S. I. A. s.r.l.
SVILUPPO INIZIATIVE ATTUARIALI

L'impatto dell'Intelligenza Artificiale nei processi di lavoro attuariale

13 Maggio 2026

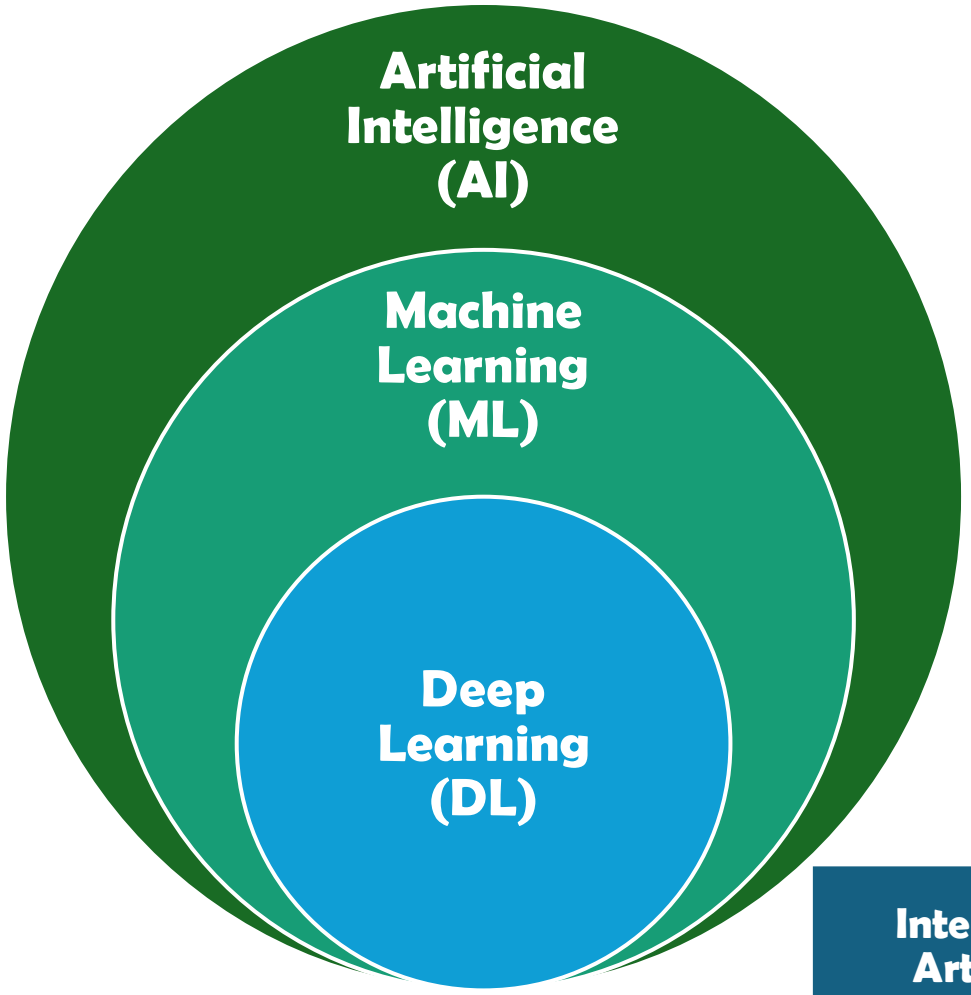
Claudio G. Giancaterino

Agenda

-Introduzione: l'universo dell'Intelligenza Artificiale
-Evoluzione dell'Intelligenza Artificiale Generativa con l'avvento di ChatGPT
-Evoluzione dell'Intelligenza Artificiale Predittiva
-Interpretabilità e gestione del rischio di modello per la validazione attuariale
-Ricerca, casi d'uso e prospettive future

L’universo dell’Intelligenza Artificiale

L'Universo dell'Intelligenza Artificiale



Data Science

Intelligenza Artificiale	Apprendimento Automatico	Apprendimento Profondo	Scienza dei Dati
• Si concentra sulla creazione di sistemi intelligenti	• Consente ai computer di apprendere dai dati e fare previsioni	• Utilizza reti neurali multilivello per apprendere rappresentazioni complesse dai dati	• Combina strumenti statistici e computazionali per elaborare grandi quantità di dati

Principali pianeti dell'Intelligenza Artificiale

IA Simbolica => Utilizza regole logiche e rappresentazioni esplicite per risolvere problemi complessi.
IA Predittiva => Impiega dati storici e modelli per prevedere eventi e tendenze future.
IA Prescrittiva => Suggerisce azioni ottimali, valutando scenari e conseguenze, per guidare le decisioni strategiche.
IA Generativa => Crea nuovi contenuti come testi, immagini, suoni o dati, simulando il processo creativo umano.
IA Agentica (evoluzione della IA Generativa) => Sfrutta i modelli linguistici per ragionare, pianificare ed eseguire compiti complessi in autonomia o con un minimo intervento umano.
IA Causale => Identifica e modella relazioni causa-effetto per andare oltre le semplici correlazioni statistiche.
IA Spiegabile => Fornisce motivazioni trasparenti e interpretabili per i risultati generati dai modelli di apprendimento automatico.
IA con Apprendimento Continuo => E' focalizzata sull'apprendimento continuo, adattandosi a contesti e dati in costante mutamento.
IA Decisionale => Raccomanda o esegue azioni autonome sulla base di obiettivi e vincoli definiti.
IA Robotica => Applica l'IA a sistemi fisici per percepire e manipolare il mondo reale.

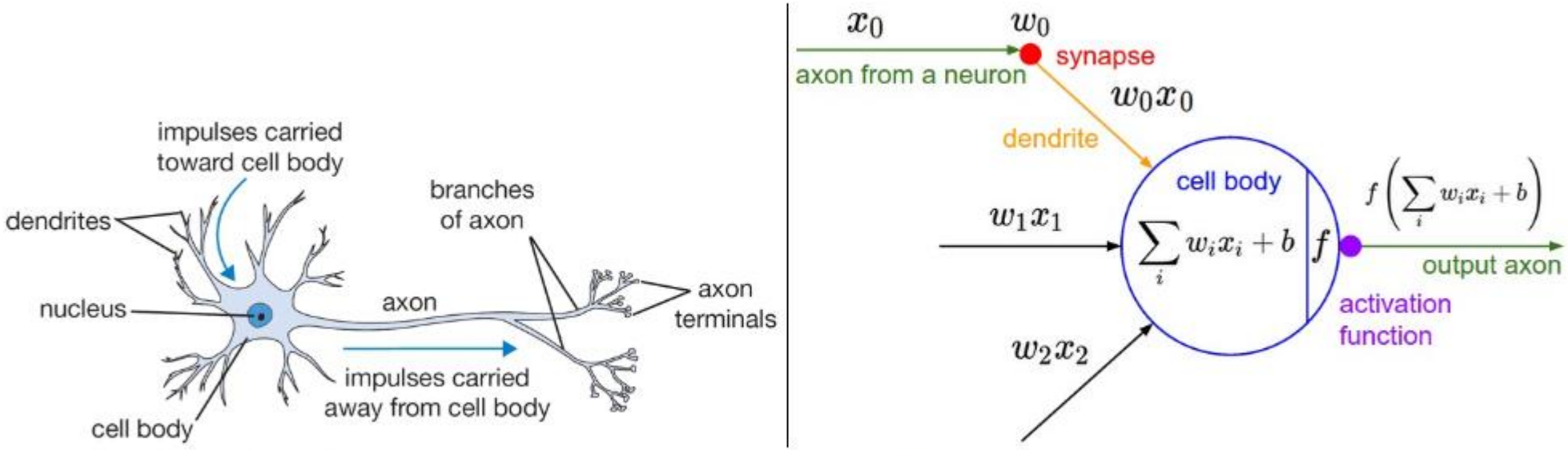
Evoluzione dell'Intelligenza Artificiale Generativa con l'avvento di ChatGPT

Il mattoncino fondamentale dei modelli generativi di apprendimento profondo: il ruolo delle reti neurali artificiali

Ispirate al cervello
Le unità di calcolo (neuroni) sono collegate in strati, come i neuroni biologici

Combinare e trasformare
Ogni neurone somma i segnali ricevuti e applica una trasformazione non lineare

Come i Lego
Si assemblano strati di neuroni per costruire sistemi sempre più complessi



A cartoon drawing of a biological neuron (left) and its mathematical model (right).

fonte

Il mattoncino fondamentale dei modelli generativi di apprendimento profondo: il ruolo delle reti neurali artificiali

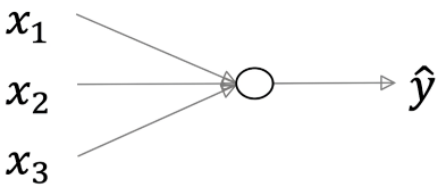
Neurone Artificiale

$$a_j^{(l)} = \sigma\left(\sum_i w_{ji}^{(l)}\, a_i^{(l-1)} + b_j^i\right)$$

Rete Neurale a strati (feedforward)

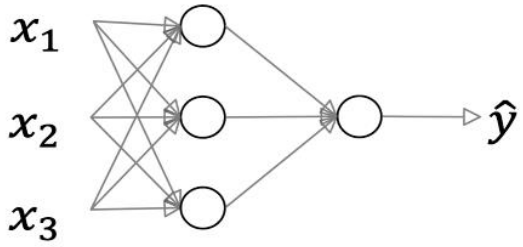
$$y = \sigma^{(L)}\left(W^{(L)}\sigma^{(L-1)}\left(\dots\sigma^{(1)}\left(W^{(1)}x + b^{(1)}\right)\dots\right) + b^{(L)}\right)$$

1 neurone



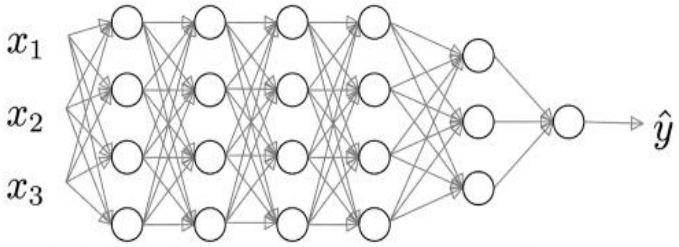
Il caso più semplice:
input -> trasformazione->output

1 strato interno



La rete inizia a «ragionare»:
estrae caratteristiche intermedie

5 strati interni

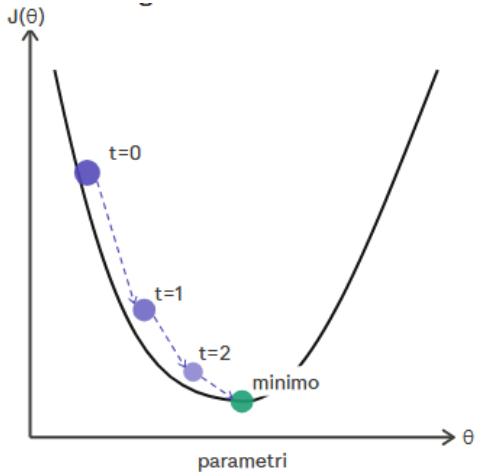


Rete profonda:
maggiore capacità di apprendimento

fonte

Il mattoncino fondamentale dei modelli generativi di apprendimento profondo: il ruolo delle reti neurali artificiali

$$\theta_{t+1} = \theta_t - \eta \frac{\partial}{\partial \theta_t} J(\theta);$$
$$\theta = (W, b)$$



Discesa del gradiente

Obiettivo
Trovare i parametri che minimizzano l'errore del modello sui dati

Meccanismo
Come scendere da una montagna con la nebbia: un passo alla volta nella direzione opposta alla pendenza (gradiente)

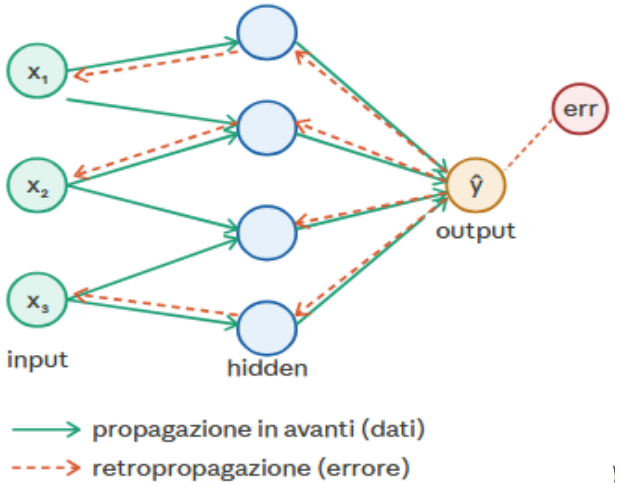
A cosa serve
Strategia di ottimizzazione: aggiorna i parametri utilizzando il gradiente per ridurre l'errore. Risponde alla domanda: «Come aggiorniamo i pesi?»

Retro-Propagazione

Obiettivo
Calcolare il contributo di ogni parametro all'errore finale del modello

Meccanismo
L'errore viene propagato all'indietro strato per strato: ogni peso riceve il gradiente che guida il suo aggiornamento

A cosa serve
L'algoritmo calcola in modo efficiente il gradiente per ciascun peso, propagando l'errore all'indietro. Risponde alla domanda: «Come calcoliamo il gradiente?»



I modelli linguistici e l'elaborazione del linguaggio naturale (NLP)

L'NLP è un'area che combina le lingue, l'informatica e l'intelligenza artificiale per studiare le interazioni tra computer e linguaggio umano. L'obiettivo dell'NLP è progettare modelli che permettano ai computer di comprendere il linguaggio naturale al fine di eseguire determinate attività.

I modelli linguistici possono essere definiti come una distribuzione di probabilità su sequenze di parole.

$$p(x_1, \dots, x_L)$$

Dato un vocabolario di parole, un modello linguistico assegna una probabilità ad ogni sequenza, con l'obiettivo di prevedere la parola successiva. Un semplice esempio di modello può essere tratto dai processi di Markov. Dall'introduzione del Transformer da parte di Google nel 2017, di BERT da Google nel 2018 e di GPT da OpenAI nel 2018, si parla sempre più dei modelli linguistici di grandi dimensioni (Large Language Models - LLMs).

$$p(x_{1:L}) = p(x_1)p(x_2|x_1)p(x_3|x_2, x_1) \dots p(x_L|X_{1:L-1}) = \prod_{i=1}^{L} p(x_i|x_{i:i-1})$$

LMSYS Chatbot Arena Leaderboard

Questa classifica i modelli linguistici in base alle prestazioni ottenute sui compiti testuali (text-to-text).

