AI LITERACY LAB

Reti neurali e IA generativa

Guida didattica visuale, approfondita e operativa per accompagnare Neural Network Explorer

PERCHÉ QUESTA DISPENSA

Capire una rete neurale non significa imparare a memoria formule. Significa costruire un modello mentale: quali segnali entrano, come vengono trasformati, come cambiano i pesi durante l'addestramento e come questi principi si collegano ai sistemi di IA generativa.

Dispensa per docenti e formatori

Ideazione didattica e progettuale: Prof. Erasmo Modica

Realizzazione tecnica dell'app: ChatGPT (OpenAI), sulla base delle indicazioni dell'autore.

Versione ampliata e visuale • Agosto 2026

0 Come usare la guida docente

Una struttura a doppio livello: concetto tecnico essenziale + mediazione didattica per non specialisti.

FINALITÀ

Non formare progettisti di reti neurali, ma fornire un modello mentale accurato quanto basta per comprendere i meccanismi di base, evitare antropomorfismi e collegare correttamente reti neurali, Transformer e sistemi generativi.

Risultati di apprendimento attesi

- Distinguere parametri appresi, attivazioni e coefficienti di attention.
- Spiegare forward propagation, loss e backpropagation con linguaggio accessibile ma non fuorviante.
- Usare AND e XOR per mostrare il salto concettuale tra singolo neurone e rappresentazioni nascoste.
- Collegare la rete feed-forward dell'app ai Transformer esplicitando analogie e differenze.
- Far comprendere perché plausibilità linguistica e accuratezza fattuale non coincidono.

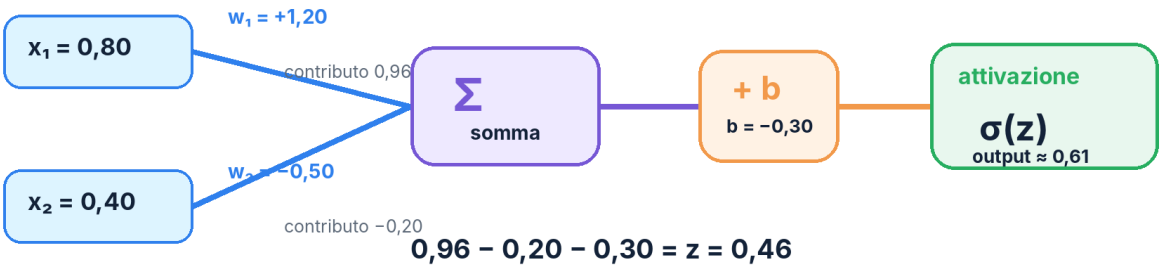
PRINCIPIO METODOLOGICO

Ogni metafora va accompagnata dal suo limite. "Volume", "soglia", "paesaggio della loss" e "indizi intermedi" sono strumenti di accesso; il docente deve poi ricondurre l'analogia alla trasformazione matematica effettiva.

1 Il nucleo matematico minimo

Quanto basta per mantenere correttezza concettuale senza trasformare il corso in un modulo di analisi numerica.

Dentro un neurone artificiale



0,96 − 0,20 − 0,30 = z = 0,46

Il neurone non “capisce”: esegue una trasformazione numerica.

$$z = W \cdot a + b \quad ; \quad a^{+} = f(z)$$

Per uno strato: trasformazione affine seguita da una non linearità.

Per il pubblico non specialistico è utile partire dal caso scalare $z = \Sigma(x_i w_i)+b$ e solo successivamente mostrare la notazione vettoriale. L’elemento da far emergere è che un neurone non contiene una regola simbolica esplicita: contiene parametri numerici che determinano come i segnali vengono trasformati.

FORMULAZIONE CONSIGLIATA

“Ogni connessione trasporta un numero e lo moltiplica per un parametro. Il neurone combina questi contributi e applica una trasformazione. La rete intera è la composizione di moltissime trasformazioni di questo tipo.”

