Chi continua a insegnare alla macchina?

Nessuna intelligenza nasce neutra: dati, obiettivi e addestramento ne orientano capacità, limiti e comportamento.



Gruppo Radar · Comprendere oggi. Costruire domani
by effettoantropico.life

Una persona chiede all'intelligenza artificiale di preparare una lettera. La risposta è scorrevole, ma contiene un errore: attribuisce al cliente una decisione che non ha mai preso. La persona corregge il passaggio, spiega il contesto e chiede una nuova versione. Questa volta il testo è giusto. Poi formula una terza richiesta e il sistema tiene conto della correzione.

La sensazione è quasi inevitabile: *gli ho insegnato qualcosa*.

Forse sì. Ma non necessariamente nel modo in cui immaginiamo.

La correzione potrebbe essere ancora presente nella conversazione e orientare soltanto le risposte successive. Potrebbe essere stata salvata in una memoria personale separata. Potrebbe finire in un log (registri delle attività del sistema) usato per sicurezza o analisi. Potrebbe diventare un feedback (riscontri sui risultati) che qualcuno valuterà. Oppure, dopo selezione, controlli e autorizzazioni, potrebbe entrare in un dataset (raccolte organizzate di dati) destinato a un futuro aggiornamento. Sono percorsi diversi. Hanno effetti diversi, tempi diversi e responsabilità diverse.

Il punto di partenza di questo Speciale è semplice: **ottenere una risposta diversa non dimostra che i pesi del modello siano cambiati**. Durante una normale conversazione, il modello già addestrato usa ciò che gli viene fornito nel contesto. Può adeguarsi con grande efficacia senza riscrivere i propri parametri. Gli esempi nel prompt (istruzioni o richieste fornite al modello) possono orientare il comportamento nell'esecuzione corrente proprio senza aggiornamento dei pesi. È una delle capacità documentate fin dai primi grandi modelli few-shot (capaci di svolgere un compito a partire da pochi esempi).[E02]

Capire questa differenza non è una banalità né una curiosità tecnica. Significa capire chi continua davvero a insegnare alla macchina: chi raccoglie i dati, chi definisce l'obiettivo, chi sceglie le preferenze, chi trasforma un errore in un esempio, chi conserva un log, chi decide che cosa meriti di arrivare a una nuova versione.

1. Addestrare non significa riempire un archivio

La metafora più comoda è anche una delle più ingannevoli. Immaginiamo l'addestramento come una biblioteca: milioni di libri entrano da una parte e restano, ordinati, dentro il modello. Quando facciamo una domanda, la macchina cercherebbe il passaggio migliore e lo restituirebbe.

Non funziona così.

Un modello linguistico è composto da un'architettura e da un grande insieme di valori numerici, i **parametri** o **pesi**. Durante l'addestramento, un algoritmo mostra esempi al modello, misura l'errore rispetto a un obiettivo e modifica gradualmente quei valori. Nell'architettura Transformer, le relazioni fra le parti dell'input vengono elaborate attraverso meccanismi di attenzione: non scaffali, ma trasformazioni matematiche apprese.[E01]

Il **pretraining**, cioè l'addestramento di base, espone il sistema a grandi raccolte di testi, immagini, audio o altri dati e gli fa apprendere regolarità statistiche. Nel caso del linguaggio, una forma classica dell'obiettivo è prevedere la continuazione plausibile di una sequenza. Ripetuto su scala enorme, questo compito produce capacità che vanno molto oltre il completamento meccanico di una frase: sintesi, traduzione, classificazione, scrittura, associazione di concetti.

Ma l'obiettivo iniziale conta. Contano la miscela dei dati, le lingue presenti, le ripetizioni, i filtri, le esclusioni e la qualità delle fonti. Contano anche le scelte che sembrano soltanto operative: quanto pesa un insieme rispetto a un altro, quali contenuti vengono decuplicati, quali errori vengono tollerati. **L'addestramento non raccoglie semplicemente "il mondo". Costruisce una rappresentazione a partire da una selezione del mondo.**

Questo non trasforma il modello in un archivio consultabile. Eppure sarebbe altrettanto scorretto affermare che nulla dei dati possa riemergere. Studi sperimentali hanno estratto da modelli linguistici sequenze presenti nei dati di addestramento, comprese informazioni sensibili, in condizioni e con tecniche specifiche. Il risultato non significa che ogni frase sia recuperabile o che ogni utente incontri normalmente quel rischio. Dimostra però che apprendimento statistico e memorizzazione non si escludono.[E09–E10]

> ⚠️ **Da osservare**
>
> Due formule assolute sono entrambe sbagliate: "il modello è un database" e "il modello non conserva mai nulla". La domanda corretta è quali sequenze siano state memorizzate, quanto siano estraibili, con quale accesso e in quale versione.

Ciò che rimane dopo il pretraining è quindi un artefatto numerico capace di produrre comportamenti. Non contiene una cartella nominata "Italia", una chiamata "diritto" e una chiamata "Pippo". Contiene relazioni distribuite che possono sostenere risposte su questi temi, insieme a lacune, distorsioni e talvolta frammenti memorizzati.

2. Che cosa resta dentro il modello

Dire che i dati diventano parametri non risolve la questione di che cosa resti realmente nel modello. La rende più precisa.