Chatbot Arena - a Hugging Face Space by lmarena-ai

Rank	Rank Spread	Model	Score	Votes	Price $/M	Context
1	1 – 4	claude-opus-4-7-thinking Anthropic · Proprietary	1503 ±6	8945	$5 / $25	1M
2	1 – 3	claude-opus-4-6-thinking Anthropic · Proprietary	1502 ±5	23.616	$5 / $25	1M
3	1 – 6	claude-opus-4-6 Anthropic · Proprietary	1498 ±5	25.089	$5 / $25	1M
4	3 – 8	gemini-3.1-pro-preview Google · Proprietary	1492 ±4	29.468	$2 / $12	1M
5	2 – 8	claude-opus-4-7 Anthropic · Proprietary	1491 ±6	9614	$5 / $25	1M
6	3 – 9	muse-spark Meta · Proprietary	1490 ±6 Preliminary	10.491	N/A	N/A
7	4 – 14	gemini-3-pro Google · Proprietary	1486 ±4	41.381	$2 / $12	1M
8	4 – 18	gpt-5.5-high OpenAI · Proprietary	1484 ±7	6488	$5 / $30	1.1M
9	6 – 19	grok-4.20-beta1 xAI · Proprietary	1480 ±5	18.791	N/A	N/A
10	8 – 19	gpt-5.2-chat-latest-20260210 OpenAI · Proprietary	1477 ±5	23.717	$1.75 / $14	128K

Il Motore dell'Innovazione: costruire i modelli linguistici del futuro con il Trasformatore

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

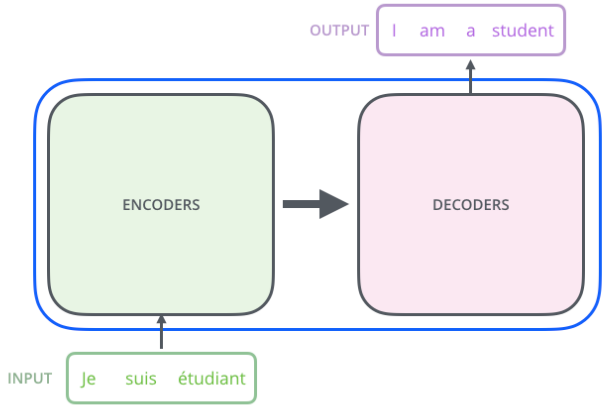
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

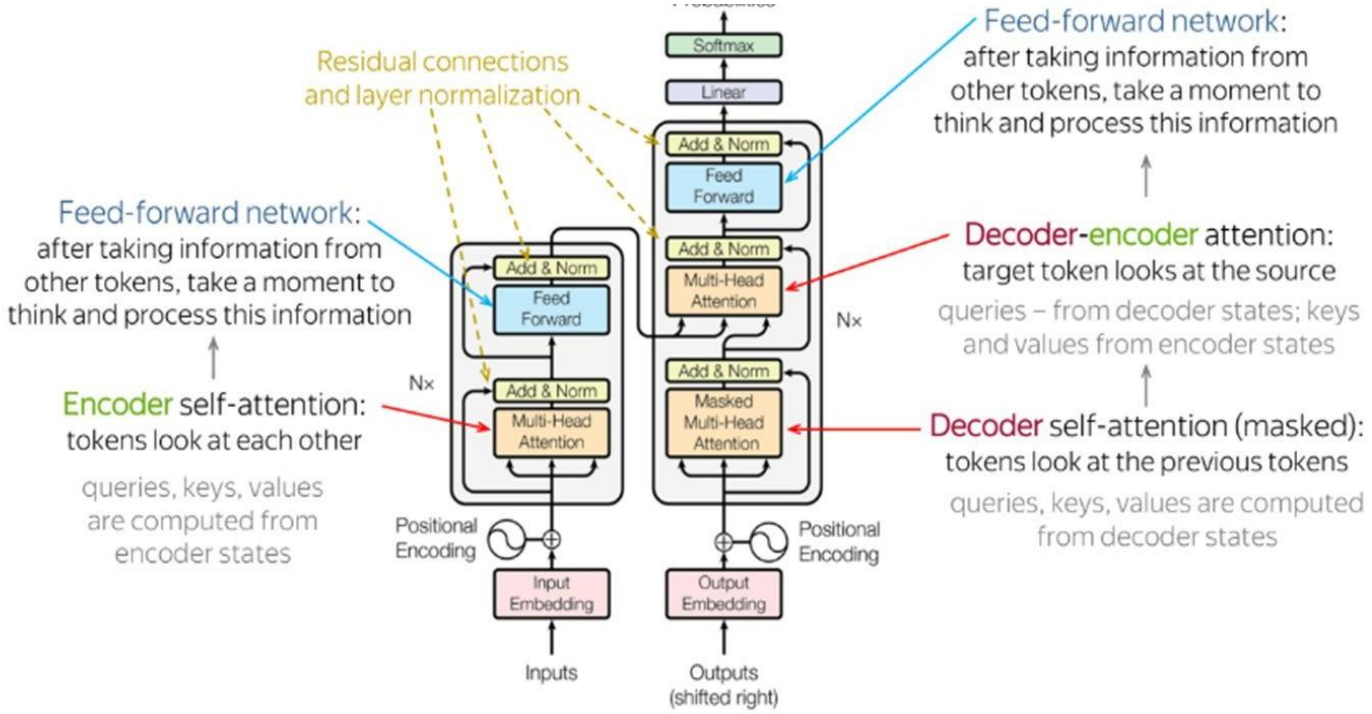
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to



fonte



fonte

Funzionamento del trasformatore

Passaggio 1: Convertire le parole in numeri (Token IDS) -> Suddivisione del testo in singole unità (codifica del testo)

E' il processo che consiste nello spezzare un testo in parti più piccole, chiamate token (unità). Queste unità possono essere parole, segni di punteggiatura o anche segmenti di parole, a seconda del metodo utilizzato.

I love machine learning

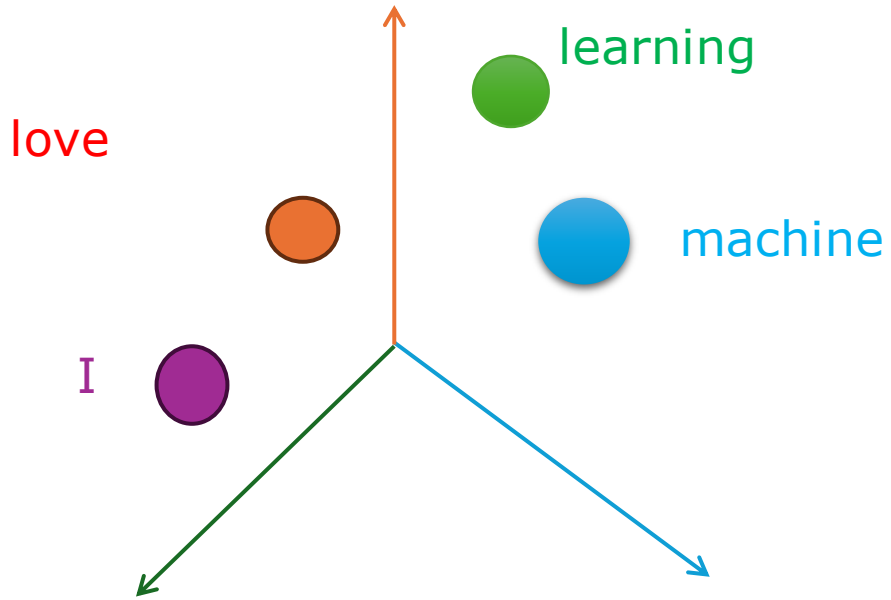
Dizionario delle unità:

«I»: 245
«love»: 1876
«machine»: 3421
«learning»: 2109

Funzionamento del trasformatore

Passaggio 2: Convertire gli ID delle unità (token IDS) in vettori -> Rappresentazione vettoriale delle unità

Le rappresentazioni vettoriali delle unità esistono in uno spazio ad alta dimensione, in cui ogni unità viene mappato in una posizione unica, chiamata vettore.
Le rappresentazioni vettoriali sono progettate in modo tale che elementi semanticamente simili abbiano vettori vicini nello spazio, facilitando così l'elaborazione del linguaggio naturale e l'applicazione dei modelli di apprendimento automatico.



Coordinate vettoriali:
«I» (245):[0.2,0.5,0.7]
«love» (1876): [0.3,0.6,0.4]
«machine» (3421): [0.7,0.5,0.2]
«learning» (2109): [0.5,0.2,0.3]

Funzionamento del trasformatore

Passaggio 3: Calcola la posizione di ciascuna unità nella sequenza del testo → Rappresentazione vettoriale del posizionamento

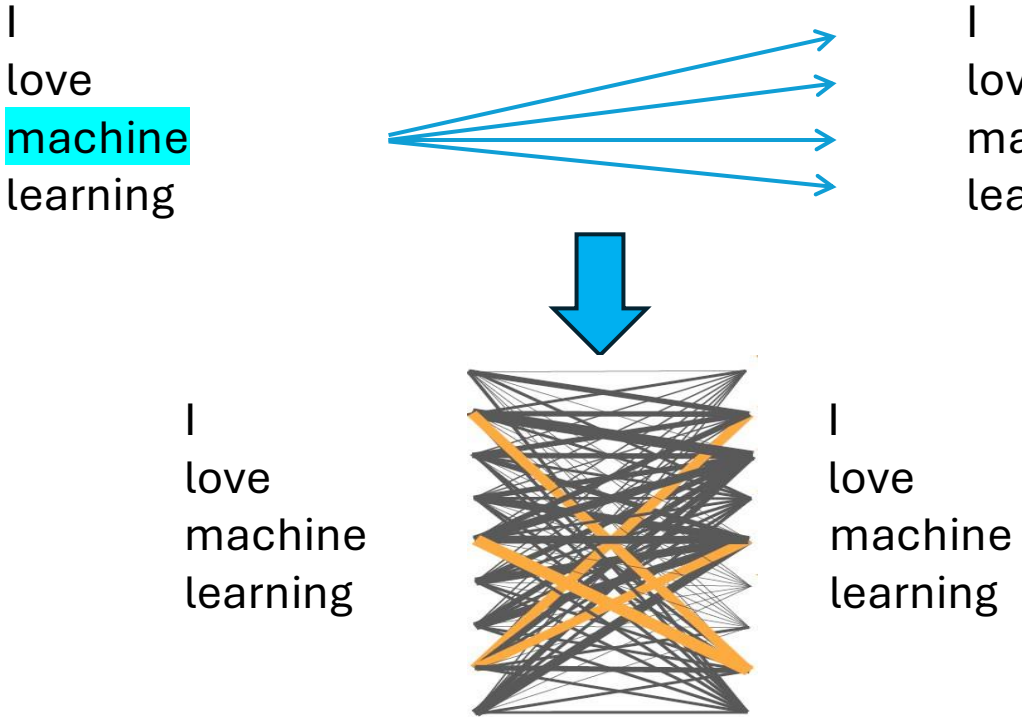
La rappresentazione vettoriale delle unità e del posizionamento sono aggiunte per preservare l'informazione sull'ordine delle unità nel testo

Word	token IDS	token embedding	positional embedding	final input embedding	
I	245	[0.2,0.5,0.7]	[0.0,0.0,0.0]	[0.2,0.5,0.7]	position:0
love	1876	[0.3,0.6,0.4]	[0.1,0.1,0.1]	[0.4,0.7,0.5]	position:1
machine	3421	[0.7,0.5,0.2]	[0.2,0.2,0.2]	[0.9,0.7,0.3]	position:2
learning	2109	[0.5,0.2,0.3]	[0.3,0.3,0.3]	[0.8,0.5,0.6]	position:3

Funzionamento del trasformatore

Passaggio 4: Attribuzione dei pesi a ogni parola rispetto a tutte le altre -> Attenzione

Ci sono molti strati di attenzione. L'intuizione è che ogni strato di attenzione imparerà un aspetto diverso del linguaggio → Multi-Attenzione



$$Attention\,(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_K}}\right)V$$

Funzionamento del trasformatore

Passaggio 5: Predire la parola successiva assegnando dei punteggi per ogni parola → Rete Neurale Feed-Forward

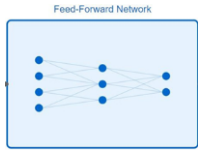
La rete neurale feed-forward prende in input tutti i pesi dal processo di attenzione e li elabora. Essa formula una previsione su quale possa essere la parola successiva in base ai dati di addestramento. Queste previsioni non sono ancora probabilità, ma sono punteggi, o "logits". Tutti questi punteggi costituiscono un "vettore di logits".

Passaggio 6: Convertire i punteggi in probabilità -> Funzione Softmax

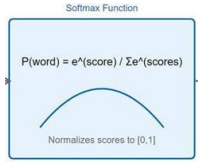
La parola con la probabilità più alta viene scelta come previsione

Funzionamento del trasformatore

- **First word generation**:
  - Attention weights
    245 [I]: 0.85
    1876 [love] 0.12
    3421 [machine] 0.02
    2109 [learning] 0.01



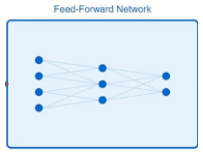
FFN output (logits)
4521 [Ich]: 9.2
3892 [Du]: 2.1
1847 [Wir]: 1.3



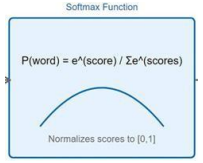
Softmax output -> Probabilities
4521 [Ich]: 87.3%
3892 [Du]: 8.2%
1847 [Wir]: 2.5%

Selected: 4521 [Ich]: 87.3%

- **Second word generation**:
  - Attention weights
    245 [I]: 0.15
    1876 [love] 0.78
    3421 [machine] 0.04
    2109 [learning] 0.03



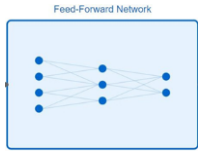
FFN output (logits)
2834 [liebe]: 8.7
5621 [mag]: 5.2
3109 [habe]: 3.1



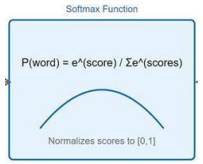
Softmax output -> Probabilities
2834 [liebe]: 76.4%
5621 [mag]: 18.3%
3109 [habe]: 5.3%

Selected: 2834 [liebe]: 76.4%

- **Third word generation**:
  - Attention weights
    245 [I]: 0.05
    1876 [love] 0.10
    3421 [machine] 0.72
    2109 [learning] 0.13



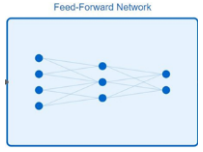
FFN output (logits)
7654 [maschinelles]: 9.1
2341 [Maschinen]: 6.8
8923 [Computer]: 4.2



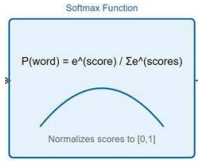
Softmax output -> Probabilities
7654 [maschinelles]: 68.9%
2341 [Maschinen]: 24.7%
8923 [Computer]: 6.4%

Selected: 7654 [maschinelles]: 68.9%

- **Fourth word generation**:
  - Attention weights
    245 [I]: 0.03
    1876 [love] 0.08
    3421 [machine] 0.25
    2109 [learning] 0.64



FFN output (logits)
4123 [Lernen]: 9.5
6789 [Studieren]: 4.1
5432 [Training]: 3.3



Softmax output -> Probabilities
4123 [Lernen]: 82.1%
6789 [Studieren]: 11.2%
5432 [Training]: 6.7%

Selected: 4123 [Lernen]: 82.1%

Funzionamento del trasformatore

Passaggio 7: Convertire i token IDS in parole -> decodifica del testo

-Token IDS

4521 2834 7654 4123

-Decodifica del testo

Ich liebe maschinelles Lernen

Traduzione:
Input: "I love machine learning“
Output: "Ich liebe maschinelles Lernen"

Architettura del Trasformatore

Oltre alla generazione di testo, le architetture dei Trasformatori possono essere classificate in tre tipologie: solo codifica, solo decodifica e codifica-decodifica. Ogni architettura è progettata per rispondere a esigenze specifiche.