FORMULAZIONI DA EVITARE

“Il neurone pensa”, “sceglie la caratteristica importante”, “sa che questo è un gatto”. Sono scorciatoie linguistiche che rischiano di attribuire intenzionalità al calcolo.

2 Pesi, bias, attivazioni: spiegazione e misconcezioni

Le parole più semplici sono anche quelle che generano più equivoci.

Termine	Definizione tecnica essenziale	Misconcezione frequente	Correzione
Peso	coefficiente moltiplicativo appreso	“importanza assoluta” del collegamento	il contributo dipende anche dal valore dell’attivazione in ingresso
Bias	termine additivo appreso	“pregiudizio etico”	è un offset matematico; il bias sociale è un altro concetto
Attivazione	valore prodotto dalla funzione f(z)	“neurone acceso/spento” sempre binario	molte attivazioni sono continue
Layer	insieme di trasformazioni parallele	“livello di concetti umani”	le rappresentazioni possono essere distribuite e non interpretabili singolarmente

Pesi appresi vs attention weights

DISTINZIONE CRITICA

Nei Transformer, “peso” può indicare sia i parametri appresi del modello sia, colloquialmente, i coefficienti di attention calcolati dinamicamente per un dato input. Nella formazione conviene usare due espressioni diverse: parametri appresi e coefficienti di attention.

3 AND come ponte tra logica e neurone

L'obiettivo non è insegnare le porte logiche, ma rendere visibile il concetto di frontiera decisionale.

PROBLEMA DIDATTICO

Molte dispense mostrano la tabella AND e poi saltano direttamente a pesi casuali. Per un non specialista manca il nesso causale: perché proprio quei pesi e quel bias producono AND?

x_1	x_2	AND
0	0	0
0	1	0
1	0	0
1	1	1

$$z = x_1 + x_2 - 1{,}5 \quad \rightarrow \quad \text{step}(z)$$

La soglia 0 equivale a chiedere: la somma x_1+x_2 è almeno 1,5? Solo 1+1 soddisfa la condizione.

Input	Somma x_1+x_2	Dopo bias −1,5	Output
0,0	0	−1,5	0
0,1	1	−0,5	0
1,0	1	−0,5	0
1,1	2	+0,5	1

DOMANDA DA PORRE

"Se volessi OR invece di AND, quale soglia dovrebbe cambiare?" Lasciare sperimentare il bias porta naturalmente a scoprire che una soglia più bassa (es. b = −0,5) basta per attivare l'uscita quando almeno un input vale 1.

4 XOR spiegato bene: il motivo dello strato nascosto

È il punto in cui il corsista può capire davvero perché “più strati” non è solo “più neuroni”.

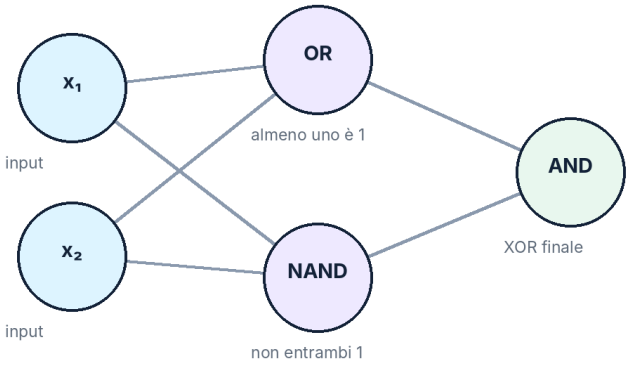
Perché XOR è speciale: serve una rappresentazione intermedia

Tabella di verità XOR

x_1	x_2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Vero solo quando gli input sono diversi.

Una soluzione con uno strato nascosto



Idea chiave: lo strato nascosto costruisce due “indizi” utili (OR e NAND). L’output li combina. È un esempio semplice di rappresentazione appresa.

1. Il problema geometrico

I quattro punti (0,0), (0,1), (1,0), (1,1) hanno etichette 0,1,1,0. Nessuna singola retta separa i due punti positivi dai due negativi. Un neurone lineare + soglia produce una sola frontiera lineare: quindi non può implementare XOR.

