LINUX DAY — ITI G. MARCONI, PONTEDERA

AI LOCALE, PRIVACY E ARCHITETTURE RAG

Capire come funziona un LLM, imparare a farlo girare sul proprio computer e farlo lavorare sui nostri documenti

Canocchi Lorenzo, Donati Paolo

ROADMAP

Anatomia dell’AI

Dai token alla rete neurale: comprendere i meccanismi interni degli LLM.

Inferenza Locale

Modelli open weight, requisiti hardware ed esecuzione offline sul nostro computer.

Connessione Dati (RAG)

Integrazione sicura dei documenti privati tramite Retrieval-Augmented Generation.

TUTTI NOI ORMAI CONOSCIAMO QUESTI LLM







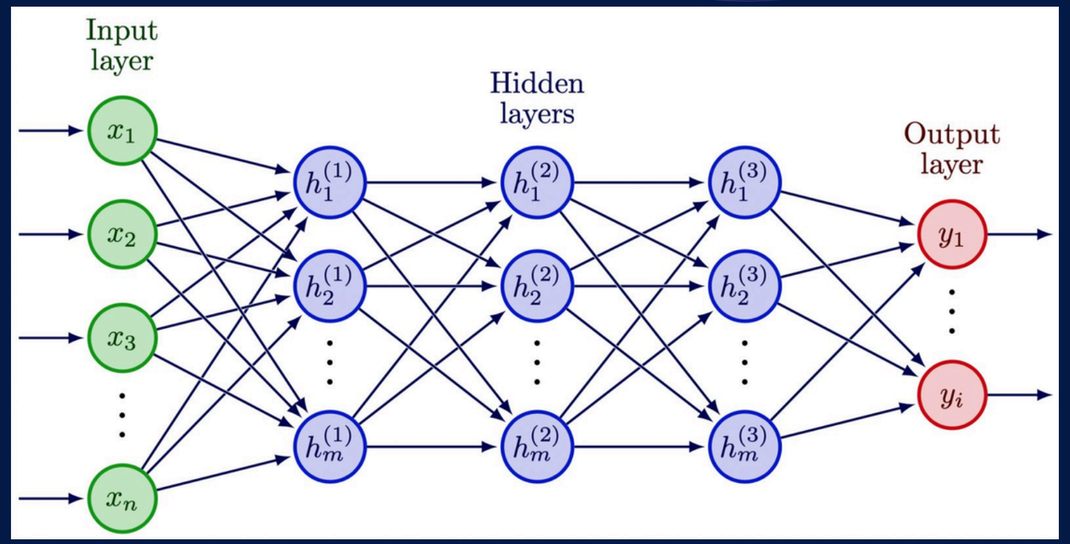
Ma come funzionano davvero?

LA RETE NEURALE

1. Input Vettoriale & Token
Il testo viene frammentato in token e convertito in sequenze numeriche per l'elaborazione.

2. Hidden Layers & Pesi
Il Transformer elabora i vettori attraverso diversi layer, utilizzando operazioni matematiche e i pesi appresi durante l'addestramento. Infine mette in relazione fra loro i token e interpreta il contesto.

3. Inferenza Probabilistica
Dopo l'elaborazione, il modello calcola una distribuzione di probabilità sui possibili token successivi. Seleziona un token e ripete il processo fino a generare la risposta.



TOKEN E CONTEXT WINDOW

Token

L'unità atomica di elaborazione del testo.

Un token non corrisponde necessariamente a una parola intera.

> **Regola empirica:** 100 token corrispondono approssimativamente a circa 75 parole in lingua inglese o 60 in lingua italiana.

Context window

La memoria di lavoro attiva del modello.

Determina la quantità massima di token elaborabili in un singolo scambio.

> **Soglia critica:** Se il contenuto supera il limite, una parte del contesto deve essere esclusa o gestita diversamente.

PARAMETRI E QUANTIZZAZIONE

Parametri

I pesi, associati ad ogni sinapsi e ottimizzati in fase di addestramento, utilizzati dalla rete per elaborare i dati.

Impatto logico: Solitamente un modello da 4B (miliardi) di parametri offre maggiore profondità rispetto a modelli ridotti.

Quantizzazione

Riduzione matematica della precisione dei pesi (es. da 16-bit a 4-bit o 6-bit) per abbattere l'occupazione di RAM.