Trasformatore di codifica: sfruttano un meccanismo di self-attention bidirezionale che permette di considerare il contesto circostante ad ogni parola (sinistra e destra). E' utilizzato per compiti in cui è richiesta la comprensione dell'intera sequenza di testo: classificazione, analisi delle opinioni, riconoscimento delle entità nominate (NER). Esempi di linguaggi: BERT, RoBERTa.

Trasformatore di decodifica: adottano un meccanismo di self-attention unidirezionale che permette di considerare solo il contesto che precede ogni parola (sinistra). E' utilizzato per la generazione sequenziale di testo, dove ogni parola è predetta in base a quelle precedenti: produzione di testo, assistenti virtuali, riassunti, traduzioni. Esempi di linguaggi: GPT, Llama.

Trasformatore di codifica-decodifica: integrano la capacità di entrambi gli approcci. E' utilizzato in compiti che richiedono sia la comprensione che la generazione di sequenze di testo: traduzioni, riassunti, Q&A. Esempi di linguaggi: T5, BART.

Trasformatori: punti di forza e punti di debolezza

Punti di forza:

- I Trasformatori eccellono nel catturare le dipendenze a lungo termine nei dati, grazie al loro meccanismo di auto-attenzione, rendendoli altamente efficaci per compiti come la generazione del linguaggio e la traduzione;
- I Trasformatori sono altamente parallelizzabili, permettendo l'addestramento su dataset di dimensioni massicce e superando le limitazioni del processamento sequenziale delle reti neurali ricorrenti.

Limitazioni:

- I Trasformatori spesso richiedono grandi dataset e rilevanti risorse computazionali per essere addestrati efficacemente, il che può rappresentare un ostacolo per progetti di ricerca di dimensioni ridotte o applicazioni con dati limitati;
- I Trasformatori possono incontrare difficoltà nel mantenere coerenza e consistenza in output molto lunghi, come testi estesi o dialoghi, senza ulteriori modifiche architetturali.

Da GPT-1 a GPT-4o

Model Name	Release Date (Month, Year)	Approximate Number of Parameters	Approximate Training Data Size	Key Training Data Composition	Context Window Size (Tokens)
GPT-1	June 2018	117 Million	~4.5 GB	BookCorpus	512
GPT-2	February 2019	1.5 Billion	~40 GB	WebText	1024
GPT-3	June 2020	175 Billion	~570 GB	Common Crawl (filtered), WebText2, Books1, Books2, Wikipedia	2048
GPT-3.5 Turbo	November 2022	~200-300 Billion	Proprietary	Proprietary, likely includes conversational and instruction data	Up to 16,385
GPT-4o	May 2024	~1.8 Trillion (Est. MoE)	Proprietary	Multimodal (Text, Audio, Vision), Web-scale data	128,000

Il Trasformatore può fare calcoli?

Leggi la ricerca di Anthropic

Il Transformer può eseguire calcoli aritmetici?

I Transformer, nella loro forma base, **non eseguono calcoli aritmetici in modo affidabile.** Ecco perché:

Cosa fanno davvero

I Transformer sono fondamentalmente modelli di *pattern matching* statistico. Quando sembrano "calcolare" 2 + 2 = 4, in realtà stanno recuperando dalla memoria il pattern visto durante il training, non eseguendo un'operazione aritmetica vera e propria.

Limiti concreti

Per operazioni semplici (addizioni a poche cifre, moltiplicazioni memorizzabili), i modelli funzionano bene perché il risultato compare spesso nei dati di training. Ma per operazioni più complesse — numeri grandi, divisioni lunghe, radici quadrate — l'accuratezza crolla drasticamente. Non c'è nessuna "unità aritmetica" interna; ogni risultato è un'inferenza probabilistica.

Perché questo accade

L'architettura Transformer elabora token in parallelo attraverso meccanismi di attenzione e reti feed-forward. Questi strati sono ottimizzati per modellare relazioni nel linguaggio, non per implementare algoritmi deterministici come quelli di un processore.

Come si ovvia al problema

I modelli moderni vengono spesso dotati di **strumenti esterni** (come un interprete Python o una calcolatrice) che vengono invocati quando serve un calcolo preciso. È ciò che fa anche Claude: quando eseguo codice per risolvere un'operazione matematica, uso uno strumento reale, non il ragionamento interno del modello.

Un'eccezione parziale: Chain-of-Thought

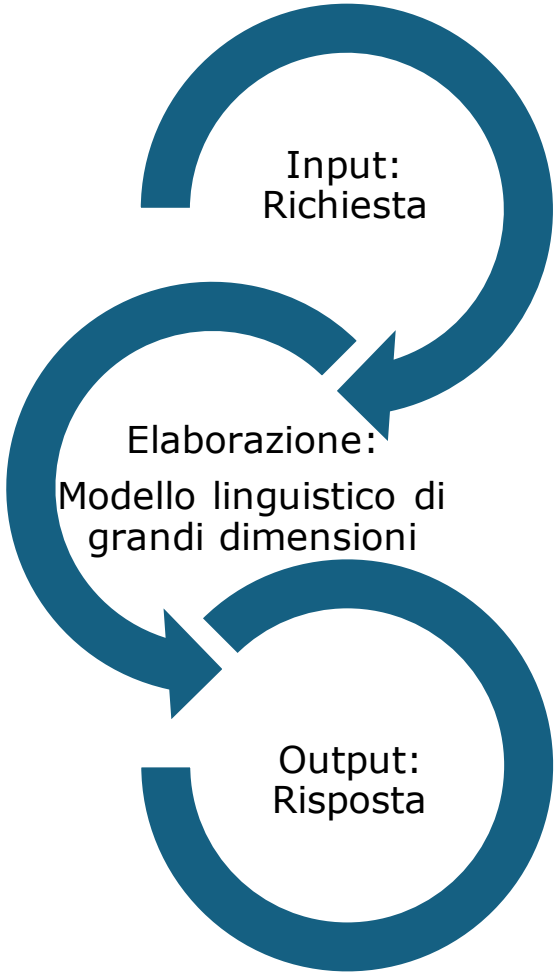
Chiedere al modello di ragionare passo per passo (*chain-of-thought prompting*) migliora sensibilmente le prestazioni aritmetiche, perché scompone il problema in sotto-passaggi più semplici, ciascuno dei quali ha maggiore probabilità di essere corretto. Ma rimane comunque una simulazione, non un calcolo garantito.

In sintesi: i Transformer *simulano* il calcolo, non lo *eseguono*. Per risultati affidabili serve sempre un tool esterno.

Risposta da Sonnet 4.6 (Claude)

Cos'è l'ingegneria delle istruzioni ? (Prompt Engineering)

Il processo di progettazione e creazione di istruzioni o domande chiare e specifiche per i modelli linguistici di grandi dimensioni (LLMs), finalizzato ad ottenere le risposte desiderate.



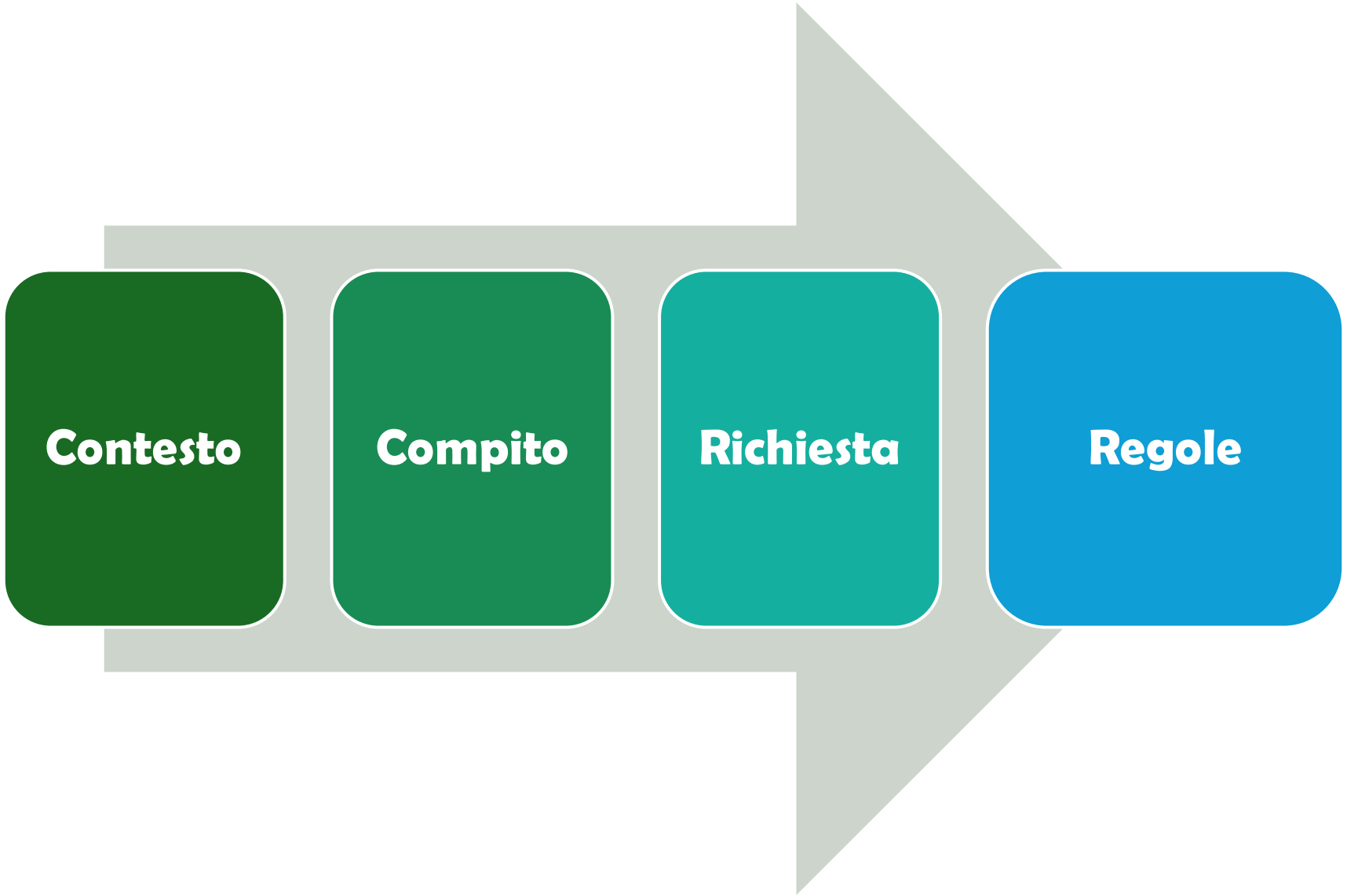
Richieste non efficaci

- Puoi scrivermi le condizioni di assicurazione di una polizza vita?

Richieste efficaci

- Puoi scrivermi le condizioni di assicurazione di una polizza vita, temporanea caso morte a premio annuo e capitale costante, con tasso tecnico 1%, tavola Istat 2014 (maschi 60%, femmine 40%), durata 10 anni, capitale assicurato pari a 100.000 € e caricamenti del 15% …..?

Flusso per una istruzione efficace



Flusso per una istruzione efficace nel dettaglio

Contesto: stabilisce l'ambientazione dell'interazione con il modello linguistico. Include tutti i dettagli di sfondo, le impostazioni o i parametri che definiscono l'ambiente circostante al compito. Il contesto aiuta il modello a comprendere la rilevanza e il quadro del compito in esame.

Compito: rappresenta l'obiettivo che si desidera raggiungere con il modello linguistico. Può essere qualsiasi cosa, dalla generazione di testo, alla risposta a una domanda, dalla traduzione di lingue, alla creazione di codice.

Richiesta: rappresenta il testo di input fornito al modello linguistico. È formulato per guidare il modello verso l'output desiderato.

Regole: sono le direttive fornite al modello che spiegano cosa ci si aspetta come risultato. Sono progettate per essere chiare e specifiche, così da ridurre al minimo l'ambiguità nelle risposte del modello.

In breve, il contesto stabilisce l'ambientazione, il compito definisce cosa deve essere fatto, la richiesta è il testo di input effettivo che include il compito ed eventualmente anche il contesto, mentre le regole sono i passaggi dettagliati all'interno della richiesta che indicano al modello come portare a termine il compito.

Generazione Aumentata dal Recupero di Informazioni (RAG)

Cos'è la RAG (Retrieval Augmented Generation)?

- Introdotto da Facebook AI Research (FAIR) nel 2020, unisce la potenza dei modelli linguistici di grandi dimensioni (LLMs) a fonti di conoscenza esterne, garantendo risposte fluide e aggiornate.
- La generazione di testo aumentata dal recupero di informazioni esterne (RAG) è un approccio progettato per migliorare le capacità dei modelli linguistici di grandi dimensioni (LLMs) integrando conoscenze esterne aggiornate. Combina un meccanismo di recupero (retrieval) con un modello generativo (LLM). Il modello linguistico è come un narratore, mentre il meccanismo di recupero funge da bibliotecario.

Perché la RAG è importante?

I modelli linguistici di grandi dimensioni (LLMs) tradizionali generano risposte esclusivamente in base ai dati su cui sono stati addestrati, il che può comportare diversi problemi.

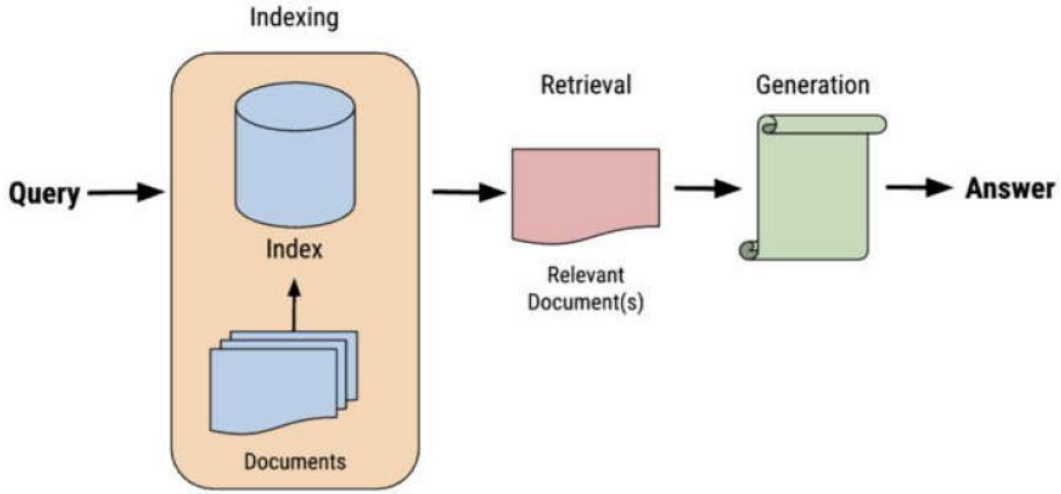
- **Obsolescenza dei dati** -> I modelli linguistici di grandi dimensioni (LLMs) hanno limiti temporali nella loro conoscenza e potrebbero non essere aggiornati sulle ultime informazioni, a meno che non vengano riaddestrati con regolarità. Il riaddestramento è un processo costoso e richiede un uso intensivo di risorse.
- **Allucinazioni** -> Quando un modello linguistico di grandi dimensioni (LLM) non dispone di informazioni di base sufficienti, rischia di generare risposte che sembrano convincenti ma risultano inventate o non correlate al mondo reale. Le allucinazioni possono trarre in inganno l'utente e influire in modo significativo sulla fiducia nei sistemi di intelligenza artificiale.
- **Incoerenza** -> Le risposte possono essere incoerenti con la conoscenza specialistica o non basate su dati del mondo reale.