2. La rappresentazione intermedia

Un hidden layer può trasformare gli input in due caratteristiche: h_1 = OR e h_2 = NAND. Nel nuovo spazio (h_1,h_2), XOR diventa semplicemente AND. Questa costruzione non è l’unico modo di realizzare XOR, ma è pedagogicamente trasparente.

x_1	x_2	OR	NAND	AND(OR,NAND)=XOR
0	0	0	1	0
0	1	1	1	1
1	0	1	1	1
1	1	1	0	0

3. Una parametrizzazione a soglia

$$h_1 = \text{step}(x_1+x_2-0{,}5) \quad ; \quad h_2 = \text{step}(-x_1-x_2+1{,}5)$$

h_1 realizza OR; h_2 realizza NAND.

$$y = \text{step}(h_1+h_2-1{,}5)$$

L'output realizza AND delle due caratteristiche intermedie.

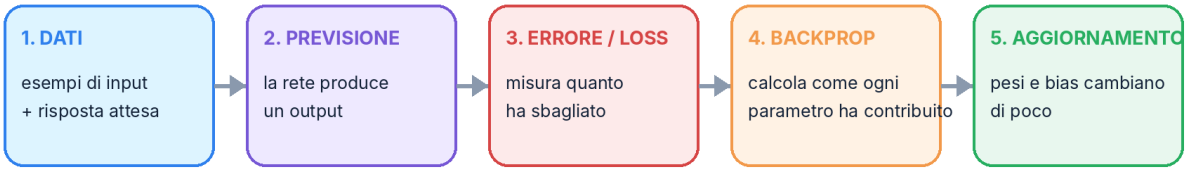
MESSAGGIO DIDATTICO

Lo strato nascosto non è un "luogo dove la rete pensa": è una trasformazione che crea nuove coordinate/rappresentazioni del problema. La profondità permette di comporre trasformazioni e ottenere frontiere decisionali più articolate.

5 Addestramento: dalla loss all'aggiornamento dei parametri

Spiegare il ciclo senza nascondere l'idea del gradiente.

Come impara una rete: un ciclo ripetuto moltissime volte



↻ **Ripeti: se la loss diminuisce, la rete sta adattando i parametri ai dati di addestramento.**

La loss è una funzione scelta per quantificare l'errore. La backpropagation, applicando la regola della catena, permette di calcolare in modo efficiente le derivate della loss rispetto ai parametri. L'ottimizzatore usa questi gradienti per proporre piccoli aggiornamenti.

$$\theta \leftarrow \theta - \eta \cdot \nabla L(\theta)$$

θ = parametri; η = learning rate; ∇L = gradiente della loss.

SEMPLIFICAZIONE CORRETTA

"Il gradiente indica localmente come cambia l'errore se modifichiamo i parametri. L'ottimizzatore usa questa informazione per fare piccoli passi." È più corretto di "la rete capisce dove ha sbagliato".

Tre fenomeni da introdurre

Fenomeno	Spiegazione accessibile
Underfitting	il modello non riesce neppure ad adattarsi bene ai dati di addestramento
Overfitting	si adatta troppo ai dettagli dei dati di addestramento e generalizza peggio
Generalizzazione	riesce a produrre buoni risultati su esempi nuovi, non visti durante l'addestramento

6 Dalla rete feed-forward ai Transformer

Analogia sì, equivalenza no.

Un Transformer contiene componenti feed-forward ma introduce soprattutto self-attention, residual connections, normalizzazioni e una struttura pensata per elaborare sequenze. La web app rende visibili pesi, bias e attivazioni di una rete piccola; questa trasparenza non va confusa con l'architettura completa di un LLM.

Concetto nell'app	Analogo utile in un LLM	Differenza da dichiarare
input numerico	token/embedding e rappresentazioni	il testo viene codificato in vettori ad

	contestuali	alta dimensionalità
peso di connessione	parametri delle matrici del modello	nei LLM i parametri sono numerosissimi e distribuiti
attivazione del neurone	attivazioni interne dei blocchi	non c'è un unico "neurone della parola"
output	logit/probabilità del token successivo	l'output viene prodotto su un vocabolario molto grande

ATTENTION

I coefficienti di attention sono calcolati per ogni input e determinano come le posizioni della sequenza si combinano nel costruire nuove rappresentazioni. Sono dinamici; i parametri che li producono sono invece appresi durante l'addestramento.

