto si specializza nel catturare una diversa sfumatura dei dati – uno potrebbe essere sensibile alla combinazione di studio e voto, un altro più al voto da solo, un altro ancora allo sforzo di studio – e la loro cooperazione consente alla rete di fare previsioni sfumate e accurate. Per l'esempio numerico completo, con tutti i calcoli passo per passo e l'interpretazione dei pesi, si rimanda alle Appendici sulle reti neurali e la backpropagation.

Le reti che abbiamo descritto fin qui – ANN e DNN – sono potenti ma hanno un limite: trattano ogni dato di ingresso come se fosse indipendente da tutti gli altri. Non hanno memoria e non colgono relazioni di ordine o di vicinanza tra i dati. In una frase come "il presidente, dopo aver partecipato a un vertice con i ministri, ha ribadito la sua posizione", queste reti non saprebbero collegare il verbo finale al soggetto iniziale. Analogamente, analizzando un'immagine pixel per pixel, non coglierebbero che certi gruppi di pixel formano un occhio, un naso, un volto.

Per superare questi limiti, sono state sviluppate due architetture specializzate.

Le Reti Neurali Ricorrenti (RNN – *Recurrent Neural Network*), le cui basi teoriche sono state formalizzate da Rumelhart, McClelland e il PDP Research Group (Rumelhart, McClelland & PDP Research Group, 1986), sono progettate per elaborare sequenze – frasi, audio, serie temporali. La loro innovazione chiave è una forma di "memoria": ogni neurone, oltre a ricevere i dati correnti, riceve anche l'output del passo precedente, creando un ciclo che permette alla rete di tenere traccia del contesto. È grazie alle RNN che sono nati i primi sistemi di traduzione automatica (Google Translate dal 2006), gli assistenti vocali (Siri nel 2011, Alexa nel 2014, Google Assistant nel 2016) e i primi modelli di analisi del sentimento sui social media. Tuttavia, le RNN hanno un tallone d'Achille: tendono a "dimenticare" le informazioni più lontane nel tempo (un problema noto come vanishing gradient), e l'elaborazione strettamente sequenziale le rende lente da addestrare. Varianti come le LSTM (*Long Short-Term Memory*) hanno migliorato la situazione introducendo meccanismi di memoria più sofisticati, ma la vera svolta sarebbe arrivata solo nel 2017 con un'architettura radicalmente diversa: il Transformer, di cui parleremo nella prossima sezione.

Per una descrizione di dettaglio sul funzionamento interno delle RNN, si rimanda all'Appendice sulle reti neurali ricorrenti.

Le Reti Neurali Convoluzionali (CNN – *Convolutional Neural Network*) sono invece specializzate nell'analisi di immagini e video. L'ispirazione viene dalla corteccia visiva del cervello umano, dove diversi neuroni si attivano per porzioni specifiche del campo visivo. Il principio chiave è la "convoluzione": un piccolo filtro scorre sull'immagine ed estrae caratteristiche locali – bordi, colori, texture – e passando attraverso strati successivi la rete impara a riconoscere pattern sempre più complessi, dai dettagli elementari fino a oggetti completi. Le prime CNN risalgono al modello Neocognitron di Kunihiko Fukushima (Fukushima, 1980), ispirato al funzionamento del cervello visivo, ma fu Yann LeCun (LeCun et al., 1998) a sviluppare una delle prime architetture realmente operative: LeNet-5, usata con successo per il riconoscimento automatico delle cifre sugli assegni bancari e poi adottata da banche e sistemi postali. Oggi le CNN sono ovunque: nel riconoscimento facciale dei nostri smartphone, nella guida autonoma, nella diagnostica medica per immagini. Anche le CNN, tuttavia, presentano dei limiti: i filtri analizzano solo porzioni ristrette dell'immagine, e cogliere relazioni tra elementi visivi distanti richiede reti molto profonde e costose da addestrare.

Per una descrizione di dettaglio sul funzionamento interno delle CNN, si rimanda all'Appendice sulle reti neurali convoluzionali.