Architettura Principale di un sistema RAG

- **Indicizzazione (Indexing):** la fase iniziale in cui i dati vengono elaborati.
- **Recupero (Retrieval):** la fase intermedia responsabile dell'identificazione e dell'estrazione delle informazioni più pertinenti in base alla domanda dell'utente.
- **Generazione (Generation):** la fase finale in cui un modello linguistico di grandi dimensioni (LLM) utilizza il contenuto recuperato per elaborare e generare una risposta coerente alla domanda dell'utente.

Oltre a queste fasi fondamentali, i sistemi RAG di successo integrano anche aspetti pratici di sviluppo:

- **Prompting**: la creazione di istruzioni appropriate per l'LLM.
- **Definizione del modello**: selezione e configurazione di un LLM specifico che fungerà da "cervello" dell'applicazione RAG, gestendo la generazione delle risposte.
- **Interfaccia Utente (UI)**: il punto di interazione principale tra l'utente e il sistema RAG.
- **Valutazione**: un processo continuo e cruciale per esaminare e migliorare le prestazioni del sistema RAG.

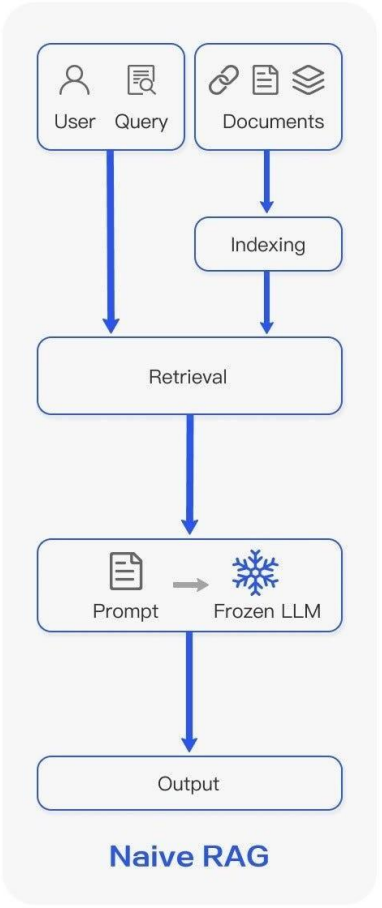


fonte

Architetture di un sistema RAG

Iniziamo con il Naive RAG

- **Indicizzazione (Indexing):** i dati vengono caricati, suddivisi in porzioni (*chunks*) più piccoli, quindi memorizzati e indicizzati tramite un archivio vettoriale (*vector store)* e una mappatura vettoriale (*embedding)*.
- **Recupero e Generazione (Retrieval and Generation):** i segmenti di testo pertinenti vengono estratti dalla memoria tramite uno strumento di recupero (*retriever)*. Questo processo si basa sulla somiglianza semantica tra la domanda e le porzioni di testo.
- Un LLM produce una risposta in base a istruzioni (*prompt)* che includono la domanda e i dati recuperati.



fonte

Architetture di un sistema RAG

...ci muoviamo verso il sistema RAG avanzato

Pre-Recupero (Pre-Retrieval)

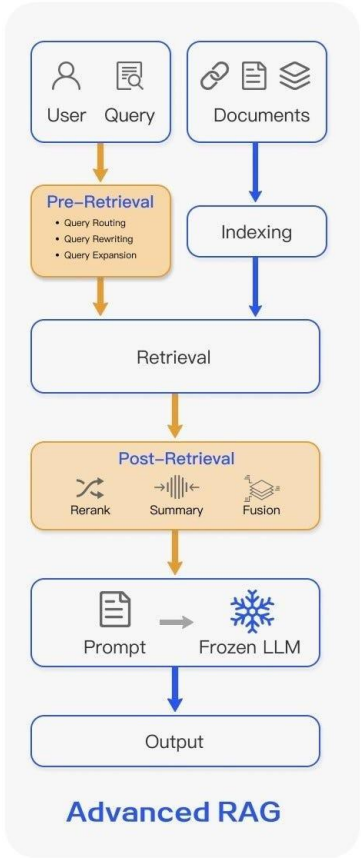
- **Analisi dei PDF (PDF Parsing):** fondamentale per estrarre con precisione informazioni da documenti non strutturati, inclusi testi, tabelle e immagini.
- **Suddivisione Semantica (Semantic Chunking):** divide il testo in base al contenuto e al contesto per preservarne il significato, andando oltre la semplice suddivisione basata su regole.
- **Riformulazione della domanda (Query Rewriting):** trasforma la query originale per allinearla meglio alla semantica del documento e correggere eventuali formulazioni imprecise.

Recupero (Retrieval)

- **Ricerca Ibrida / Insieme di Recuperatori (Ensemble Retriever/Hybrid Search):** combina molteplici tecniche o fonti di recupero per migliorare la pertinenza e la completezza dei risultati di ricerca.

Post-Recupero (Post-Retrieval)

- **Riassegnazione del Recupero (Re-Ranking):** riordina e filtra i documenti recuperati per posizionare i più pertinenti in prima posizione.
- **RAG Correttivo (Corrective RAG):** identifica e corregge imprecisioni o incongruenze nelle risposte generate incrociando le informazioni recuperate.
- **Fusione di RAG (RAG Fusion):** genera molteplici query derivate dalla query iniziale, recupera tutti i documenti, li riordina e unisce i risultati per la generazione.



fonte

RAG vs LLM vs Fine-Tuning

- **LLM**: si basa esclusivamente su dati preaddestrati.
- **Affinamento (Fine-tuning)**: modifica permanente dei pesi del modello su nuovi dati.
- **RAG**: recupera dinamicamente informazioni pertinenti senza necessità di riaddestramento.

Il RAG è l'ideale per fornire risposte in tempo reale, basate su dati reali e su specifiche pertinenti a un determinato settore.

Vantaggi & Sfide di un'architettura RAG

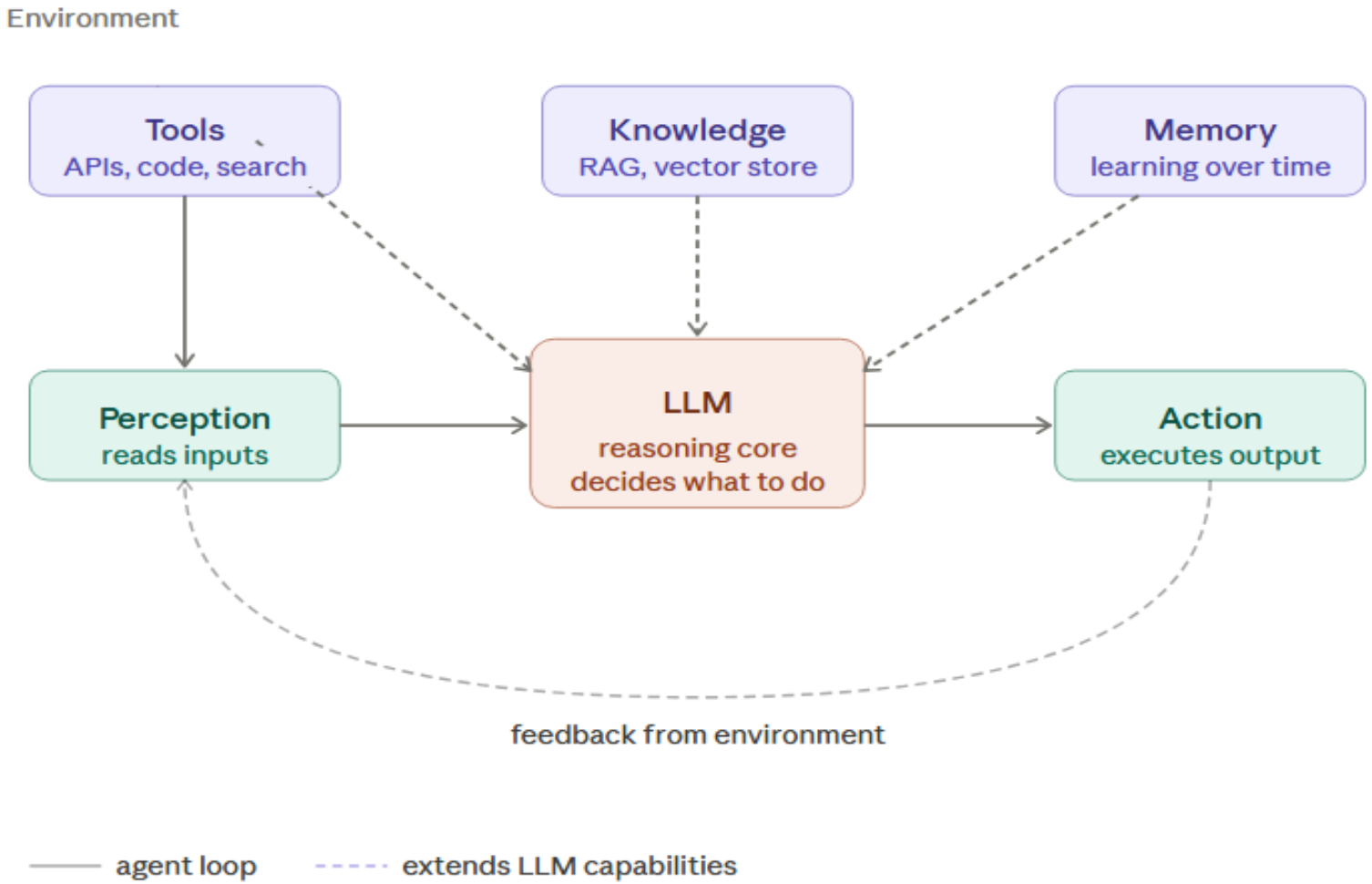
Vantaggi

- Migliore accuratezza e pertinenza dei risultati.
- Personalizzazione per casi d'uso specifici di un settore.
- Flessibilità: funziona con dati strutturati e non strutturati.
- Espande la conoscenza del modello oltre i suoi dati di addestramento.
- Minimizza la generazione di contenuti errati grazie al riscontro con dati reali.

Sfide

- Dipende dalla qualità e dalla pulizia dei dati.
- Comporta un carico computazionale maggiore e una complessità di sistema superiore.
- Richiede un'integrazione accurata tra i database e i modelli linguistici di grandi dimensioni (LLMs).
- Rischio di eccesso di informazioni o di recupero di dati non pertinenti.

Sistemi di IA ad Agenti



Una nuova era in cui i sistemi di intelligenza artificiale non si limitano a comprendere il linguaggio, ma ragionano, pianificano e interagiscono con il contesto circostante per svolgere compiti complessi.

claude.ai

Ragionamento

Collaborazione

Perché usare i sistemi di IA ad Agenti?

-> Svolgere i compiti in autonomia
-> Flessibilità e apprendimento
-> Aumento di produttività ed efficienza
-> Scalabilità e personalizzazione

Azione

Osservazione

Auto-Perfezionamento

Pianificazione

Architetture dei sistemi di IA ad Agenti

Singolo Agente: un unico agente di IA gestisce tutte le attività. Semplice da progettare e mantenere, ma meno scalabile per compiti complessi.

Sistema Multi-Agente (MAS): più agenti collaborano tra loro, abilitando l'elaborazione parallela e la specializzazione. Offrono maggiore scalabilità e robustezza, ma introducono sfide di coordinamento.

https://deepai.org/machine-learning-model/text2img / Google nano banana

Opportunità attuariali

LLMs => generazione di codice e di dati sintetici.

RAG => consultazione della documentazione tecnica (note tecniche), delle condizioni contrattuali e delle normative vigenti.

RAG con capacità di Agente => Audit/Debug del modello di Pricing: l'agente estrae le ipotesi da una nota tecnica e, autonomamente, scrive e lancia uno script in Python per verificare se il premio calcolato dal motore in produzione rispecchia le formule depositate.

Sistemi Multi-Agente (MAS) => R&S di prodotto: un agente è specializzato nell'ideazione del prodotto, un altro effettua le verifiche normative, un agente è dedicato allo sviluppo del modello di pricing, un agente redige la nota tecnica e, infine, un agente orchestratore gestisce l'intero flusso operativo.

Applicazione LLM: generare dati sintetici

Richiesta a un LLM di generare dati sintetici. Forniti due prompt su due dataset auto (train set), per i quali sono state fornite le caratteristiche e alcuni esempi di record.

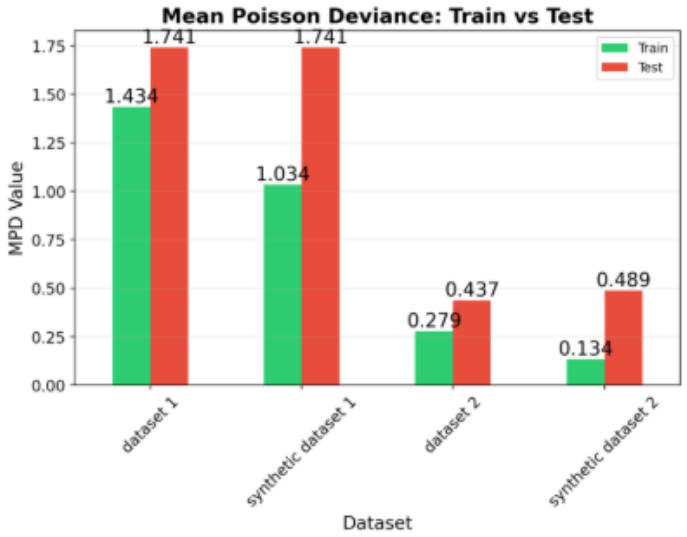
repository

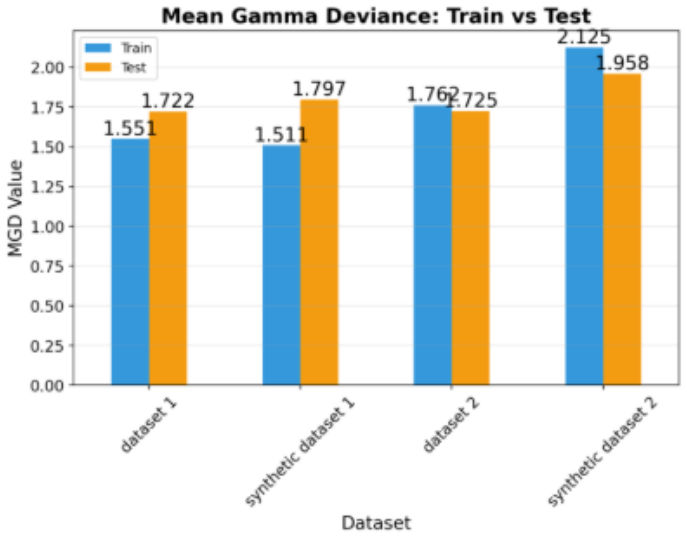
```
context: you are an expert actuarial data scientist with 20 Years of experience.
task: You are involved in a project to build synthetic datasets for the non-life insurance field.
request: please, generate 53320 rows of a synthetic dataset and save in a csv file with the following characteristics


I retrieved features from the dataset

1) columns and datatypes
Exposure    float64
VehValue    float64
VehAge      object
VehBody     object
Gender      object
DrivAge     object
ClaimOcc    int64
ClaimNb     int64
ClaimAmount float64

2) distribution of categorical columns

            count   unique  top         freq
VehAge      53320   4       old cars    15748
VehBody     53320   13      Sedan       17524
Gender      53320   2       Female      30331
DrivAge     53320   6       older work. people  12668


3) classes of categorical variables
Unique classes in 'VehAge': ['old cars' 'young cars' 'oldest cars' 'youngest cars']
Unique classes in 'VehBody': ['Hatchback' 'Utility' 'Station wagon' 'Hardtop' 'Panel van' 'Sedan'
 'Truck' 'Coupe' 'Minibus' 'Motorized caravan' 'Bus' 'Convertible'
 'Roadster']
Unique classes in 'Gender': ['Female' 'Male']
Unique classes in 'DrivAge': ['young people' 'older work. people' 'oldest people' 'working people'
 'old people' 'youngest people']

4) distribution of numerical variables

              count       mean        std     min     25%     50%     75%     max
Exposure    53320.0     0.466535    0.288448    0.002738    0.21629     0.443532    0.706366    0.999316
VehValue    53320.0     1.784108    1.217545    0.000000    1.01000     1.500000    2.160000    34.560000
ClaimOcc    53320.0     0.069449    0.254218    0.000000    0.00000     0.000000    0.000000    1.000000
ClaimNb     53320.0     0.074194    0.280946    0.000000    0.00000     0.000000    0.000000    4.000000
ClaimAmount 53320.0     136.092076  1004.945014 0.000000    0.00000     0.000000    0.000000        46868.1799
```