7 Simulare la generazione senza introdurre falsi modelli mentali

Come usare bene l’esempio “Il gatto beve …”.

La simulazione deve mostrare una catena concettuale: contesto → attivazioni → logit → softmax → distribuzione → campionamento. I numeri sono didattici. Il caso “acqua” è utile perché permette anche di discutere la differenza fra plausibilità linguistica, stereotipi dei dati e conoscenza del mondo.

ESEMPIO CORRETTO

Per “Il gatto beve …”, acqua deve essere il candidato principale nella simulazione. Latte può rimanere come alternativa meno probabile per mostrare che i modelli apprendono anche associazioni culturali presenti nei dati.

Logit, softmax e temperatura

I logit sono punteggi grezzi. La softmax li trasforma in una distribuzione positiva che somma a 1. La temperatura modifica la “concentrazione” della distribuzione prima del campionamento: valori più bassi favoriscono scelte più deterministiche, valori più alti aumentano la varietà.

$$p_i = \exp(z_i/T) / \Sigma_j \exp(z_j/T)$$

Formula concettuale della softmax con temperatura T.

CAUTELA

Probabilità del token successivo non equivale a probabilità che un fatto sia vero. Questa distinzione è centrale per spiegare allucinazioni e necessità di verifica.

8 Percorso didattico da 60 minuti

Una sequenza pronta per un breve modulo di formazione.

Tempo	Attività	Obiettivo
0-10 min	Domanda iniziale: “Che cosa immaginate quando sentite neurone artificiale?”	far emergere preconoscenze e antropomorfismi
10-20 min	Neurone: input, peso, bias, attivazione con esempio numerico	costruire il nucleo matematico
20-32 min	Preset AND, poi modifica del bias	comprendere soglia e frontiera
32-45 min	XOR e hidden layer con OR + NAND	comprendere rappresentazioni intermedie
45-55 min	Simulazione generativa: token, logit, softmax, temperatura	collegare alla GenAI
55-60 min	Exit ticket a tre domande	verificare il modello mentale

EXIT TICKET

1) Che cosa cambia durante l’addestramento? 2) Perché XOR è più istruttivo di AND? 3) Perché la parola più probabile non è necessariamente vera?

9 Percorso didattico da 90-120 minuti

Per corsi di formazione con sperimentazione attiva.

Fase	Attività del corsista	Domanda del formatore
Esplora	modifica input e un singolo peso	“Quale quantità è cambiata davvero?”
Confronta	stessa rete con attivazioni diverse	“Perché l’output non cambia in modo lineare?”
Costruisci	ricostruisci AND regolando bias/pesi	“Che soglia serve?”
Problematizza	prova XOR con un solo neurone	“Perché non riesci?”
Rappresenta	usa OR/NAND nello hidden layer	“Che nuovo spazio hai costruito?”
Collega	simula token successivo e temperatura	“Quali numeri cambiano durante la generazione?”
Rifletti	discute allucinazione e verifica	“Plausibile = vero?”

Attività di gruppo: “Spiega senza metafore”)

Dividere i corsisti in gruppi. Ogni gruppo riceve un termine (peso, bias, attivazione, loss, attention, temperatura) e deve produrre: una definizione tecnica di una frase, una metafora utile, il limite della metafora. È un ottimo esercizio di AI literacy perché separa precisione concettuale e comunicazione.

10 Domande frequenti e risposte consigliate

Una piccola banca di interventi per la conduzione del laboratorio.

“Ogni neurone rappresenta una parola o un concetto?”

No. Alcune unità possono correlare con caratteristiche interpretabili, ma le rappresentazioni sono generalmente distribuite su molte dimensioni e molti neuroni.