I limiti delle reti ricorrenti e convoluzionali – la difficoltà nel catturare dipendenze a lungo raggio, la lentezza dell'elaborazione sequenziale, la rigidità nell'analisi locale – hanno aperto la strada a una nuova fase del deep learning. Come vedremo nella prossima sezione, tre innovazioni hanno cambiato radicalmente il panorama: gli autoencoder, le Generative Adversarial Network e soprattutto i Transformer, l'architettura che ha reso possibile l'IA generativa che conosciamo oggi.

2.6 Reti neurali artificiali: architetture avanzate

Le limitazioni strutturali delle reti neurali ricorrenti (RNN) e di quelle convoluzionali (CNN) hanno stimolato una nuova fase evolutiva nel campo del *deep learning*. È in questa fase evolutiva del *deep learning* che sono emersi tre modelli fondamentali che hanno contribuito in modo decisivo alla nascita dell'intelligenza artificiale generativa: gli *autoencoder*, le *Generative Adversarial Networks* (GAN) e i modelli *Transformer*.

Gli ***autoencoder***, sono stati tra i primi modelli in grado di apprendere una rappresentazione compressa (latente) di un dato e di ricostruirlo, generando variazioni plausibili a partire da tale rappresentazione. Questo approccio ha fornito le basi per la generazione controllata di contenuti, come immagini, suoni o sequenze, ponendo le fondamenta per i futuri modelli generativi. Successivamente, le **GAN** hanno rivoluzionato il campo: introducendo un meccanismo competitivo tra due reti (generatore e discriminatore), hanno reso possibile la generazione di dati completamente nuovi e realistici, senza bisogno di etichettature. Per la prima volta, l'IA non si limitava a classificare o riconoscere, ma iniziava a "immaginare", inaugurando il paradigma dell'IA generativa.

Ma la vera rivoluzione è arrivata nel 2017 con l'introduzione dei ***Transformer***. Questi modelli hanno abbandonato completamente l'idea di ricorrenza, che fino ad allora era stata il cuore di tutte le reti neurali progettate per il linguaggio. Al suo posto, i *Transformer* hanno introdotto un meccanismo del tutto nuovo: "l'attenzione".

Gli ***autoencoder*** sono una particolare architettura di rete neurale progettata per imparare a comprimere e poi ricostruire i dati. A differenza di altri modelli che si concentrano su classificazione o previsione, l'obiettivo di un *autoencoder* è quello di ridurre la complessità dell'informazione mantenendo il più possibile le sue caratteristiche essenziali. La rete è composta da due parti: un ***encoder***, che

prende i dati in ingresso e li trasforma in una versione "compressa" (detta anche rappresentazione latente), e un ***decoder***, che cerca di ricostruire i dati originali a partire da quella versione compressa.

Un esempio pratico può aiutare a capire meglio questo meccanismo: immaginiamo di fornire alla rete l'immagine di una racchetta da tennis. L'*encoder* analizza l'immagine e la converte in pochi numeri chiave, come 5,5 e 3,6, che riassumono le informazioni fondamentali: la forma dell'oggetto, la disposizione delle corde, il colore, le proporzioni. Il *decoder*, a sua volta, prende questi numeri e li trasforma nuovamente in un'immagine, cercando di ricostruire la racchetta originale il più fedelmente possibile.

Questo esempio numerico (due valori latenti) è chiaramente puramente illustrativo: nella realtà, un *autoencoder* lavora con decine, centinaia o anche migliaia di variabili. L'*encoder* non produce semplici numeri, ma una rappresentazione matematica complessa, codificata nei pesi di una rete neurale molto estesa, che viene addestrata su grandi quantità di dati per cogliere le strutture profonde del contenuto.