5) few examples of rows are here

	Exposure	VehValue	VehAge	VehBody	Gender	DrivAge	ClaimOcc	ClaimNb	ClaimAmount
0	0.588638	2.370	old cars	Station wagon	Female	working people	0	0	0.000000
1	0.539357	0.740	oldest cars	Sedan	Female	older work. people	0	0	0.000000
2	0.941821	2.110	young cars	Sedan	Male	older work. people	0	0	0.000000
3	0.087611	2.610	youngest cars	Station wagon	Female	young people	0	0	0.000000
4	0.999316	0.000	oldest cars	Bus	Male	oldest people	1	1	490.910000
5	0.095825	9.800	young cars	Convertible	Male	young people	0	0	0.000000
6	0.106776	1.440	old cars	Coupe	Female	working people	0	0	0.000000
7	0.624230	2.070	old cars	Hardtop	Female	older work. people	1	1	353.770000
8	0.396988	1.549	youngest cars	Hatchback	Female	youngest people	0	0	0.000000
9	0.114990	2.980	old cars	Minibus	Female	working people	0	0	0.000000
10	0.999316	17.000	youngest cars	Motorized caravan	Female	oldest people	0	0	0.000000
11	0.944559	3.230	young cars	Panel van	Male	youngest people	1	1	2639.439995
12	0.060233	3.670	youngest cars	Roadster	Male	young people	0	0	0.000000
13	0.561259	1.070	oldest cars	Sedan	Male	old people	0	0	0.000000
14	0.922656	4.470	youngest cars	Station wagon	Female	young people	1	1	353.770000
15	0.052019	2.520	old cars	Truck	Male	working people	0	0	0.000000
16	0.199863	2.100	old cars	Utility	Female	old people	0	0	0.000000
17	0.227242	0.500	oldest cars	Station wagon	Female	old people	0	0	0.000000
18	0.602327	1.360	youngest cars	Hatchback	Female	older work. people	0	0	0.000000
19	0.522930	0.390	oldest cars	Sedan	Female	oldest people	0	0	0.000000
20	0.509240	0.440	oldest cars	Minibus	Female	working people	0	0	0.000000
21	0.076660	3.520	young cars	Truck	Female	young people	0	0	0.000000
22	0.344969	1.490	young cars	Hatchback	Female	youngest people	0	0	0.000000
23	0.386037	1.350	youngest cars	Hatchback	Female	older work. people	0	0	0.000000
24	0.347707	2.880	young cars	Utility	Male	working people	0	0	0.000000
25	0.624230	3.090	old cars	Station wagon	Male	old people	1	1	866.820000
26	0.528405	1.030	old cars	Hatchback	Female	working people	1	1	4945.919998
27	0.720055	1.100	young cars	Hatchback	Female	working people	1	1	1604.730000
28	0.607803	2.120	youngest cars	Sedan	Female	working people	1	1	567.150000
29	0.772074	2.200	youngest cars	Hatchback	Female	young people	1	1	353.770000
30	0.626968	2.700	young cars	Station wagon	Male	working people	1	1	353.770000
31	0.635181	0.650	oldest cars	Sedan	Female	older work. people	1	1	200.000000
32	0.928131	3.770	young cars	Station wagon	Male	young people	1	1	520.730000
33	0.468172	1.790	young cars	Sedan	Male	old people	1	1	964.000000
34	0.490075	0.910	old cars	Hatchback	Female	working people	1	1	1266.709999
35	0.911704	2.050	oldest cars	Station wagon	Male	old people	1	3	6510.209987
36	0.520192	0.710	oldest cars	Hatchback	Female	young people	1	3	2453.269997
37	0.668036	1.330	young cars	Truck	Male	young people	1	3	3097.859996
38	0.731006	1.300	youngest cars	Hatchback	Female	young people	1	3	1884.479999
39	0.657084	0.740	oldest cars	Sedan	Female	working people	1	3	2351.829998
40	0.980151	1.210	old cars	Sedan	Male	oldest people	1	3	8990.809998
41	0.703628	0.720	oldest cars	Hatchback	Male	youngest people	1	3	2677.663635
42	0.887064	1.000	oldest cars	Utility	Male	older work. people	1	3	4054.109993
43	0.613279	1.800	young cars	Sedan	Male	oldest people	1	3	4076.819997
44	0.914442	2.950	old cars	Station wagon	Female	old people	1	4	2356.409999

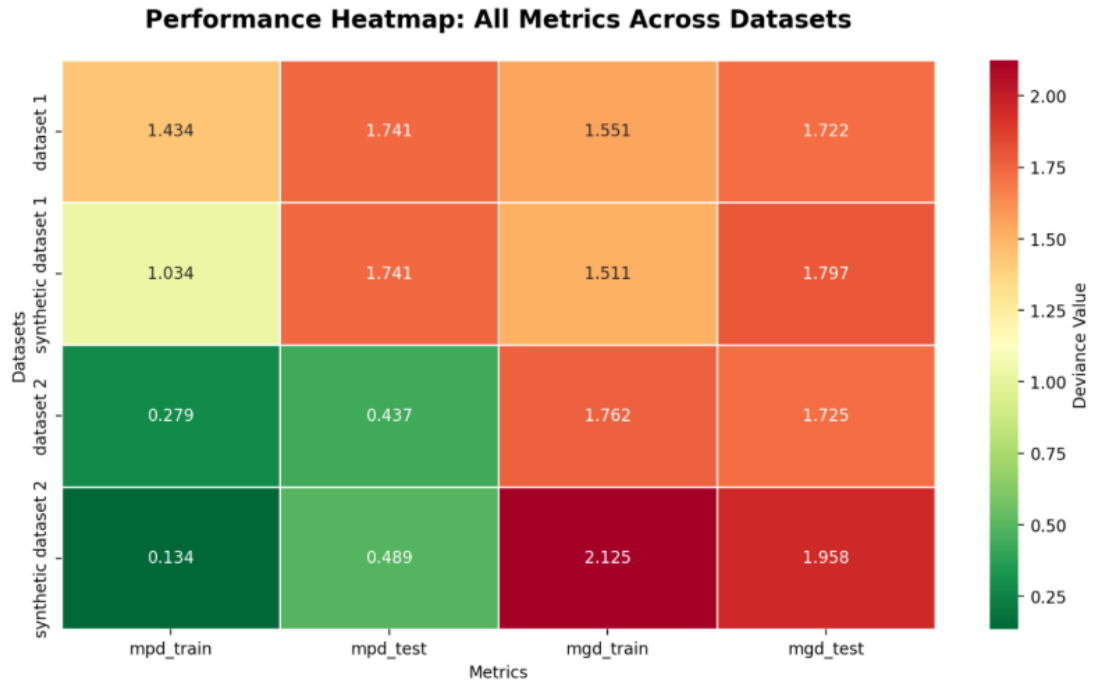
Applicazione LLM: generare dati sintetici





Confronto delle prestazioni tra dataset originari e dataset sintetici su train e test set per la frequenza dei sinistri e la stima del danno.

[web_app_results]