“Il modello cambia i pesi mentre mi risponde?”

Normalmente no: durante l’inferenza usa parametri già appresi. Cambiano le attivazioni interne e i coefficienti di attention calcolati dal contesto.

“Se la temperatura è zero il modello dice sempre la verità?”

No. Rende la scelta più deterministica rispetto ai punteggi del modello, ma non trasforma i punteggi in un meccanismo di verifica fattuale.

“Perché serve il bias?”

Per spostare la soglia/risposta della trasformazione. Senza bias alcune funzioni sarebbero vincolate a passare per l’origine e avrebbero meno flessibilità.

“XOR dimostra che una rete profonda è intelligente?”

No. Dimostra solo che una rappresentazione intermedia non lineare consente di modellare una relazione che un singolo classificatore lineare non può separare.

11 Misconcezioni da diagnosticare

Gli errori concettuali più frequenti sono indicatori utili per la formazione.

Misconcezione	Segnale osservabile	Intervento
peso = importanza assoluta	“la connessione più pesante è sempre la più importante”	mostrare che contributo = attivazione × peso
bias = pregiudizio	confusione tra termine matematico e bias sociale	separare esplicitamente i due significati
neurone = decisione cosciente	uso di “vuole”, “sa”, “capisce”	riformulare in termini di trasformazione numerica
training = memorizzazione	“ha salvato tutte le frasi”	introdurre generalizzazione e rappresentazioni distribuite
generazione = apprendimento in tempo reale	“assegna nuovi pesi mentre scrive”	distinguere parametri fissi e attivazioni dinamiche
probabile = vero	fiducia eccessiva nel testo fluido	usare un esempio plausibile ma falso e chiedere verifica

12 Valutazione formativa

Una griglia leggera per osservare la qualità del modello mentale.

Dimensione	Iniziale	Intermedio	Solido
Neurone	usa antropomorfismi	descrive input/peso ma confonde bias/attivazione	spiega correttamente z e f(z)
AND/XOR	ricorda solo tabelle	sa che XOR “serve più rete”	spiega la non separabilità lineare e il ruolo hidden
Apprendimento	“la rete memorizza”	sa che i pesi cambiano	collega loss, gradiente, aggiornamento e generalizzazione
GenAI	“sceglie parole che sa”	parla di probabilità	distingue token, logit, softmax, temperatura e verità
Transformer	assimila tutto a rete classica	nomina attention	distingue parametri appresi, attivazioni e attention dinamica

USO

La griglia è pensata per feedback formativo, non per attribuire un voto. Può essere usata come checklist finale o per confrontare le risposte prima/dopo il laboratorio.

13 Approfondimento tecnico opzionale

Per il formatore che vuole avere un livello in più, senza appesantire la lezione.

Backpropagation

La backpropagation applica sistematicamente la regola della catena per calcolare le derivate della loss rispetto a ciascun parametro. Il vantaggio è computazionale: riutilizza risultati intermedi e rende praticabile l'ottimizzazione di reti con moltissimi parametri.

Cross-entropy

Nei problemi di classificazione e nei modelli linguistici, una loss comune confronta la distribuzione prevista con il target. Per un token corretto con probabilità p, il contributo tipico è −log(p): assegnare probabilità molto bassa al target viene penalizzato fortemente.

Embedding

Un embedding rappresenta un elemento discreto, come un token, tramite un vettore continuo. Durante l'elaborazione il vettore viene trasformato dal contesto; nei Transformer, quindi, la stessa parola può produrre rappresentazioni contestuali differenti in frasi differenti.

Residual connections

Le connessioni residue permettono di sommare l'input di un blocco alla sua trasformazione. Aiutano l'ottimizzazione di reti profonde e favoriscono il flusso dell'informazione attraverso molti strati.

14 Glossario ragionato

Termini da usare in modo coerente durante la formazione.