Rimangono anzitutto le capacità apprese: strutture linguistiche, associazioni, forme ricorrenti, procedure che il sistema riesce a ricostruire. Rimangono anche i limiti della selezione iniziale. Un audit su oltre 1.800 dataset pubblici ha documentato licenze assenti o incoerenti, provenienze difficili da ricostruire e concentrazioni geografiche e linguistiche. Non descrive l'intero contenuto dei modelli proprietari, che resta in larga parte non ricostruibile, ma dimostra che la materia prima disponibile non è uniforme né neutra.[E16]

Rimangono poi gli effetti delle ripetizioni. Un contenuto molto duplicato può esercitare un'influenza diversa da un esempio raro. Rimangono le conseguenze dei filtri: ciò che viene escluso non può insegnare una distinzione che il dataset non contiene. Rimangono le scelte di formato, le traduzioni, le etichette e le trasformazioni applicate prima che il dato raggiunga il processo di ottimizzazione.

Infine, può rimanere una memorizzazione più letterale. L'EDPB, il Comitato europeo per la protezione dei dati, ha chiarito che un modello addestrato su dati personali non può essere considerato automaticamente anonimo. La valutazione deve considerare, caso per caso, quanto sia probabile estrarre informazioni o ottenerle attraverso interrogazioni, usando mezzi ragionevolmente disponibili.[E18]

Questa posizione non dichiara che ogni modello contenga dati personali recuperabili. Impedisce la scorciatoia opposta: considerare anonimo il risultato per il solo fatto che il dataset è stato trasformato in pesi.

La distinzione è importante anche per la cancellazione. Se un documento resta in un archivio esterno, è possibile identificarlo e rimuoverlo. Se ha contribuito all'aggiornamento distribuito dei parametri, non esiste necessariamente una riga da cancellare. Le due operazioni non sono equivalenti e produrranno conseguenze diverse.

3. Dall'addestramento di base al comportamento

Il pretraining costruisce una base. Non decide da solo come il sistema si comporterà davanti a un utente.

Dopo l'addestramento iniziale interviene il **post-training**: l'insieme di tecniche che orientano il modello verso istruzioni, preferenze, regole o criteri di sicurezza. Un metodo noto utilizza dimostrazioni scritte da persone, confronti fra risposte e un modello di ricompensa che trasforma le preferenze in un segnale di ottimizzazione. È la struttura documentata nel lavoro su Instruct GPT.[E03] Altri metodi ottimizzano direttamente i dati di preferenza, senza replicare lo stesso ciclo di reinforcement learning (apprendimento per rinforzo, guidato da un segnale di ricompensa).[E05]

Qui emerge un punto decisivo: le preferenze non fluttuano nell'aria. Qualcuno sceglie gli esempi. Qualcuno formula le domande. Qualcuno decide quale risposta sia migliore. Qualcuno trasforma giudizi complessi in un dataset e in una funzione obiettivo.

Anche quando il feedback viene prodotto da un altro sistema, le regole non scompaiono. Nel metodo chiamato **Constitutional AI**, un insieme di principi guida autocritica, revisione e successivo addestramento da feedback generato dall'IA.[E04] La "costituzione" non è una coscienza morale. È un dispositivo operativo: principi scelti diventano istruzioni, esempi, preferenze e premi.

Il termine **allineamento** descrive questo tentativo di orientare il comportamento. Non è un certificato permanente. Un modello può ottenere risultati migliori nei test scelti e continuare a fallire in situazioni nuove. Può seguire bene una regola e applicarla male quando il contesto cambia. Può essere più sicuro su un rischio e meno utile su un compito legittimo.

Poi esistono i **guardrail**, cioè i limiti applicati attorno al modello: istruzioni di sistema, filtri, classificatori, permessi, blocchi di strumenti, revisioni umane. Possono impedire un'azione o modificare una risposta senza toccare i pesi. Confonderli con l'addestramento rende invisibile la governance: se un comportamento cambia, dobbiamo sapere se è cambiato il modello, la regola, il filtro o l'ambiente.

> 🧠 **Radar Insight**
>
> Il comportamento che vediamo non appartiene a un solo strato. È il risultato del modello, del contesto, della memoria, degli strumenti, dei filtri e delle decisioni del prodotto. Dire "l'IA ha deciso" comprime una catena organizzativa in un soggetto immaginario.

4. Dopo il rilascio: ciò che cambia e ciò che resta fermo

Quando un modello viene rilasciato, i suoi pesi sono normalmente un artefatto identificabile. Il servizio può cambiare molte cose attorno a essi, ma una normale richiesta non avvia per questo un nuovo training.