Compromesso: Riduce l'ingombro hardware consentendo l'esecuzione su laptop consumer standard.

DUE MODI DI USARE L'AI

AI in Cloud

Potenza illimitata:
Accesso immediato ai modelli più grandi e aggiornati (GPT-Astra, Opus 5.5).

Integrazione rapida:
API e servizi pronti all'uso senza gestione dell'hardware.

Privacy delegata:
I dati, compresi quelli sensibili, lasciano la rete locale.

AI in Locale

Funzionamento offline:
Installato il modello, puoi utilizzarlo anche senza connessione Internet.

Vincolo hardware:
Prestazioni limitate dalla RAM/VRAM disponibile localmente.

Controllo totale:
Nessuna trasmissione dati verso server esterni.

COSA SERVE PER FAR GIRARE UN'AI IN LOCALE



Modello

File strutturato contenente i pesi addestrati e quantizzati (es. formato GGUF).

+



Runtime

Motore di esecuzione, spesso open source (es. Llama.cpp), che gestisce la logica di inferenza.

+



Hardware

Quantità di RAM sufficiente a caricare i pesi e CPU/GPU per i calcoli matriciali.

IL MODELLO OPEN-WEIGHT CHE USEREMO OGGI

Spark-X2.5-4B-GGUF-Q6_K_XL

4B (Parametri)

Quattro miliardi di pesi. Ottimizzato per girare in modo fluido anche su laptop senza GPU dedicate.

GGUF

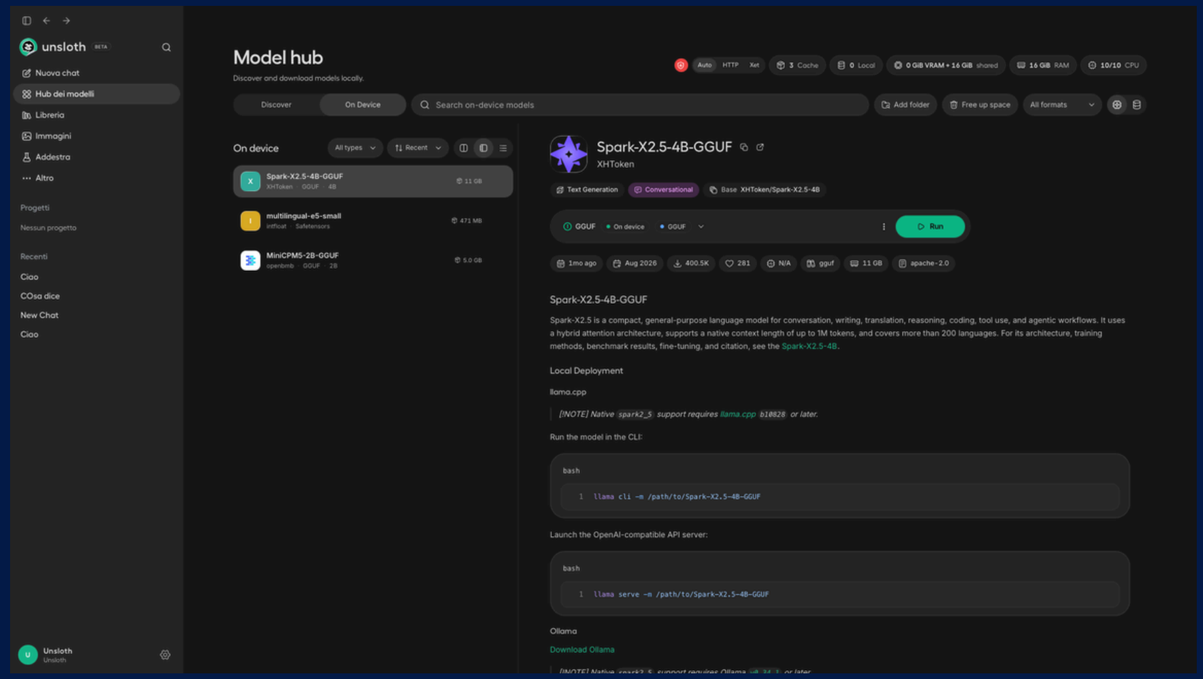
Formato di file modulare ottimizzato per velocizzare l'inferenza usando direttamente la CPU o la GPU.

Q6_K_XL

Quantizzazione a 6 bit. Consuma meno RAM rispetto al modello originale, preservando altissima coerenza.