Termine	Definizione
Attivazione	valore intermedio prodotto da un'unità o da un blocco per un dato input
Bias	parametro additivo di una trasformazione
Embedding	vettore che rappresenta un elemento discreto in uno spazio continuo
Epoch	passaggio completo sui dati di training
Gradiente	insieme delle derivate della loss rispetto ai parametri
Inference	uso del modello addestrato senza normale aggiornamento dei parametri
Learning rate	ampiezza degli aggiornamenti proposti dall'ottimizzatore
Logit	punteggio non normalizzato per un candidato
Loss	funzione obiettivo da minimizzare
Parametro	numero appreso e mantenuto nel modello
Softmax	normalizzazione esponenziale dei logit in una distribuzione
Token	unità discreta della sequenza elaborata
Transformer	architettura basata su self-attention e blocchi feed-forward
Weight	parametro moltiplicativo appreso; da distinguere dai coefficienti dinamici di attention

15 Banca di esempi didattici pronta per l'aula

Esempi brevi, calcoli guidati e domande di mediazione per rendere visibili concetti che altrimenti restano astratti.

1. PESO - "Stesso dato, importanza diversa"

Scrivere alla lavagna: x = 0,8. Calcolare 0,8×0,1; 0,8×1; 0,8×(-1). Chiedere: "È cambiato il dato o è cambiato il modo in cui la rete lo usa?". La risposta corretta è la seconda.

- Domanda guida: che cosa significa un peso vicino a zero?
- Misconcezione da evitare: "peso" non è il peso fisico dell'informazione.

2. BIAS - "La stessa evidenza, una soglia diversa"

Usare $z = x_1w_1+x_2w_2+b$ con somma ponderata pari a 1,2. Con b=-0,5 si ha z=0,7; con b=-1,5 si ha z=-0,3. Chiedere quale parametro ha modificato la predisposizione del neurone ad attivarsi.

3. ATTIVAZIONE - "Non basta sommare"

Mostrare z=-2, 0, 2 e confrontare ReLU e Sigmoid. Far verbalizzare: ReLU azzera i negativi; Sigmoid comprime tutto tra 0 e 1. Questo prepara l'idea che la non linearità permette alle reti profonde di rappresentare funzioni più complesse.

15.1 AND e XOR: sequenza didattica consigliata

Far predire sempre il risultato prima di mostrare il calcolo.

Caso	AND: $z=x_1+x_2-1,5$	Output
0,0	-1,5	0
0,1	-0,5	0
1,0	-0,5	0
1,1	+0,5	1

Perché AND è didatticamente utile

Con soli due pesi e un bias si vede una regola completa. Cambiando soltanto il bias da -1,5 a -0,5, la stessa struttura diventa OR. È un modo molto efficace per mostrare che i parametri definiscono il comportamento della rete.

XOR - la frase da usare

"Un solo neurone può tracciare una sola frontiera lineare. XOR richiede di ricodificare prima il problema." Poi costruire h_1=OR e h_2=NAND e infine AND(h_1,h_2). Il punto non è la porta logica in sé: è mostrare che uno strato nascosto crea rappresentazioni intermedie utili.

x_1	x_2	OR	NAND	XOR
0	0	0	1	0
0	1	1	1	1
1	0	1	1	1
1	1	1	0	0

Mini-attività: "Inventate una rappresentazione intermedia"

Prima di mostrare OR e NAND, chiedere ai gruppi: "Quali due domande intermedie potreste fare agli input per rendere XOR più semplice?". Anche risposte non perfette sono utili: spostano l'attenzione dal calcolo alla costruzione di feature.

15.2 Addestramento: esempi per evitare il “mistero” della backpropagation

Separare sempre tre livelli: errore osservato, gradiente, aggiornamento del parametro.

Micro-esempio senza derivate

Modello $\hat{y}=w\cdot x$, $x=1$, target $y=1$. Se $w=0{,}2$, $\hat{y}=0{,}2$. Se dopo un piccolo aggiornamento $w=0{,}36$, $\hat{y}=0{,}36$: l’errore è diminuito. Ripetere per 3-4 passi. Poi spiegare che il gradiente serve a scegliere sistematicamente direzione e ampiezza dell’aggiornamento.