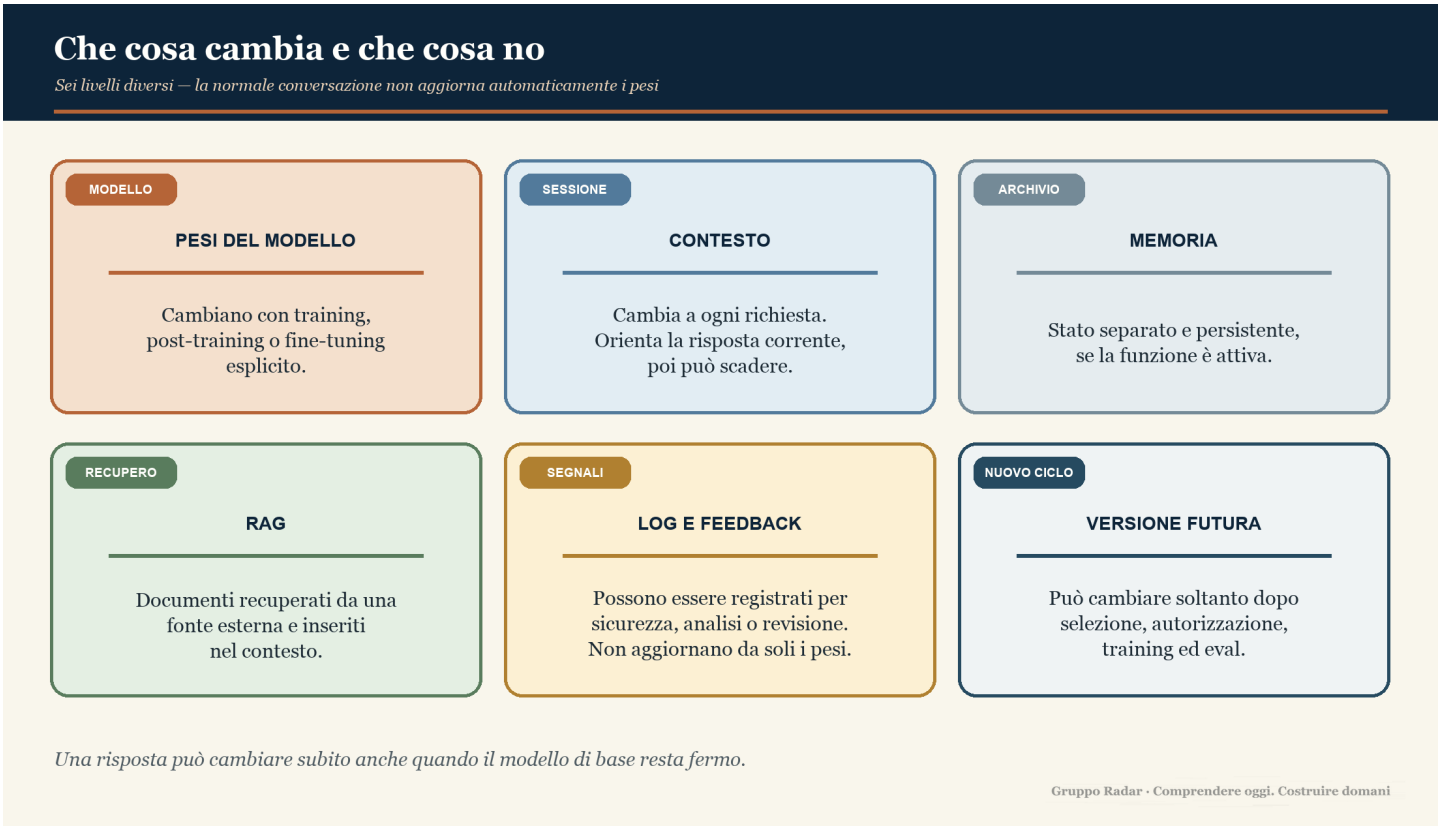
Durante l'**inferenza** — il momento in cui il modello produce una risposta — il sistema riceve una finestra di contesto: istruzioni, messaggi precedenti, eventuali documenti recuperati e risultati di strumenti. Se la correzione della lettera è ancora lì, il modello può usarla. La persona osserva un adattamento reale, ma locale alla conversazione.

Il prodotto può inoltre conservare uno **stato applicativo**. Una conversazione può essere salvata; una risposta può essere marcata come persistente; un'applicazione può riportare informazioni nelle richieste successive. La documentazione tecnica di una piattaforma API (interfacce che permettono a programmi diversi di comunicare) mostra, per esempio, che parametri di conservazione e oggetti conversazione governano lo stato delle risposte.[E25] Anche questo è un livello separato dai pesi.

Poi c'è la **memoria**. Nella documentazione di Claude, le voci di memoria sono elementi separati che possono essere letti e aggiornati durante la chat, mentre la ricerca delle conversazioni usa recupero di contenuti precedenti.[E30] La sensazione che il sistema "mi conosca" deriva quindi da informazioni reinserite nel contesto. Non serve ipotizzare un modello personale che si riaddestra a ogni frase.

Esiste infine l'**apprendimento online**, nel quale un sistema aggiorna parametri durante o subito dopo il flusso operativo. È una possibilità tecnica, ma non va attribuita a un prodotto in assenza di documentazione. Nel corpus esaminato non esiste una base comparabile per dire quanto sia diffuso nei servizi commerciali. Questo è un dato non sufficiente, non uno spazio da riempire con l'intuizione.

Ciò che resta fermo, dunque, sono i pesi della versione usata, salvo un meccanismo esplicito di aggiornamento o la sostituzione con una nuova versione. Ciò che può cambiare è quasi tutto il contorno: istruzioni, contesto, memoria, archivi, strumenti, permessi, filtri e policy (regole del servizio).



Che cosa cambia e che cosa no — sei livelli distinti; una normale conversazione può cambiare la risposta senza aggiornare i pesi.

5. Quando l'utente "insegna" davvero e quando no

Torniamo alla correzione iniziale.

Se la persona scrive "*non attribuire questa decisione al cliente*" e la frase resta nel contesto, sta guidando l'inferenza. Se il prodotto salva "*evitare attribuzioni non confermate*" in una memoria, sta aggiornando un archivio personale. Se l'azienda registra il caso fra gli errori, sta creando un log. Se un revisore trasforma l'episodio in una coppia di risposte valutate, sta costruendo un segnale di preferenza. Se quella coppia entra in un job (lavorazione programmata) di fine-tuning, allora può modificare parametri.