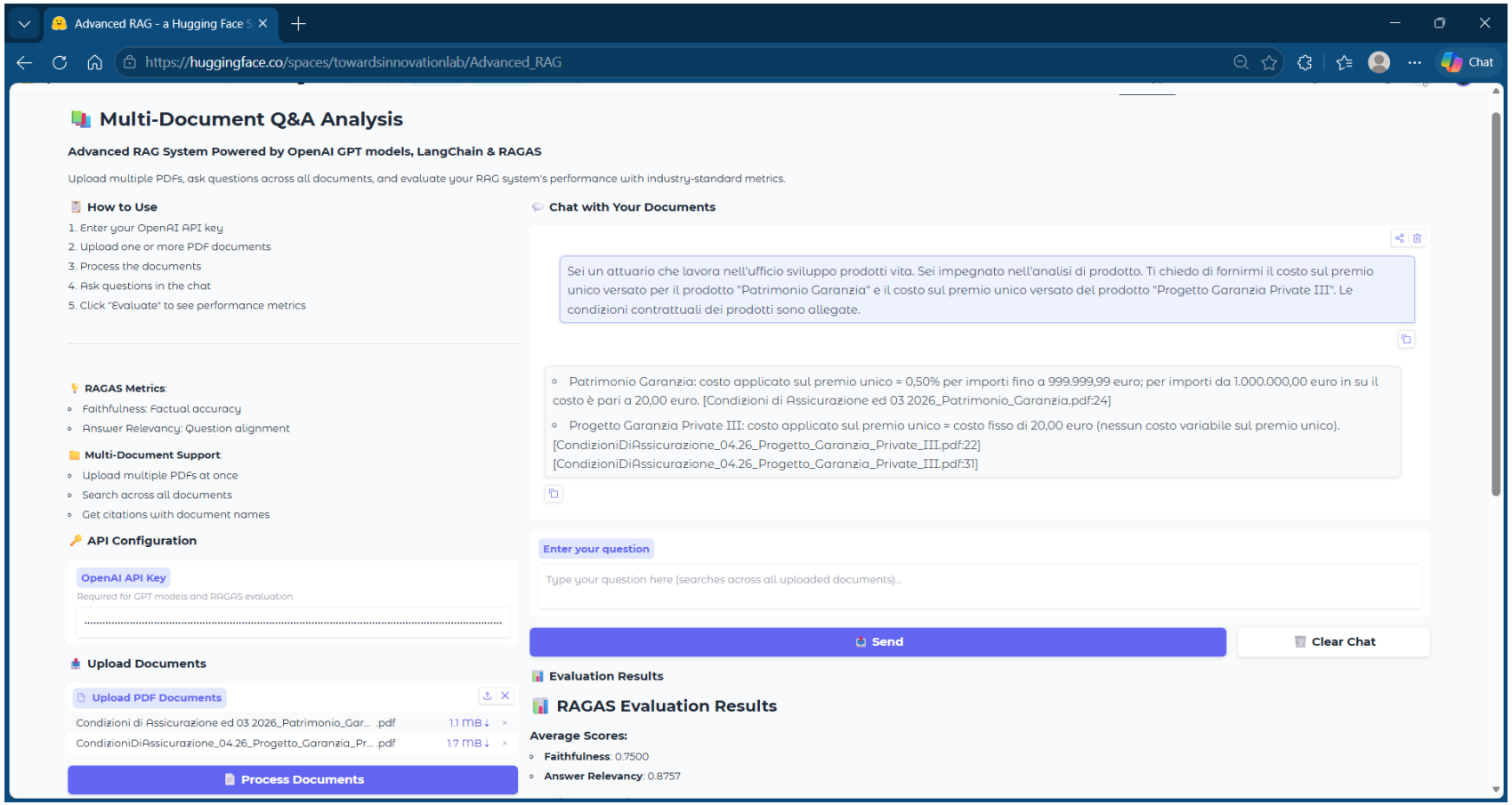


Applicazione RAG: consultazione di condizioni contrattuali

Condizioni contrattuali Patrimonio Garanzia

Condizioni contrattuali Progetto Garanzia Private III

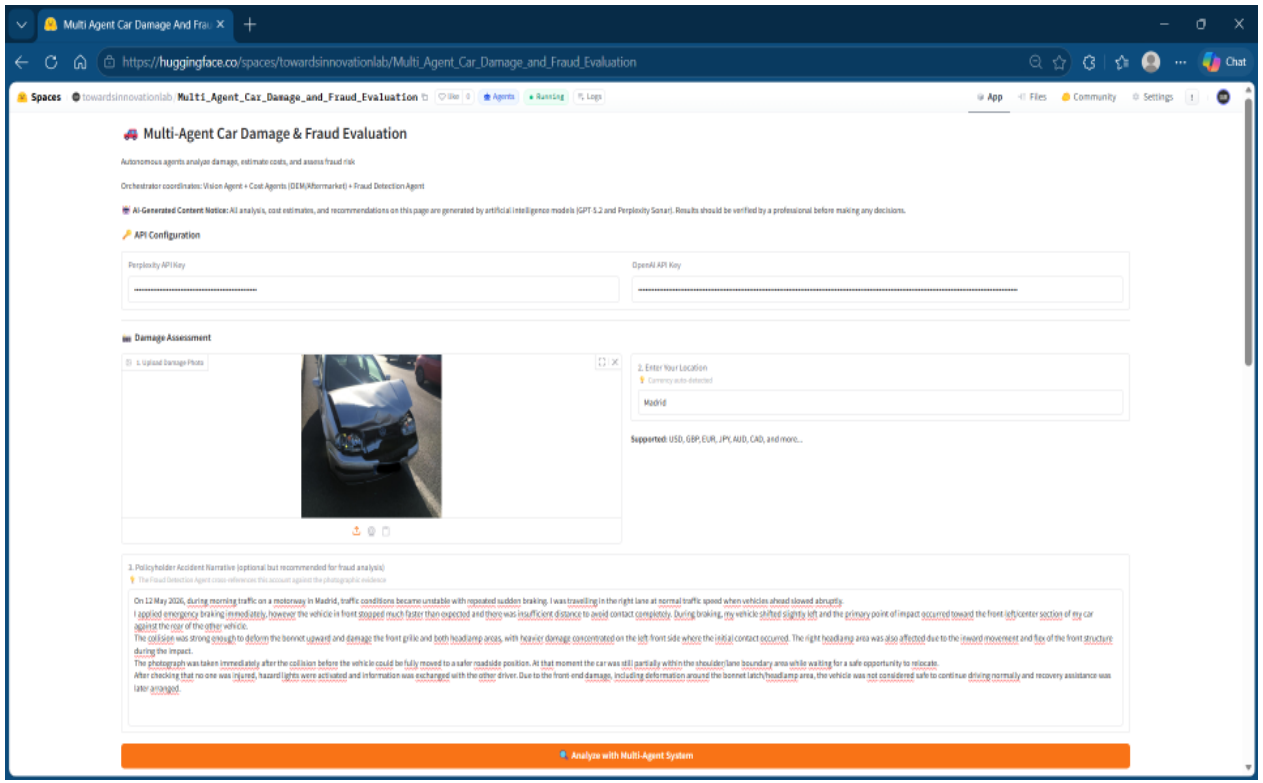
RAG tool

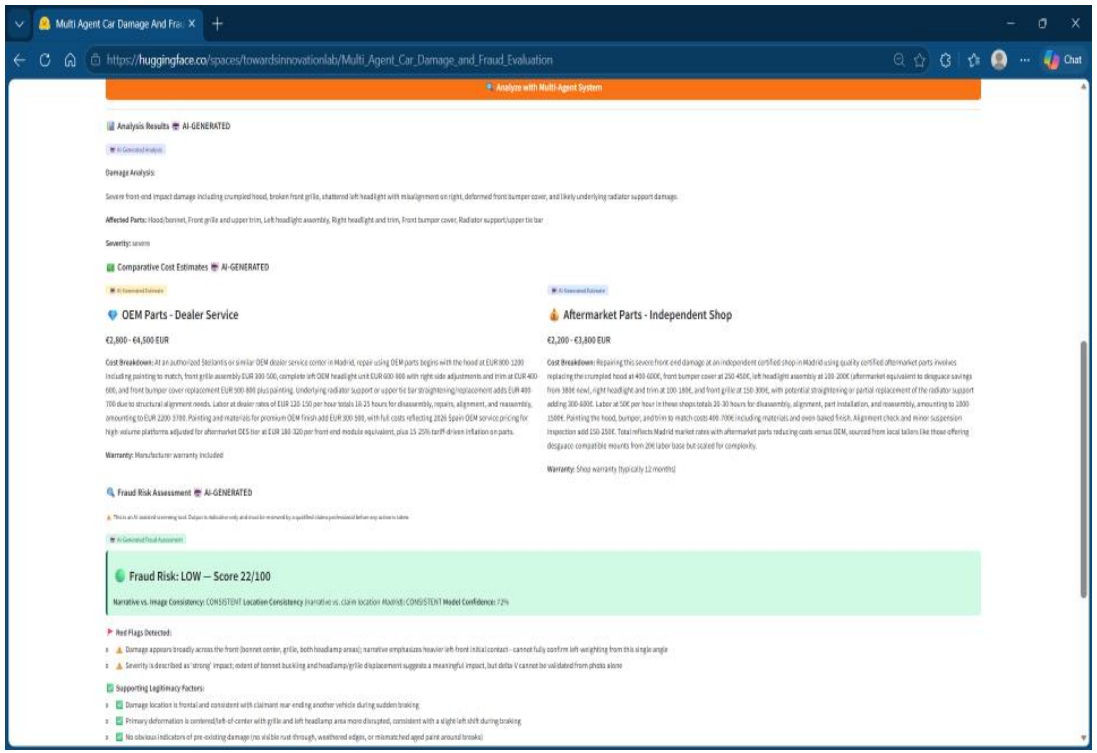


Applicazione Sistema Multi-Agente: valutazione danno auto e rischio frode

CAR_Damage_evaluation

Rischio frode: basso

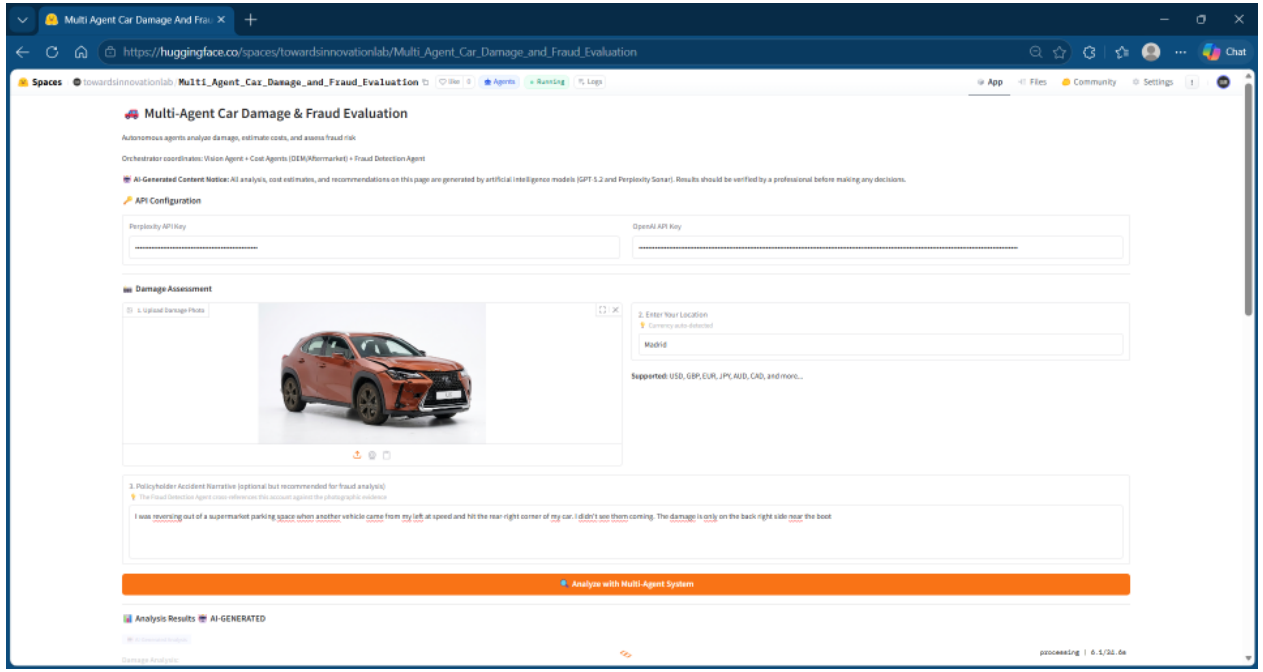


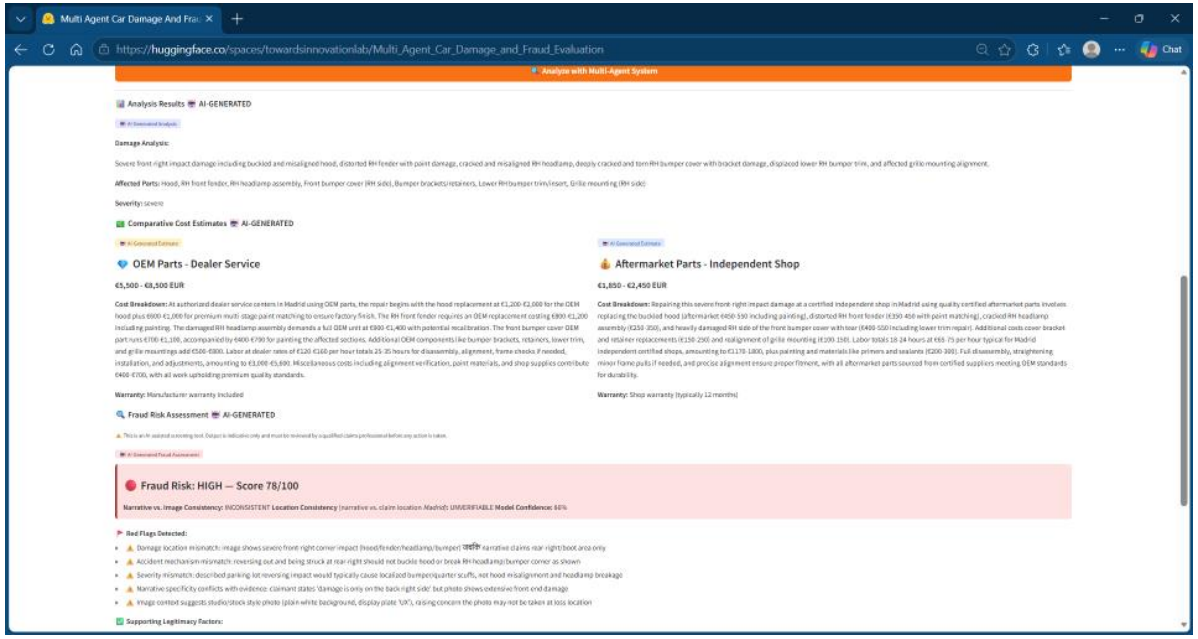


Applicazione Sistema Multi-Agente: valutazione danno auto e rischio frode

[CAR_Damage_evaluation]

Rischio frode: alto





Evoluzione dell’Intelligenza Artificiale Predittiva

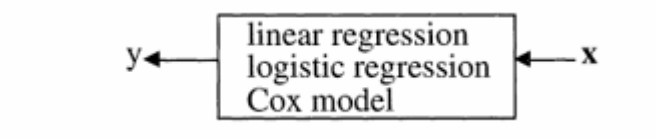
Statistical Modeling: The Two Cultures

Apprendimento Statistico (Statistical Learning)

-Nasce come sottoinsieme nel campo della Statistica

-Focalizzato sui modelli e la loro interpretabilità, sulla precisione e l'incertezza

-Validazione tramite bontà di adattamento

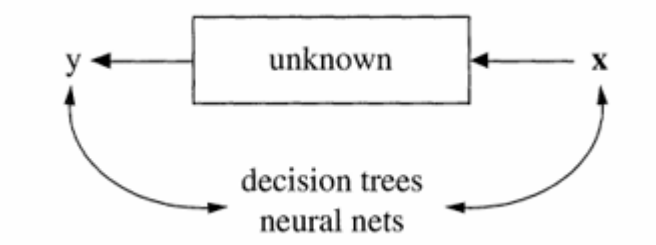


Apprendimento Automatico (Machine Learning)

-Nasce come sottoinsieme nel campo dell'Intelligenza Artificiale

-Focalizzato nelle applicazioni su larga scala e capacità predittiva

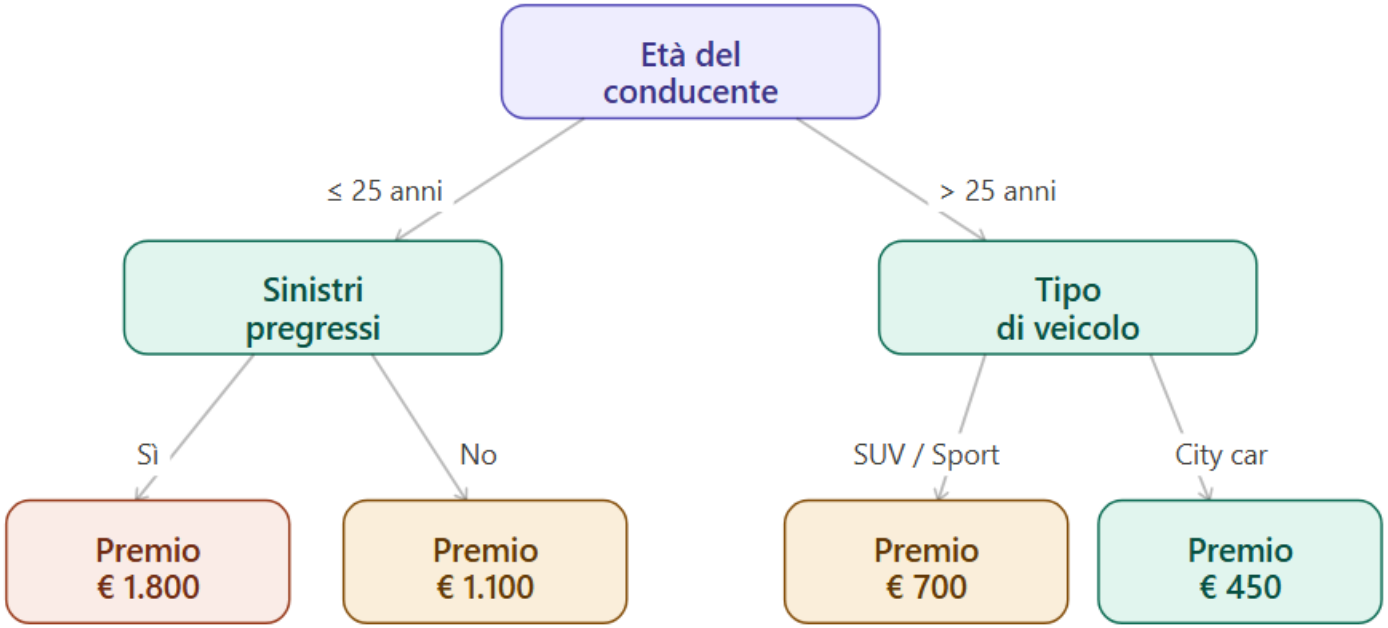
-Validazione tramite accuratezza predittiva



fonte

C'è molta sovrapposizione tra le due culture e la distinzione è diventata sempre più sfumata

Gli Alberi di Regressione



$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2$$

Come funziona?

① Segmenta il portafoglio in gruppi omogenei usando variabili predittive → ② In ogni foglia, prevede con la media dei premi osservati in quel gruppo → ③ I tagli sono scelti per minimizzare l'errore quadratico sui dati

Il risultato è una regola semplice e interpretabile: ogni ramo è una domanda, ogni foglia è una stima.
Vantaggio principale: trasparenza — l'attuario può spiegare ogni decisione.