Esempio di learning rate

Se la direzione corretta suggerisce di aumentare w, un learning rate molto piccolo produce correzioni lente; uno troppo grande può superare il punto migliore e oscillare. Metafora controllata: non “passi nel buio”, ma dimensione numerica dell’aggiornamento lungo il gradiente.

Overfitting in un caso scolastico

Classificatore “cane/non cane” addestrato quasi solo con cani su erba verde. Se usa il prato come indizio dominante, può fallire su cani in casa. Domande: quale correlazione spuriosa ha appreso? quali dati aggiungereste al training set?

15.3 IA generativa: esempi pronti da proiettare

Numeri volutamente illustrativi; non attribuirli a modelli reali.

Prompt parziale	Candidato più plausibile	Che cosa discutere
Il gatto beve ...	acqua	probabilità ≠ regola rigida; contesto e dati appresi
La capitale d'Italia è ...	Roma	plausibilità molto alta, ma non sostituisce la verifica delle fonti
2 + 2 = ...	4	pattern fortissimo; distinguere generazione linguistica da calcolo verificato
Il cielo è ...	blu	dipendenza dal contesto: notte, temporale, tramonto cambiano la distribuzione

Temperatura: dimostrazione da 2 minuti

Scrivere tre candidati con probabilità 0,70 / 0,20 / 0,10. Spiegare che una temperatura bassa rende la distribuzione più concentrata, una alta più piatta. Domanda: "A temperatura alta il modello sa più cose?" Risposta: no, cambia la politica di campionamento, non la conoscenza appresa.

Attention: una frase utile

"Il cane che inseguiva i gatti era stanco." Chiedere: a chi si riferisce "era stanco"? Usare l'esempio per introdurre l'idea che una posizione combina informazione da altre posizioni. Precisare che i coefficienti di attention non sono spiegazioni causali complete del comportamento del modello.

Polisemia: banca

Confrontare "deposito i soldi in banca" e "una banca di sabbia". Il token viene elaborato nel contesto e assume rappresentazioni contestualizzate differenti. Ottimo esempio per contrastare l'idea "un neurone corrisponde a un concetto".

15.4 Domande diagnostiche con risposte attese

Usarle come formative assessment durante la dimostrazione.

Domanda	Risposta attesa / indicatore
Durante una risposta di un LLM cambiano i pesi?	Di norma no: cambiano soprattutto input, stati interni e attivazioni; i pesi sono usati in inferenza.
Peso e attention weight sono la stessa cosa?	No. I primi sono parametri appresi persistenti; i secondi sono coefficienti calcolati dinamicamente sul contesto.
Perché XOR richiede hidden layer in questa costruzione?	Perché una singola frontiera lineare non separa i casi; lo strato nascosto ricodifica il problema.
Probabilità alta significa verità?	No. Significa alta plausibilità secondo il modello e il contesto.
Che cosa fa il bias?	Sposta il livello di attivazione/soglia del neurone indipendentemente dai singoli input.

Chiusura consigliata

Far chiedere a ogni corsista di completare oralmente: “Una rete neurale non memorizza soltanto regole: trasforma rappresentazioni numeriche attraverso parametri appresi. Durante la generazione, i parametri vengono usati per produrre attivazioni e punteggi da cui deriva una distribuzione di probabilità.”

16 Nota di trasparenza e uso responsabile

DICHIARAZIONE

Ideazione didattica e progettuale: Prof. Erasmo Modica · Realizzazione tecnica: ChatGPT (OpenAI), sulla base delle indicazioni dell'autore. L'impiego dell'IA è dichiarato per scelta di trasparenza; l'app è progettata anche come strumento di AI literacy, in coerenza con le finalità richiamate dall'art. 4 del Regolamento (UE) 2024/1689 — AI Act.

La web app e questa guida utilizzano simulazioni semplificate. Non espongono né ricostruiscono i parametri reali di specifici modelli commerciali. Il loro scopo è rendere osservabili concetti altrimenti astratti e favorire una comprensione critica dell'IA.