La distanza fra il primo e l'ultimo passaggio è un'intera filiera.

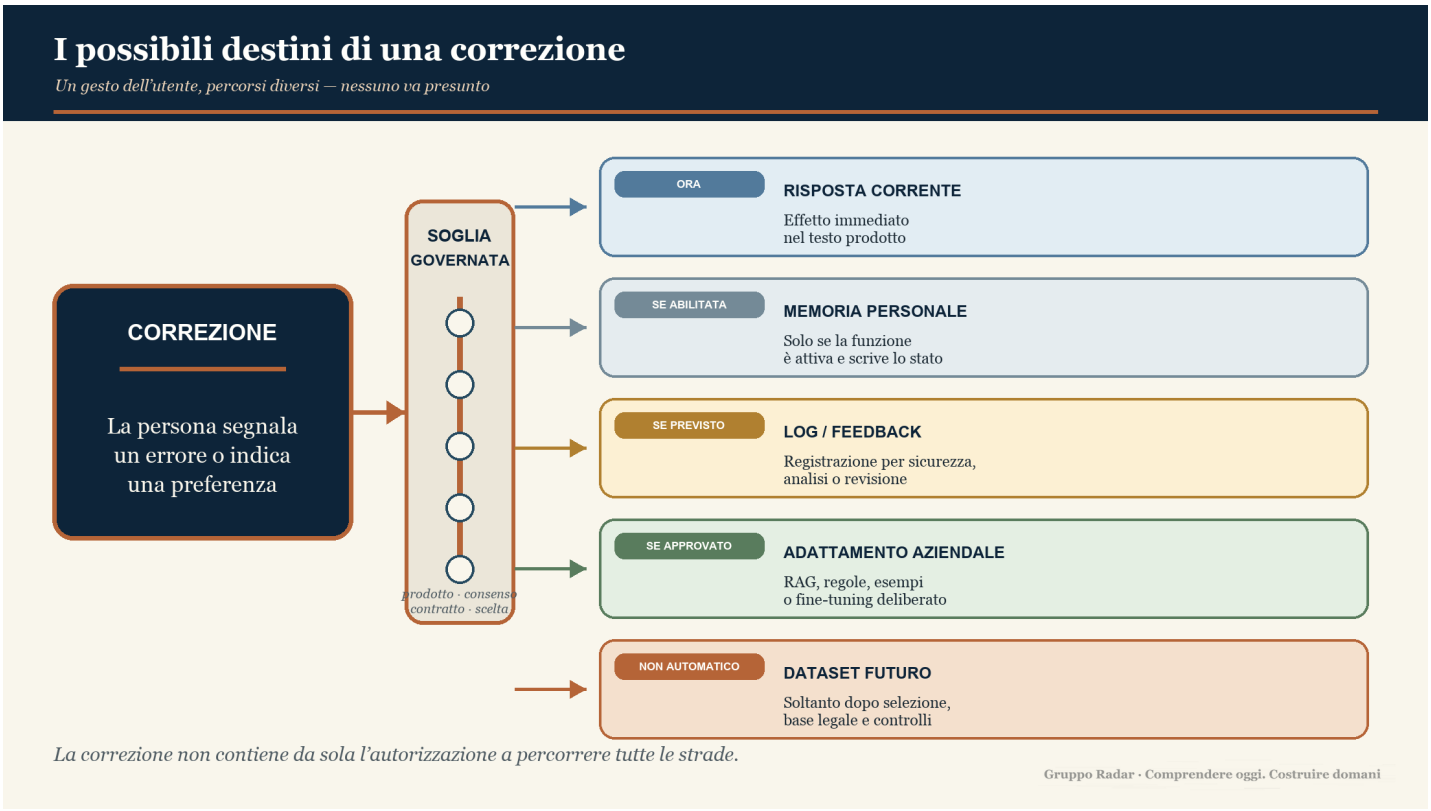
Un feedback umano diventa addestramento solo quando viene raccolto, selezionato, rappresentato in un formato utile e inserito in un ciclo di ottimizzazione. La ricerca sull'apprendimento da preferenze mostra precisamente questo: le persone confrontano comportamenti, il sistema costruisce un modello di ricompensa e l'agente viene poi ottimizzato.[E08] Il clic non è l'aggiornamento. È, al massimo, una materia prima.

Lo stesso vale per le **eval**, le valutazioni strutturate. Un'eval presenta casi di test, applica criteri e misura i risultati. Può mostrare che una nuova versione è peggiorata, che un guardrail è fragile o che una capacità non è affidabile. Non addestra

automaticamente il modello. Può però produrre esempi e priorità per il ciclo successivo.[E15, E27]

Il **red teaming**, cioè la ricerca intenzionale di modi in cui un sistema può fallire o essere abusato, segue la stessa distinzione. È una prova. Diventa materiale di addestramento solo se gli esiti vengono trasformati in dati, regole o ricompense e usati per modificare il sistema.

Questa differenza prepara già il secondo numero della trilogia: durante una crisi simulata possiamo osservare capacità e limiti di un sistema senza confondere l'osservazione con il processo che lo ha formato.



I possibili destini di una correzione — risposta corrente, memoria, log, adattamento aziendale e dataset futuro dipendono da una soglia governata.

6. Memoria, personalizzazione, RAG e fine-tuning

Quattro sistemi possono produrre la stessa impressione: "ora l'IA risponde come serve a me". Ma lo fanno in modi diversi.