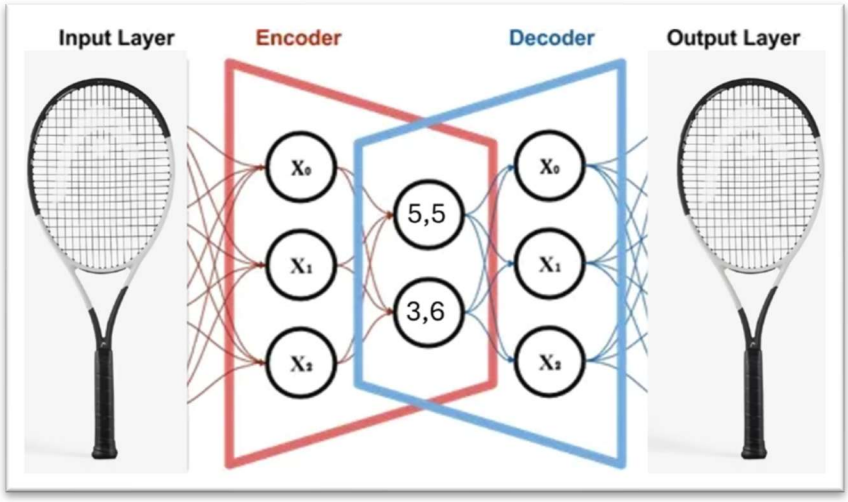

Figura 9: Struttura di encoder – decoder al lavoro

A questo punto la domanda è: cosa succede se, invece di fornire al *decoder* i valori latenti originali, gliene diamo altri, leggermente diversi o del tutto nuovi? Se il *decoder* è stato addestrato su centinaia o migliaia di immagini di racchette, sarà in grado di interpretare anche quelle nuove combinazioni e generare un'immagine di racchetta che non esiste: diversa da tutte quelle viste in fase di *training*, ma comunque coerente con ciò che ha appreso.

Ed è proprio qui che prende forma il concetto di intelligenza artificiale generativa: la possibilità, per una rete neurale, di creare nuovi contenuti plausibili, non semplicemente di riprodurre quelli già visti. Grazie alla rappresentazione compatta nello spazio latente (i nostri due numeri 5,5 e 3,6 dell'esempio semplificato), il modello può "esplorare" nuove combinazioni e produrre variazioni credibili: nuove racchette, nuovi volti, nuovi oggetti, tutti mai esistiti prima ma perfettamente realistici. Gli *autoencoder*, in particolare nella loro evoluzione come VAE (*Variational Autoencoder*), sono quindi stati tra i primi strumenti capaci di trasformare la compressione in creatività computazionale, aprendo la strada all'era dell'IA generativa.

Approfondimento: cos'è l'Intelligenza Artificiale Generativa?

L'intelligenza artificiale generativa (in inglese generative AI) è una branca dell'IA che si occupa di creare nuovi contenuti a partire da dati esistenti. A differenza delle tecniche tradizionali, che classificano, prevedono o riconoscono, i modelli generativi sono progettati per produrre immagini, testi, suoni, video o altri dati completamente nuovi, che risultino realistici e coerenti con quanto appreso durante l'addestramento.

Questa capacità si basa su reti neurali avanzate (come autoencoder, GAN e Transformer) che imparano a modellare la distribuzione dei dati, ovvero il "modo" in cui un certo tipo di informazione si presenta. Una volta appresa questa distribuzione, l'IA può generare nuove varianti mai viste prima, ma credibili.

Esempi concreti?

- *un modello può creare il volto di una persona che non esiste, ma che sembra perfettamente reale;*
- *può scrivere un testo nello stile di un autore specifico;*
- *può inventare una nuova ricetta, un logo, un brano musicale o persino una molecola.*

Questa capacità di "immaginare" nuovi contenuti rende l'IA generativa uno strumento potente per la creatività, la simulazione, la prototipazione e, naturalmente, l'innovazione.