✓ Interpretabile e trasparente
✓ Gestisce interazioni automaticamente

✗ Alta varianza (instabile)
✗ Può fare overfitting

claude.ai

Il potenziamento del gradiente (Gradient Boosting Machine (GBM))

Obiettivo: combinare tanti modelli «deboli» per ottenere uno «forte»
Tanti alberi di regressione che si correggono in sequenza:
(tasso di apprendimento=0.1 e sinistro reale = €800)

$$F_M(x) = F_0(x) + \eta \sum_{m=1}^{M} h_m(x)$$

Step 1

1° Albero
Stima iniziale= €700

Residuo = €100

Step 2

2° Albero
Studia i residui del primo albero
-Previsione del residuo=€100
-Nuova stima totale= €710
-Residuo = €90

Step 3

3° Albero
Studia i residui del secondo albero
-Previsione del residuo=€90
-Nuova stima totale= €719
-Residuo = €81

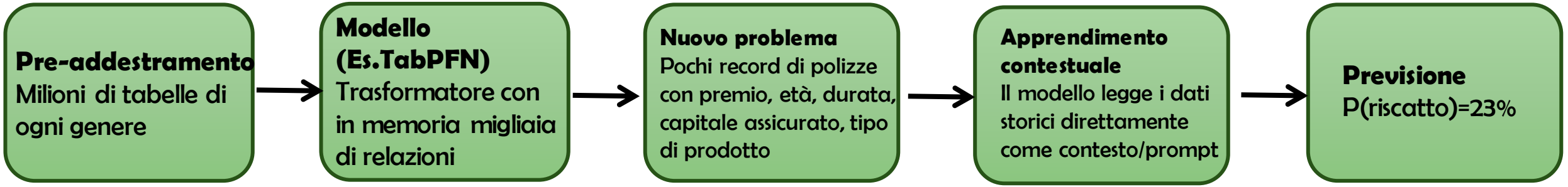
Albero 1 €700 + Albero 2 0.1*100=10 + Albero 3 0.1*90=9 = Stima parziale €719

✓ Alta accuratezza predittiva
✓ Robusto a valori anomali e dati misti

✗ Meno interpretabile
✗ Richiede sintonizzazione dei parametri

I modelli a fondazione per dati tabulari

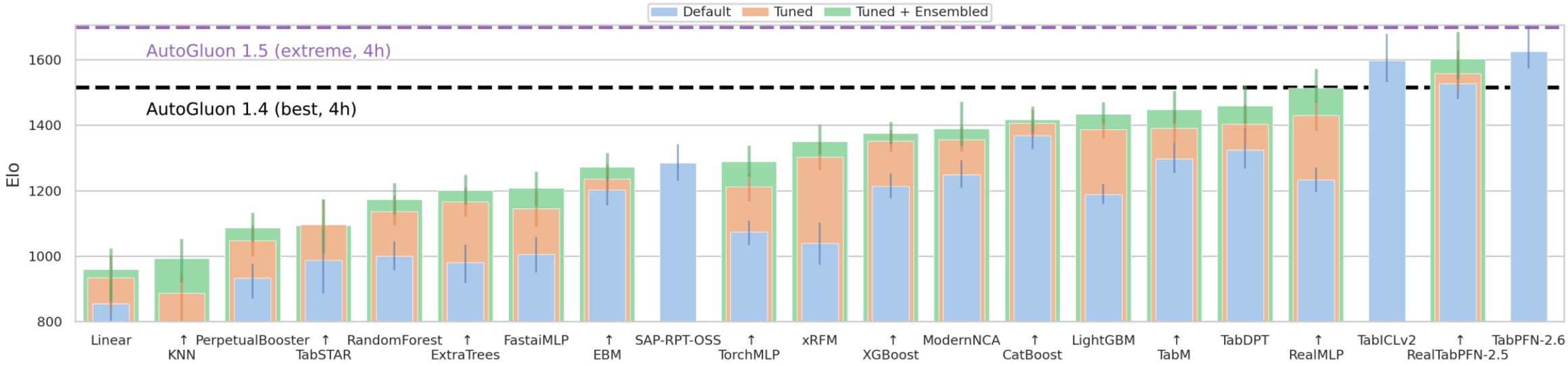
Obiettivo: fare previsioni con un modello pre-addestrato (zero shot learning)
Esempio di applicazione: prevedere la probabilità di riscatto di una polizza con pochi dati disponibili



TabArena Arena Leaderboard

Questa classifica i modelli di previsione in base alle prestazioni ottenute sui compiti di regressione e di classificazione.

[TabArena - a Hugging Face Space by TabArena]

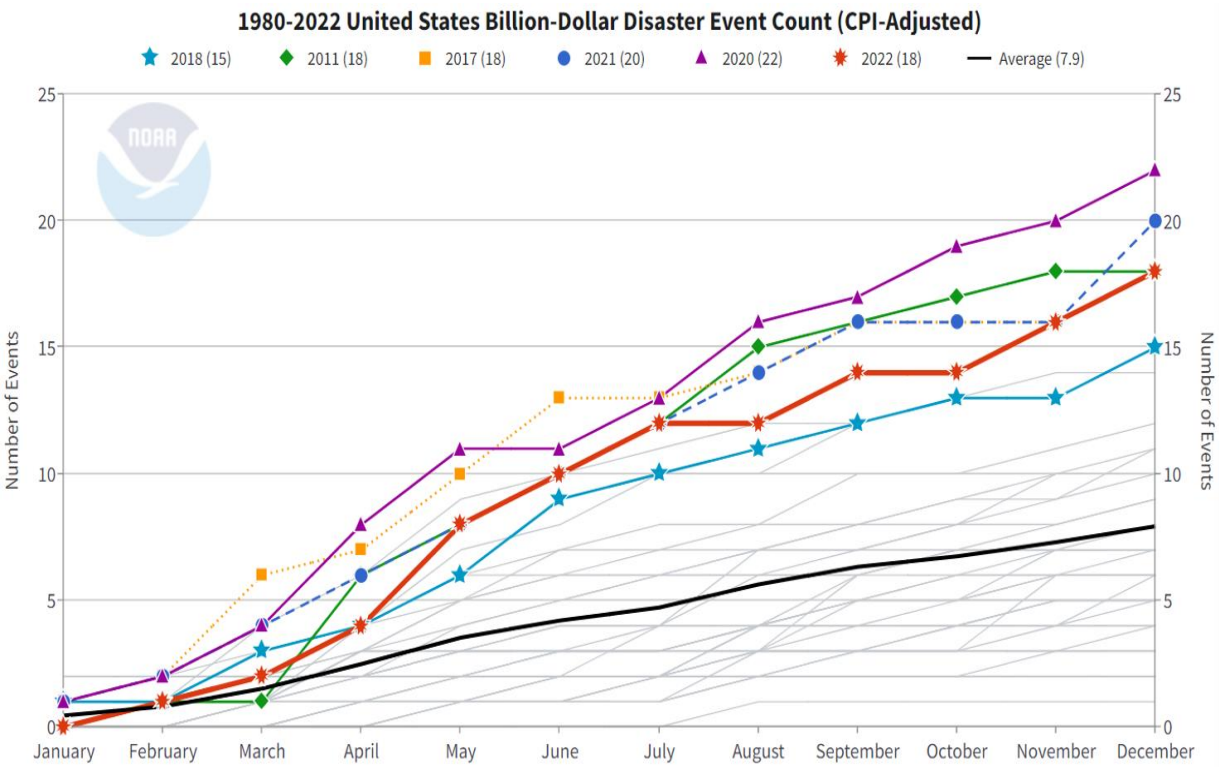


Opportunità attuariali

- Potenziamento del gradiente => Tariffazione/Riservazione/Frodi

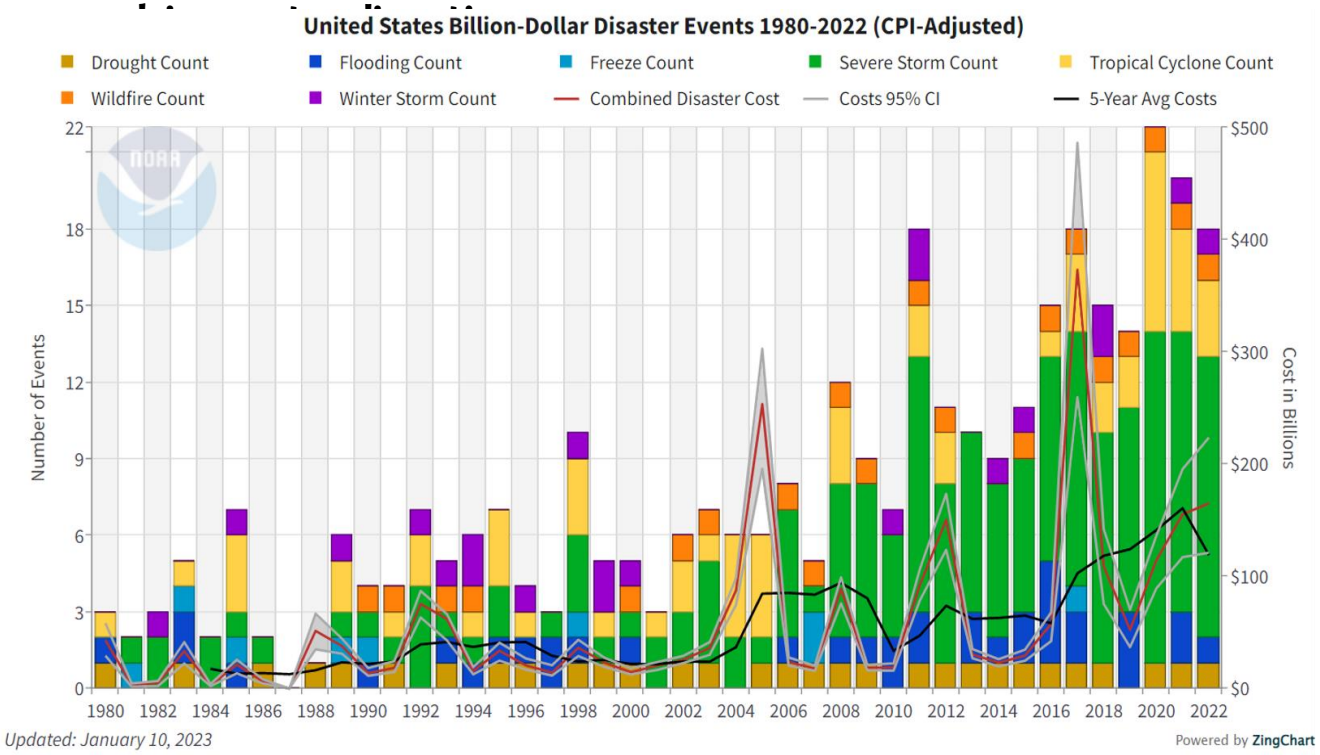
- Modelli a fondazione per dati tabulari: creazione di nuove variabili e arricchimento dei dati mancanti, tasso di abbandono.

Applicazione ML e LLM: Influenza del cambiamento climatico sugli eventi naturali

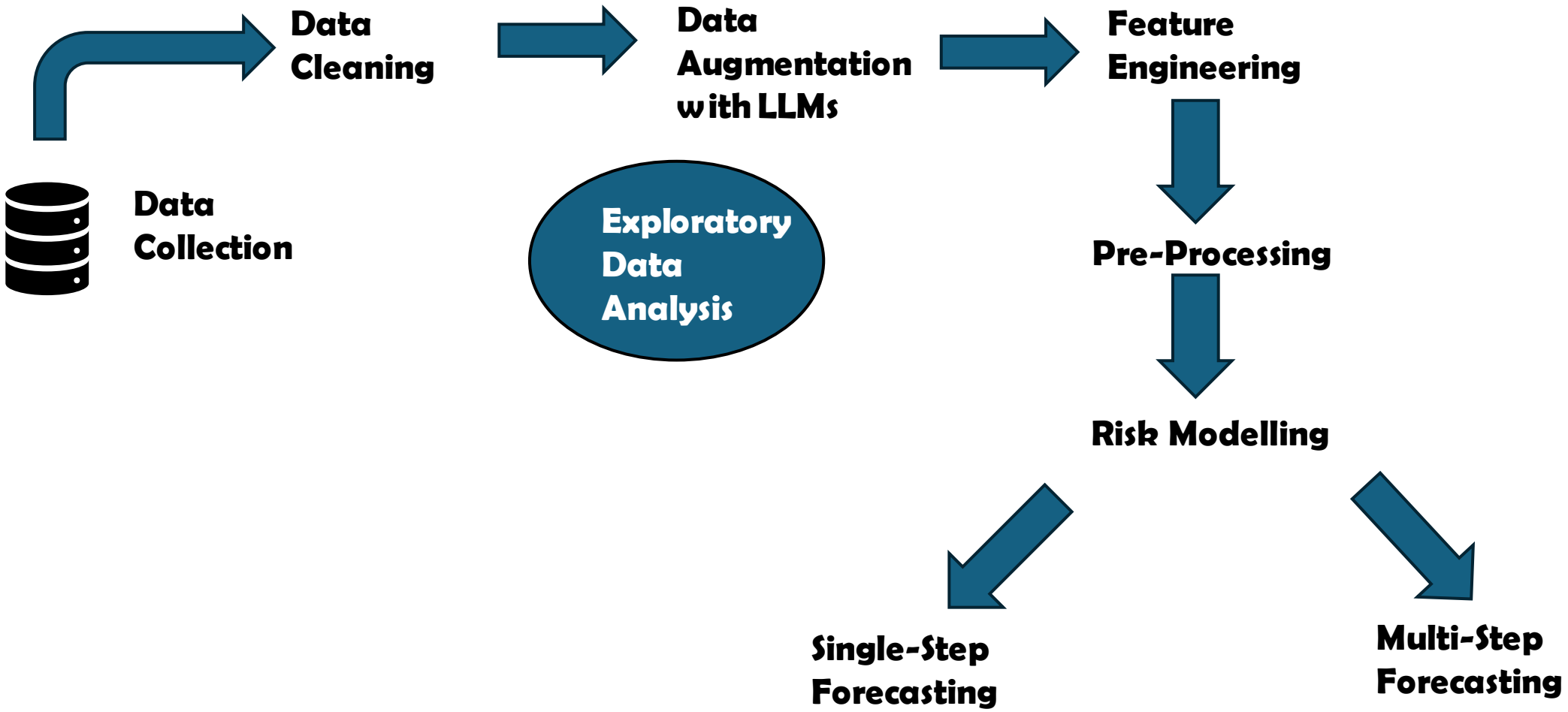


https://www.climate.gov/news-features/blogs/beyond-data/2022-us-billion-dollar-weather-and-climate-disasters-historical

Nel 2023, il Centro Nazionale per l'Informazione Ambientale (NCEI) della NOAA (National Oceanic and Atmospheric Administration) ha pubblicato il rapporto sui disastri meteorologici e climatici negli Stati Uniti del 2022. Dal 1980, gli Stati Uniti hanno subito 341 disastri meteorologici e climatici per i quali i danni e i costi complessivi hanno raggiunto o superato il miliardo di dollari, per un costo cumulativo superiore a 2,475 trilioni di dollari. Mitigare i rischi futuri richiede di affrontare i pericoli combinati derivanti dal