La **memoria** conserva preferenze o fatti in un archivio separato e li ripropone nelle conversazioni. È relativamente visibile e può essere modificabile. La sua cancellazione non coincide necessariamente con la cancellazione della chat che l'ha originata, né con un intervento sui pesi.[E30]

La **personalizzazione** è un termine ombrello. Può significare istruzioni fisse, un profilo, una memoria, strumenti scelti, recupero di documenti o un modello adattato. Senza specificare il meccanismo, non dice quasi nulla sul trattamento dei dati.

Il **RAG**, retrieval-augmented generation, cerca informazioni in un archivio esterno e le fornisce al modello insieme alla richiesta. Il lavoro fondativo distingue una memoria parametrica, incorporata nel modello, da una memoria non parametrica, recuperata da un indice. Quest'ultimo può essere aggiornato senza riaddestrare il modello.[E07] Nelle implementazioni aziendali, un file può essere inserito in un vector store (archivio che rappresenta i documenti con vettori numerici per cercare contenuti affini), recuperato per una risposta e poi cancellato secondo il ciclo dell'archivio.[E24]

Il **fine-tuning** avvia invece un processo di ottimizzazione su esempi scelti. La documentazione API lo presenta come un job separato, con dataset e valutazioni.[E23] Tecniche come LoRA congelano gran parte del modello e addestrano componenti aggiuntivi a basso rango.[E06] Il risultato può essere un adattamento specifico, applicato al bisogno, senza riscrivere ogni parametro di base.

Per un'azienda, la differenza è pratica:

- la memoria serve a continuità e preferenze;
- il RAG serve a consultare conoscenza aggiornata e controllabile;
- il fine-tuning serve a stabilizzare forme, compiti o comportamenti attraverso esempi;
- le istruzioni servono a definire il compito nell'esecuzione;
- le eval servono a capire se tutto questo funziona.

Metterli nello stesso sacco produce due errori.

Il primo è tecnico: aspettarsi che un documento caricato diventi conoscenza permanente del modello.

Il secondo è organizzativo: non sapere più dove si trova il dato e quindi chi possa correggerlo, revocarlo o auditarlo.

7. L'impresa: conoscenza interna, processi e responsabilità

Quando un'azienda integra l'IA, non sta soltanto acquistando un modello. Sta costruendo una catena di accessi e trasformazioni.

I documenti interni possono restare in un archivio RAG. Gli esempi migliori possono diventare un dataset di fine-tuning.

I casi problematici possono entrare in un registro di eval.

I log agentici possono documentare quali strumenti sono stati chiamati e quali azioni sono state tentate.

I feedback degli operatori possono servire a correzioni locali, revisione delle procedure o adattamenti futuri.

Ogni passaggio richiede una decisione:

- chi può inserire una fonte;
- chi ne certifica la qualità e la provenienza;
- quali dati personali o riservati sono ammessi;
- quale ambiente può consultarli;
- per quanto tempo restano disponibili;
- chi approva un dataset di addestramento;
- come si misura il miglioramento;
- come si torna indietro se la nuova versione peggiora.

Questa è la differenza fra "usare l'IA" e governarla.

Anche la frase "*non usiamo i vostri dati per addestrare*" va portata dentro questa catena.

Nella documentazione OpenAI per l'API, i dati non vengono usati per addestrare o migliorare i modelli salvo opt-in (consenso esplicito ad attivare una funzione o un uso dei dati). La stessa pagina dichiara però log di monitoraggio degli abusi normalmente conservati fino a 30 giorni, eccezioni e durate diverse per lo stato applicativo di specifici endpoint (punto di accesso a un servizio software).[E22] L'esclusione di una finalità non annulla automaticamente tutte le altre.

Anthropic distingue a sua volta i servizi commerciali dagli accordi di sviluppo e dal feedback esplicito.[E28] Per il prodotto consumer (destinato agli utenti privati) dichiara che una chat eliminata viene rimossa dall'interfaccia e programmata per la cancellazione backend (sistemi che operano dietro l'interfaccia) entro 30 giorni, mentre alcune categorie de-identificate conferite al miglioramento, i feedback e i segnali di sicurezza seguono finestre differenti. Dichiara inoltre che disattivare l'uso futuro non modifica i modelli già addestrati né necessariamente interrompe un ciclo già avviato.[E29]

Questi esempi non costruiscono una classifica. Le categorie non sono omogenee e le policy appartengono a prodotti diversi. Mostrano qualcosa di più utile: **training, retention** (periodo di conservazione dei dati)**, sicurezza e stato applicativo sono piani distinti**.

> 📌 **Lo sapevi?**
>
> "Zero Data Retention" non è una proprietà astratta del modello. Può dipendere dal piano, dall'endpoint e dalle eccezioni. Anthropic, per specifici modelli coperti, dichiara una retention di sicurezza di 30 giorni anche in alcuni ambienti altrimenti ZDR. È un esempio circoscritto, non una regola generale.[E31]

L'impresa, inoltre, viene modificata dallo strumento. Per rendere la conoscenza recuperabile deve ordinare documenti, assegnare permessi, definire tassonomie (classificazioni organizzate per categorie), misurare errori, formalizzare processi che prima vivevano nella testa delle persone. Questo può aumentare chiarezza e responsabilità. Può anche creare dipendenza da formati, fornitori e metriche.

Le fonti tecniche non misurano un effetto causale uniforme di lungo periodo. È un'interpretazione Radar: mentre l'azienda addestra o configura lo strumento, lo strumento costringe l'azienda a riscrivere il proprio modo di lavorare.