Continuiamo con la carrellata delle nuove idee rivoluzionarie e passiamo alla seconda: le GAN. Le ***Generative Adversarial Networks*** sono state inventate nel 2014 da Ian Goodfellow (Goodfellow et al., 2014). Il loro funzionamento si

basa su un'idea tanto semplice quanto geniale: far competere due reti neurali in una sorta di "duello" digitale, così che ciascuna si migliori attraverso il confronto con l'altra. Queste due reti svolgono ruoli opposti ma complementari:

- la prima è il "generatore". Il suo compito è creare dati falsi, ma credibili, a partire da un *input* casuale – ad esempio, una sequenza di numeri
- la seconda è il "discriminatore". Il suo ruolo è giudicare: riceve sia dati reali (provenienti da un archivio autentico, come fotografie o brani musicali), sia quelli sintetici prodotti dal generatore, e deve stabilire quali sono veri e quali sono falsi. È come un esperto d'arte che cerca di distinguere le opere originali dalle imitazioni.

Questo meccanismo competitivo è estremamente efficace: il generatore impara progressivamente a produrre contenuti sempre più realistici, mentre il discriminatore affina la sua capacità di smascherare i falsi. Il risultato è un ciclo virtuoso di miglioramento reciproco, che porta entrambe le reti a livelli sempre più alti di sofisticazione.

Un aspetto particolarmente interessante delle GAN è che non necessitano di dati etichettati. Nessuno deve indicare se un'immagine rappresenta un cane o una mela: basta fornire un insieme di dati reali, e la rete impara da sola le caratteristiche che li rendono plausibili.

I risultati sono sorprendenti: oggi le GAN possono creare immagini di volti umani che non esistono, progettare abiti futuristici, generare paesaggi mozzafiato, comporre musica, simulare voci, e perfino produrre dati sintetici per la medicina, la moda, il design e la scienza. Esistono già siti web che mostrano ritratti fotorealistici di persone… che in realtà non sono mai nate.

Ma la corsa non si è fermata qui. Dopo le GAN, ha preso piede un'architettura ancora più potente e versatile, pensata inizialmente per il linguaggio, ma oggi impiegata anche nelle immagini, nel suono e nei dati scientifici: i modelli *Transformer*, che vedremo nel prossimo paragrafo.

Questa architettura ha segnato una svolta cruciale nell'elaborazione del linguaggio naturale (NLP) e, più in generale, nella gestione di dati sequenziali come testo, codice, musica e persino immagini.

Prima della loro introduzione, come abbiamo visto, i modelli dominanti erano le reti neurali ricorrenti (RNN) e le reti convoluzionali (CNN), che tuttavia presentavano limiti strutturali significativi: le RNN faticavano a mantenere memoria di informazioni lontane all'interno di un testo lungo, mentre le CNN non riuscivano a cogliere agevolmente le relazioni globali tra parole o concetti lontani tra loro. Inoltre, entrambi i modelli avevano difficoltà a sfruttare appieno la parallelizzazione, rendendo l'addestramento lento e poco efficiente.

Nel 2017, un gruppo di ricercatori di Google guidato da Ashish Vaswani ha proposto un modello completamente nuovo nel paper ormai celebre *"Attention is All You Need"* (Vaswani et al., 2017). Nasce così il ***Transformer***, un'architettura basata su un principio rivoluzionario: l'attenzione. Piuttosto che elaborare le parole una per volta come facevano le RNN, i *Transformer* sono in grado di analizzare tutte le parole di una sequenza contemporaneamente, e di stabilire quali parole siano più rilevanti rispetto alle altre in ciascun contesto.
Il modello è strutturato secondo una logica *encoder-decoder*:

- l'*encoder* riceve una frase in *input* – ad esempio, *"Il gatto dorme sulla sedia"* – e la trasforma in una rappresentazione numerica che cattura non solo il significato delle singole parole, ma anche le relazioni tra loro. Ad esempio, la parola "dorme" sarà legata semanticamente al soggetto "gatto" e alla sua azione;
- il *decoder*, se ad esempio deve tradurre la frase in inglese, inizierà a generare parola per parola l'*output* finale: *"The cat sleeps on the chair"*. A ogni passaggio, tiene conto sia delle parole già prodotte, sia delle informazioni codificate dall'*encoder*, per decidere quale termine generare dopo.