UNSLOTH DESKTOP

- **Integrazione Click-and-Run:** Ricerca e download dei modelli direttamente da Hugging Face all'applicazione
- **Controllo Hardware Locale:** Monitoraggio in tempo reale della VRAM utilizzata e della lunghezza del contesto
- **Server API locale**: espone API standard per permettere la comunicazione con i modelli eseguiti in locale.





OTTIMO. IL MODELLO GIRA IN TOTALE SICUREZZA SUL NOSTRO COMPUTER.

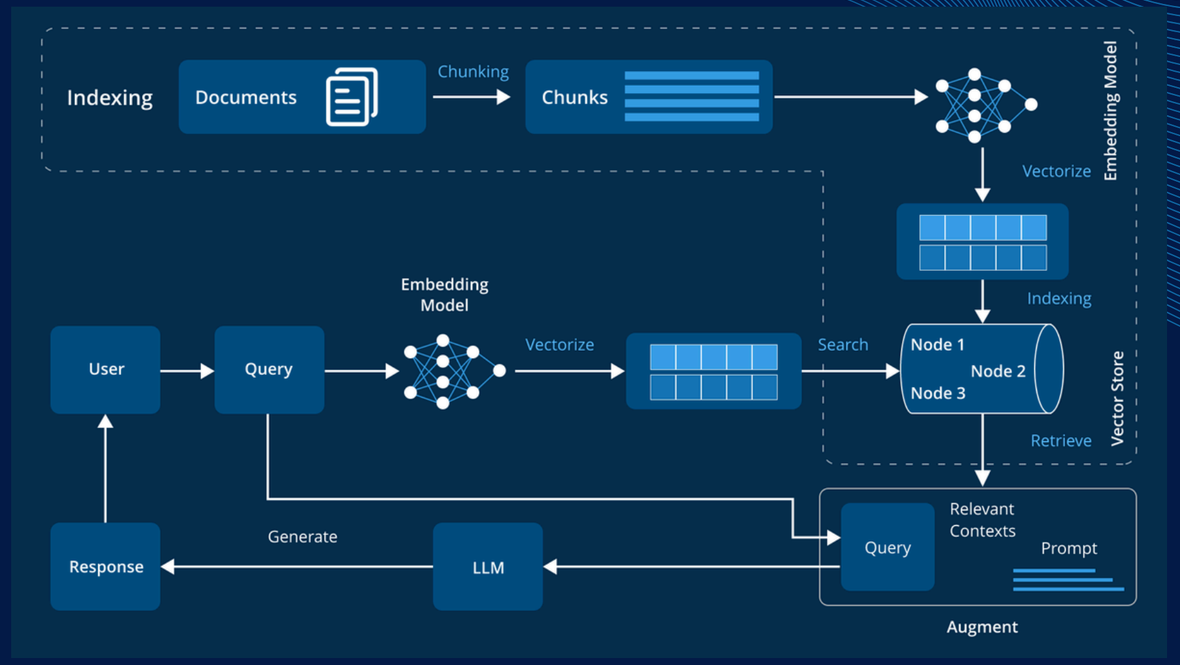
Ma come fa a conoscere i nostri documenti privati
se non è stato addestrato su di essi?

RETRIEVAL-AUGMENTED GENERATION (RAG)

1. Ingestion & Embeddings
I documenti vengono suddivisi in chunk e trasformati in embeddings.

2. Semantic Retrieval
Si scelgono i vettori dei paragrafi inerenti alla domanda.

3. Generation
L'LLM riceve i paragrafi come contesto e formula la risposta.



RAG VS CONTESTO DIRETTO

Allegare file al prompt

- ✕ Il contenuto del file occupa parte della context window, lasciando meno spazio alla conversazione.
- ✕ Nei documenti lunghi, molti contenuti possono essere irrilevanti per la domanda e rendere più difficile individuare i passaggi utili.

Utilizzare il RAG System

- ✓ Inietta nel prompt esclusivamente i 3 o 4 frammenti di testo altamente pertinenti.
- ✓ Può interrogare archivi con migliaia di documenti senza impattare drasticamente sulle prestazioni dell'LLM.

ADESSO IMPARIAMO AD USARE QUESTI STRUMENTI!

GRAZIE A TUTTI DELL'ATTENZIONE

CI SONO DOMANDE?