8. Dove vanno dati, prompt, correzioni e log

La domanda "*dove finiscono i miei dati?*" sembra chiedere un luogo. **In realtà chiede una mappa di finalità.**

Un prompt può essere elaborato per produrre la risposta. Può essere incluso in log di sicurezza. Può restare nello stato di una conversazione. Può essere memorizzato dall'applicazione cliente. Può raggiungere un servizio esterno chiamato attraverso uno strumento. Può essere conservato come feedback. Può, con le condizioni dichiarate, contribuire al miglioramento futuro.

Un allegato può essere inviato al modello, indicizzato in un archivio, conservato fino alla cancellazione o copiato in un sistema aziendale. Un voto positivo o negativo può restare una metrica aggregata oppure diventare un esempio revisionato. La stessa interazione può generare metadati diversi dal contenuto: tempi, errori, strumenti chiamati, identificativi, segnali di abuso.

Per questo una policy va letta con almeno sette domande:

1. Qual è il prodotto?
2. Quali categorie di dati copre?
3. Quale finalità esclude?
4. Quali finalità mantiene?
5. Quanto dura ogni categoria?
6. Quali eccezioni di sicurezza o legge esistono?
7. Che cosa succede dopo opt-in, opt-out (scelta di disattivare una funzione o un uso dei dati) o cancellazione?

La documentazione dei fornitori è primaria per ciò che dichiarano sulle proprie pipeline (sequenze di raccolta, elaborazione e trasferimento dei dati). Non è indipendente e non consente di osservare dall'esterno il percorso di ogni singolo record. La destinazione effettiva di un preciso prompt resta quindi un dato non sufficiente se non esistono telemetria auditabile, contratto e configurazione del sistema concreto.

La distinzione fra consumer, API ed Enterprise (destinato alle organizzazioni) è essenziale. Una policy API non descrive automaticamente il prodotto conversazionale. Una configurazione Enterprise non si presume disponibile a un account consumer. Un'applicazione costruita da un'azienda aggiunge i propri log e depositi a quelli del fornitore. Trasferire una promessa da un ambiente all'altro produce rassicurazioni senza base.

9. Cancellazione, consenso, provenienza e limiti tecnici

Il controllo dei dati comincia dal verbo giusto.

Cancellare una chat può rimuoverla dall'interfaccia e avviare la cancellazione nel backend. **Cancellare una memoria** elimina una voce persistente usata nelle conversazioni future. **Rimuovere un file** lo toglie da un archivio o da un indice RAG. **Escludere un dato dal training futuro** impedisce che venga selezionato in cicli successivi. **Rimuovere l'influenza dai pesi** è il problema dell'unlearning.

Non sono sinonimi.

La CNIL distingue chiaramente cancellazione dal dataset e interventi sul modello. Il riaddestramento da dati corretti può essere la strada più efficace; filtri sugli output possono ridurre l'emissione di un'informazione senza cancellarla dal modello. La stessa autorità riconosce che le soluzioni tecniche non sono soddisfacenti in ogni caso. [E20]

La ricerca sull'unlearning offre metodi promettenti ma circoscritti. Un lavoro ha ridotto, su un modello specifico, alcune conoscenze relative a un universo narrativo. Gli autori lo chiamano correttamente **approximate unlearning** (rimozione approssimata dell'influenza di alcuni dati).[E13] Un benchmark successivo mostra che per valutare il risultato servono almeno misure di oblio, conoscenze mantenute, qualità generale e robustezza a domande correlate.[E14]

Di conseguenza, la frase "*il dato è stato cancellato dal modello*" richiede una prova: quale dato, quale versione, quale tecnica, quali test, quali attacchi di verifica e quale perdita di utilità.

Anche la **de-identificazione** non chiude automaticamente la questione. Ridurre identificatori può abbassare il rischio, ma l'EDPB richiede di valutare isolamento, collegamento, inferenza ed estrazione nel contesto concreto.[E18, E32]

La provenienza è l'altra metà del problema. Se non sappiamo da dove arriva un dato, è più difficile rispettare diritti, licenze, qualità e rettifiche. L'AI Act europeo impone ai fornitori di modelli per finalità generali documentazione tecnica, informazioni per i soggetti che li integrano, una policy sul copyright e un riepilogo pubblico sufficientemente dettagliato dei contenuti di addestramento.[E17] Il template della Commissione distingue dataset pubblici e privati, scraping (raccolta automatizzata di contenuti dalle pagine web), dati degli utenti e dati sintetici.[E33]

È un passo importante, ma non è una trasparenza record per record. Il riepilogo descrive categorie e fonti secondo un livello bilanciato con segreti commerciali e calendario applicativo. Non permette da solo di dire se una singola pagina, correzione o conversazione sia entrata in una specifica versione. Le linee guida EDPB sullo scraping per l'IA generativa, adottate nel luglio 2026 e ancora in consultazione, insistono su finalità, trasparenza, minimizzazione, accuratezza e trattamento delle categorie particolari.[E32] La cautela qui è doppia: il contenuto è ufficiale, ma la versione non è ancora definitiva; e il fatto che un dato sia pubblicamente accessibile non elimina automaticamente le responsabilità del suo riuso.