Per una descrizione di dettaglio sul funzionamento dei Transformer, si rimanda all'Appendice dedicata.

È giunto il momento di parlare di una cosa su cui sinora abbiamo sorvolato. Stiamo ora parlando di reti neurali che gestiscono parole. Ma come è possibile passare ad una rete neurale dei dati di testo in *input*? Due concetti sono fondamentali per comprendere come tutto ciò sia possibile: gli *embedding*, e il *positional encoding*.

L'***embedding*** è un modo per tradurre parole (o elementi discreti) in numeri. I computer non "capiscono" direttamente le parole: hanno bisogno di convertirle in vettori numerici. L'*embedding* fa esattamente questo, assegnando a ogni parola un vettore di numeri che ne rappresenta il significato in uno spazio multidimensionale. In questo spazio, parole simili avranno vettori "vicini" tra loro. Ad esempio, "gatto" e "cane" avranno rappresentazioni numeriche simili, mentre "gatto" e "astronave" saranno molto distanti.

Tuttavia, una semplice lista di parole tradotte in vettori non basta. I *Transformer*, infatti, non hanno una struttura sequenziale come le Reti Neurali Ricorrenti (RNN), quindi non sanno automaticamente in che ordine sono disposte le parole. Per colmare questa lacuna, serve il ***positional encoding***.

Il *positional encoding* comprende la posizione di ciascun elemento all'interno della sequenza. È come se si aggiungesse un "codice di posizione" a ogni parola, permettendo al modello di capire, ad esempio, che "il gatto mangia il pesce" è diverso da "il pesce mangia il gatto". Questa informazione viene combinata matematicamente con l'*embedding*, così che ogni parola non sia solo rappresentata per il suo significato, ma anche per il ruolo che svolge nella frase.

Per una descrizione di dettaglio sul funzionamento di Embedding e Positional Encoding, si rimanda alle relative Appendici.

Tornando ai Transformer, come abbiamo detto, la caratteristica innovativa principale è l'uso dell'"attenzione", una tecnica che permette al modello di focalizzarsi sugli elementi più rilevanti di un *input* e di stabilire relazioni tra le parole o le frasi.

Il meccanismo di attenzione (*"self-attention")* consente a ogni parola di "guardare" tutte le altre parole nella stessa frase per capire a quali prestare attenzione nel calcolo della sua rappresentazione. A differenza delle RNN, che trattano le parole in ordine una dopo l'altra, i *Transformer* elaborano l'intera sequenza in parallelo, e per farlo devono sapere quanto ogni parola è rilevante rispetto a tutte le altre.

Tecnicamente, ogni parola viene proiettata in tre spazi distinti attraverso tre matrici: *Query* (Q), Key (K) e *Value* (V). Per ogni coppia di parole, il modello calcola un punteggio di attenzione tra la *query* della parola corrente e la *key* di ogni altra parola. Questo punteggio indica quanto una parola dovrebbe "guardare" l'altra. I punteggi vengono poi normalizzati con *softmax* e usati per calcolare una media pesata dei *value*, che diventa la nuova rappresentazione della parola.

Consideriamo due frasi:

"Il portiere ha parato il rigore al novantesimo."
"Ho lasciato le chiavi al portiere del palazzo."

In entrambi i casi la parola "portiere" è presente, ma il suo significato è completamente diverso. Come fa il modello a distinguere i due casi?

Grazie al meccanismo di attenzione, il *Transformer* può attribuire maggiore rilevanza alle parole che circondano "portiere". Nel primo esempio, parole come "rigore" e "novantesimo" hanno una forte associazione con il contesto calcistico. Il modello assegnerà quindi pesi di attenzione più alti a queste parole rispetto ad altre della frase. Nella seconda frase, parole come "chiavi" e "palazzo" indicano