Poi esistono rischi propri della filiera:

- il **data poisoning**, nel quale fonti manipolate contaminano la raccolta; la fattibilità è stata dimostrata su fonti web-scale (su grandi raccolte di contenuti web) mutabili, senza provare che il fenomeno sia ovunque diffuso;[E11]
- la **contaminazione delle eval**, quando esempi di test entrano nei dati di training; può gonfiare i punteggi, ma va dimostrata con test specifici e non presunta per ogni modello;[E15]
- l'uso ricorsivo e indiscriminato di **dati sintetici**, che in condizioni sperimentali può perdere informazione sulle code della distribuzione e degradare il modello; non significa che ogni dato sintetico sia nocivo.[E12]

La qualità dell'addestramento dipende dunque da una catena di custodia: origine, trasformazioni, versioni, separazione dei test, controlli e possibilità di correggere.

10. Nessuna intelligenza nasce neutra

La parola "*neutra*" seduce perché sembra togliere responsabilità. La tecnologia sarebbe un contenitore vuoto; soltanto l'uso successivo la renderebbe buona o cattiva.

Un modello addestrato smentisce questa idea prima ancora di essere acceso.

Non è neutra la scelta dei dati. Non è neutra la decisione di includere una lingua e rappresentarne male un'altra. Non è neutra la funzione obiettivo, perché stabilisce che cosa conti come errore. Non è neutro il post-training, perché traduce preferenze e regole in esempi e ricompense. Non è neutro il filtro, perché decide quale rischio fermare e quale costo di utilità accettare. Non è neutra la metrica, perché ciò che non misura può sparire dall'attenzione dell'organizzazione.

Questo non significa che ogni comportamento sia intenzionalmente programmato.

Molti risultati emergono dall'interazione di milioni di esempi, parametri e condizioni che nessuno ha scritto uno per uno. Significa che l'assenza di un autore diretto per ogni output non elimina le scelte a monte.

La neutralità non va sostituita con un'altra semplificazione: "*il modello riflette soltanto i pregiudizi di chi lo ha costruito*". I modelli generalizzano, combinano, falliscono e producono effetti inattesi. La relazione fra una decisione di training e un singolo output non è sempre ricostruibile. Il corpus documenta concentrazioni e scelte di processo; non misura l'effetto causale di ogni lingua, cultura o principio su ogni risposta. La conclusione responsabile è più concreta: **capacità, limiti e comportamento sono orientati da una filiera di scelte, anche quando il risultato non è interamente prevedibile**. *Chi insegna alla macchina, allora?*

La insegnano i dati selezionati e quelli assenti. La insegnano gli obiettivi matematici. La insegnano i valutatori, i principi, le preferenze e i test. La insegnano le aziende quando trasformano errori in dataset e procedure. **La insegnano gli utenti solo quando le loro interazioni attraversano una soglia governata e diventano materiale per memoria, adattamento o training futuro**.

E la macchina, a sua volta, insegna qualcosa alle organizzazioni: che la conoscenza implicita non basta più, che i permessi devono essere espliciti, che l'errore deve essere misurato, che una correzione senza provenienza può diventare un nuovo problema. Non è un risultato sperimentale quantificato. *È la trasformazione organizzativa che appare quando si segue la filiera fino in fondo.*

11. Ponte: dalla formazione del modello alla prova sotto crisi

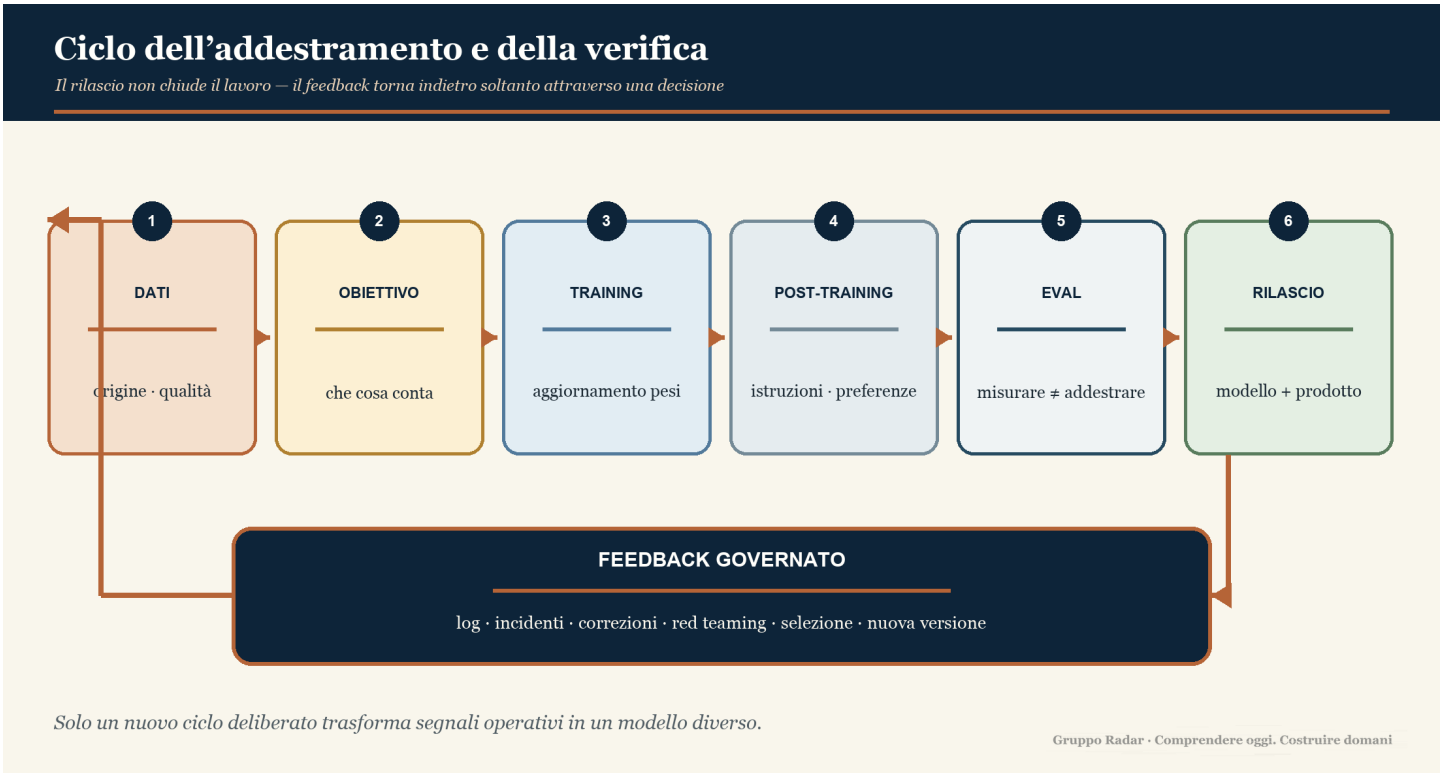
Capire come si forma una capacità è il primo passo. Il secondo è vedere che cosa accade quando quella capacità viene messa sotto pressione.

Un sistema può apparire affidabile in conversazioni ordinarie e comportarsi diversamente quando deve pianificare, usare strumenti, reagire a informazioni incomplete o operare in un ambiente avversariale. È qui che eval, red teaming e simulazioni diventano essenziali. Non perché "insegnino" automaticamente al sistema, ma perché rivelano quali comportamenti la formazione precedente abbia reso disponibili e quali confini organizzativi siano davvero efficaci.

La distinzione costruita in questo Speciale deve restare intatta:

- formare una capacità non significa osservarla;
- osservarla non significa correggerla;
- registrare un incidente non significa trasformarlo in training;
- addestrare una nuova versione non significa aver eliminato ogni vecchio rischio.

Il prossimo numero della trilogia entrerà in questo spazio: l'incidente che deve accadere prima, in un ambiente simulato, perché persone e sistemi possano essere messi alla prova prima della crisi reale.



Ciclo dell'addestramento e della verifica — eval e feedback misurano; soltanto un nuovo ciclo deliberato produce una versione diversa.

La domanda di fondo non cambierà. Diventerà più urgente.

Non soltanto: *che cosa sa fare la macchina?*

Ma: **chi ha costruito quella capacità, chi la riconosce quando emerge e chi decide che cosa farne dopo?**

Nota metodologica e tracciabilità

Questo Speciale distingue fatti tecnici (FT), risultati sperimentali (RS), dichiarazioni dei fornitori (DF), obblighi o orientamenti ufficiali (ON), interpretazioni Radar (IR) e dati non sufficienti (D). Le fonti di fornitore descrivono il proprio prodotto e non sono trasferite ad altri servizi.

La Fonte canonica, le 33 schede di evidenza e il registro dei limiti sono disponibili in:

- Fonte canonica Speciale 8A
- Matrice delle evidenze E01–E33
- Registro degli scarti e dei limiti

Restano dati non sufficienti: percorso effettivo di ogni singolo prompt; composizione granulare dei dataset proprietari; prevalenza dell'apprendimento online; cancellazione verificabile di ogni influenza dai pesi; confronto omogeneo delle retention; effetto

causale di singole scelte culturali e linguistiche; audit indipendente completo di ogni pipeline; trasformazione aziendale di lungo periodo.

Fonti essenziali

Le fonti dei provider descrivono policy e funzioni dei rispettivi prodotti: sono primarie per ciò che il soggetto dichiara, ma non costituiscono verifiche indipendenti del settore. Versioni e data di accesso sono registrate nella Fonte canonica.

- Vaswani et al., *Attention Is All You Need*, 2017.[E01]
- Brown et al., *Language Models are Few-Shot Learners*, 2020 — fonte con affiliazione provider.[E02]
- Ouyang et al., *Training language models to follow instructions with human feedback*, 2022 — fonte con affiliazione provider.[E03]
- Hu et al., *LoRA: Low-Rank Adaptation of Large Language Models*, 2021.[E06]
- Lewis et al., *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*, 2020.[E07]
- Carlini et al., *Extracting Training Data from Large Language Models*, USENIX Security 2021.[E09]
- Shumailov et al., *AI models collapse when trained on recursively generated data*, *Nature*, 2024.[E12]
- Longpre et al., *The Data Provenance Initiative*, 2023.[E16]
- Unione europea, Regolamento (UE) 2024/1689 — AI Act, articolo 53 e allegati.[E17]
- EDPB, *Opinion 28/2024 on certain data protection aspects related to AI models*.[E18]
- CNIL, *Ensuring and facilitating the exercise of data subjects' rights*, 2026.[E20]
- NIST, *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*, 2024.[E21]
- OpenAI, *Data controls in the OpenAI platform*, accesso 21 agosto 2026 — documentazione del provider riferita all'API.[E22]
- Anthropic, *How do you use personal data in model training?*, 16 marzo 2026 — policy del provider.[E28]
- Anthropic, *How long do you store my data?*, 1º luglio 2026 — policy del provider.[E29]
- Commissione europea, *Template for general-purpose AI model providers to summarise their training content*.[E33]