Applicazione ML e LLM: Influenza del cambiamento climatico sugli eventi naturali

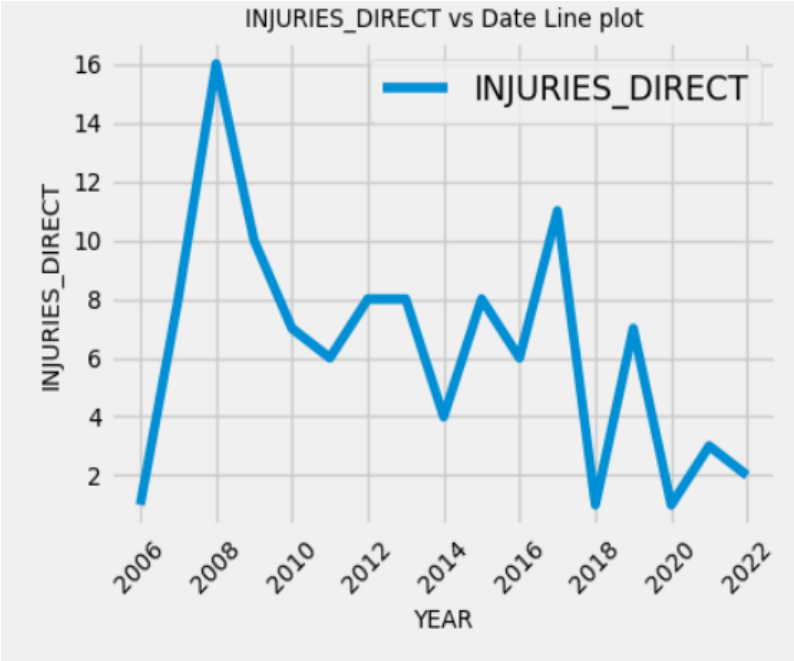


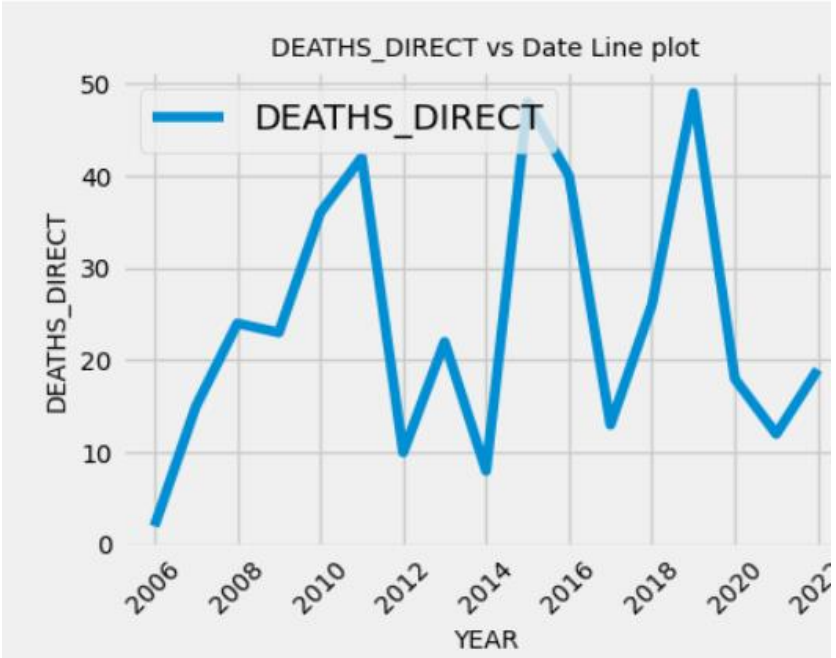
US Injuries Flood Prediction with Large Language Models Data Augmentation

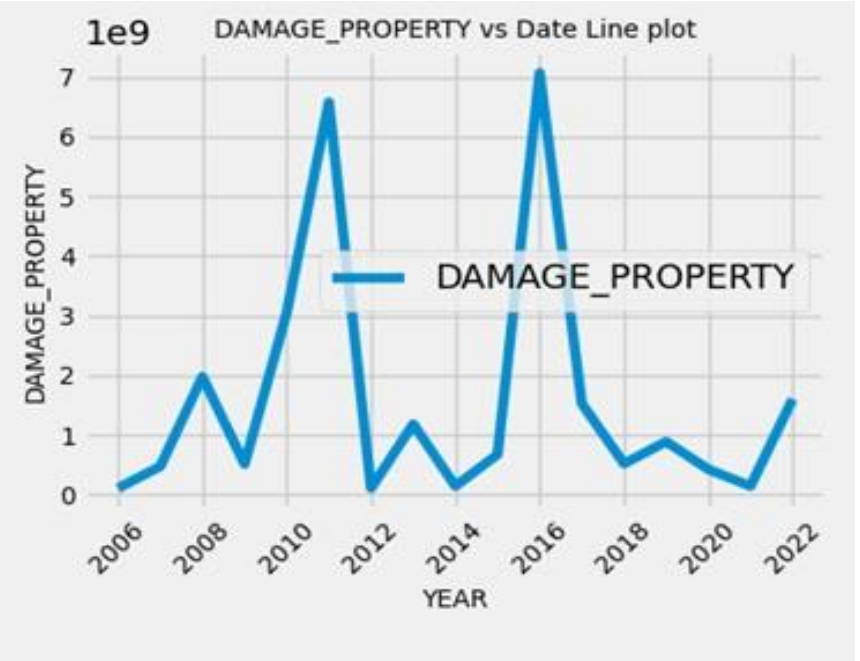
Applicazione ML e LLM: Influenza del cambiamento climatico sugli eventi naturali

Dopo la pulizia e l'aumento dei dati (data augmentation), il dataset ha raggiunto la seguente struttura:
53 colonne e 38.398 righe. Le osservazioni spaziano dal 2006 al 2022.

- ✓ **La variabile target "Injuries Direct" (Lesioni Dirette)** mostra un picco di infortuni nel 2006, seguito da una diminuzione negli anni successivi, con un secondo picco nel 2017. La California ha registrato il maggior numero di infortuni.
- ✓ **La variabile target "Deaths Direct" (Morti Dirette)** mostra un andamento fluttuante, con picchi di mortalità osservati nel 2011, 2015 e 2019. Kentucky, Missouri e North Carolina hanno registrato il maggior numero di decessi.
- ✓ **La variabile target "Damage Property" (Danni alle Proprietà)** indica due picchi di disastri nel 2011 e nel 2016, con la Louisiana che presenta il livello di danni più elevato.





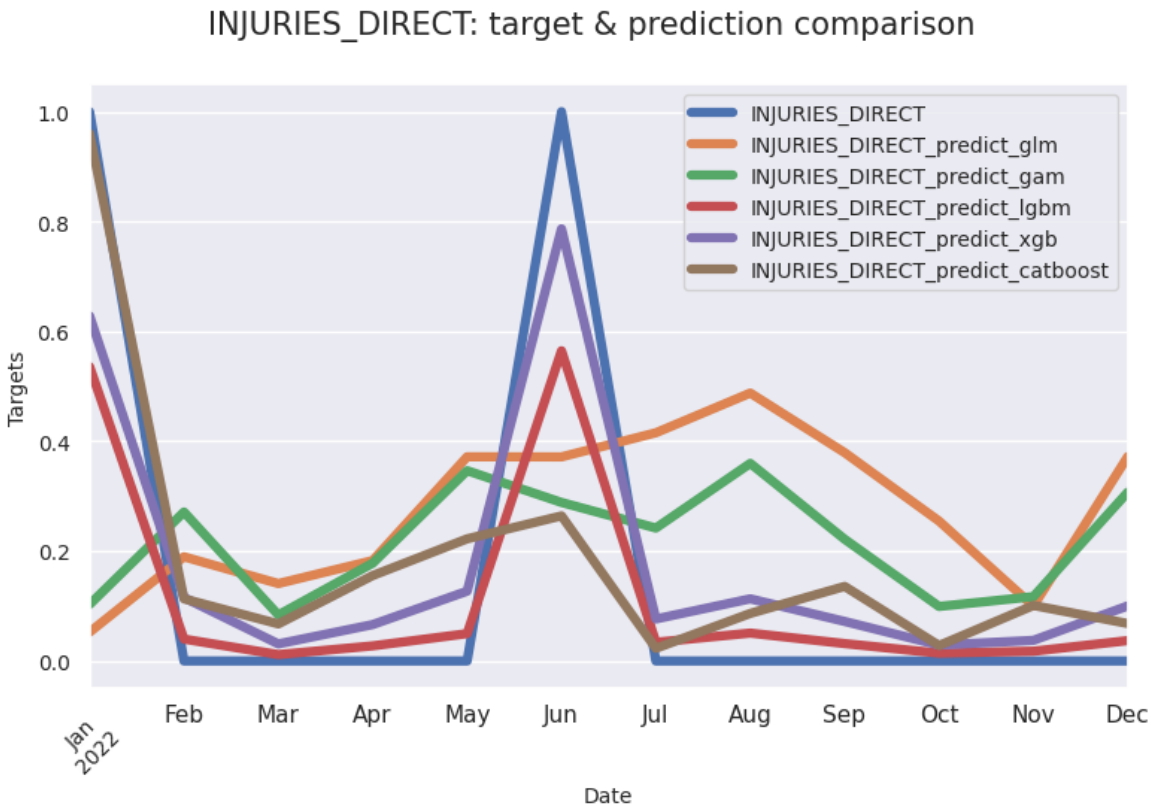


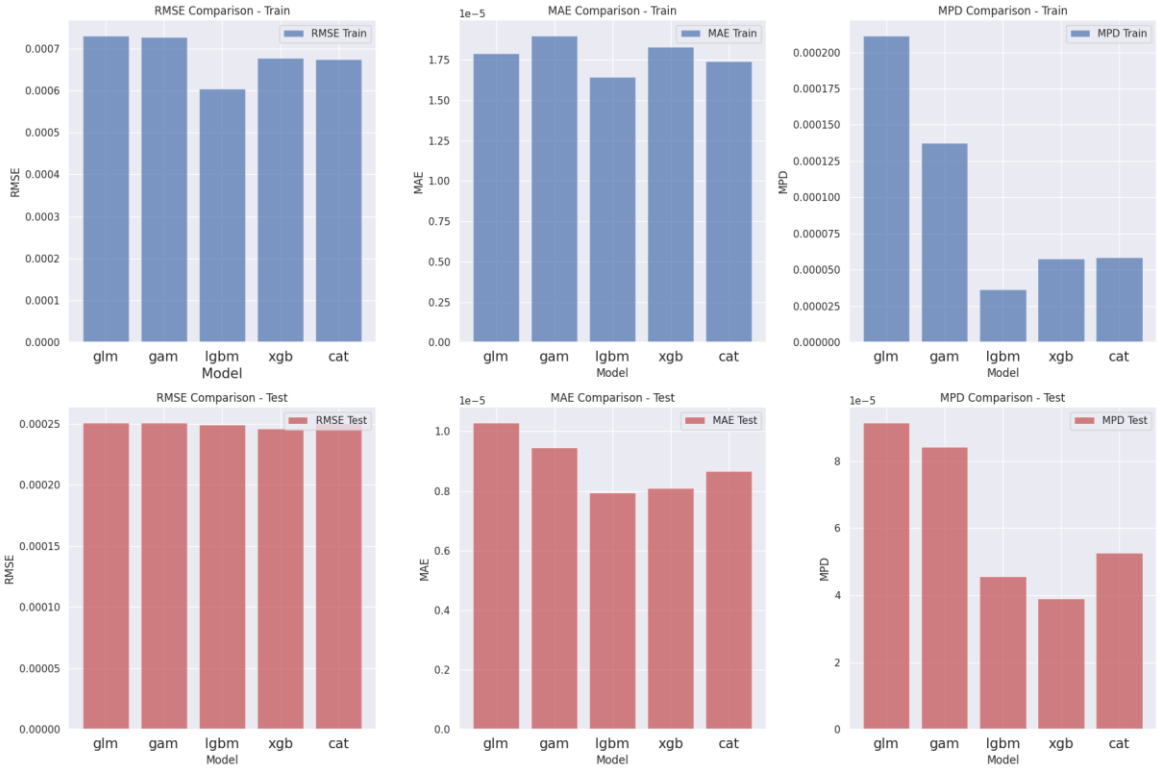
Applicazione ML e LLM: Influenza del cambiamento climatico sugli eventi naturali

Train:
2006-2021

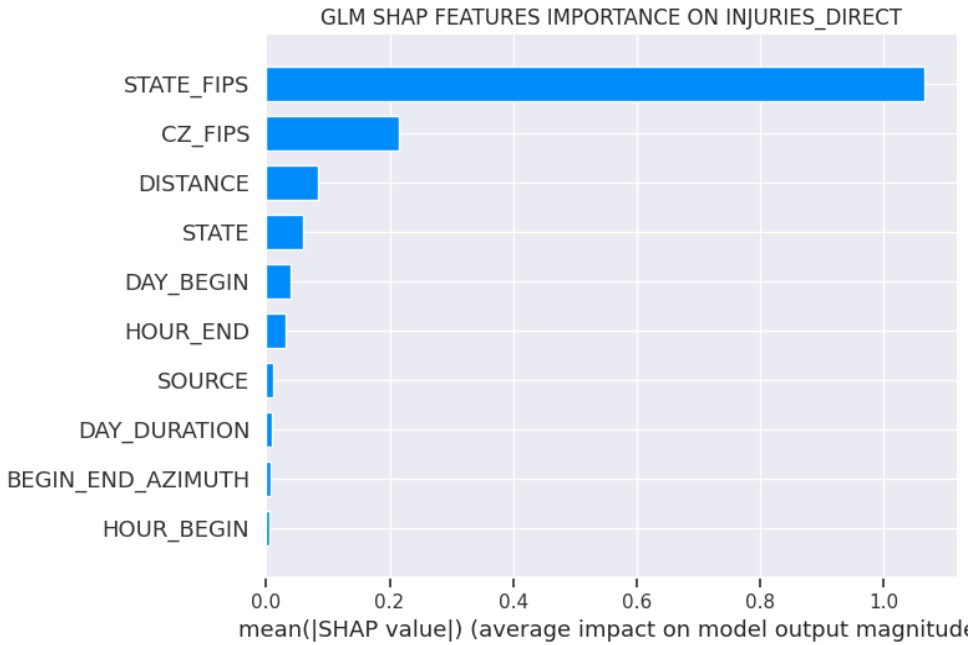
Test:
2022

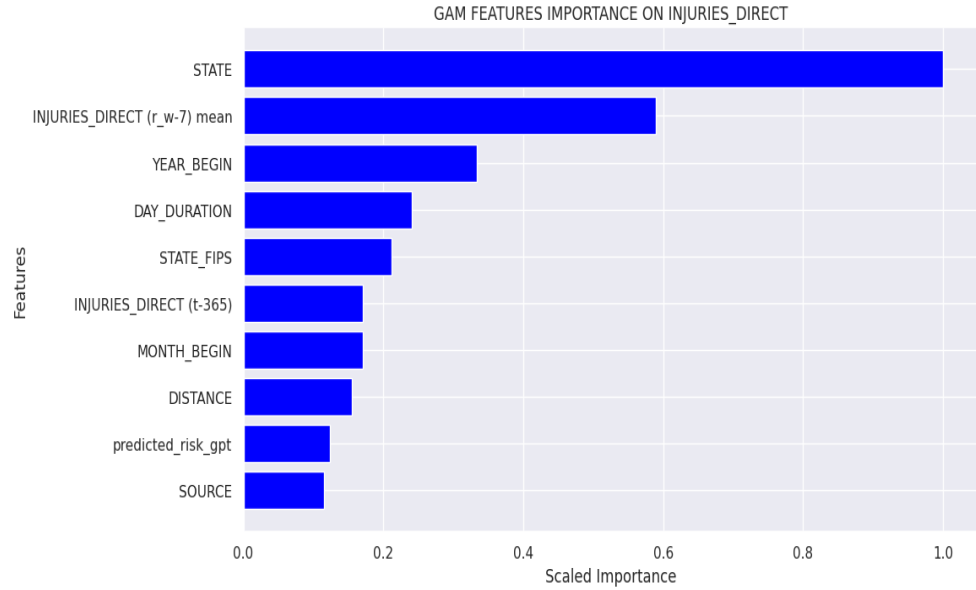
Comparazione delle prestazioni tra diverse metriche: MAE, RMSE, MPD. I modelli della famiglia Gradient Boosting offrono prestazioni superiori rispetto ai modelli attuariali tradizionali.

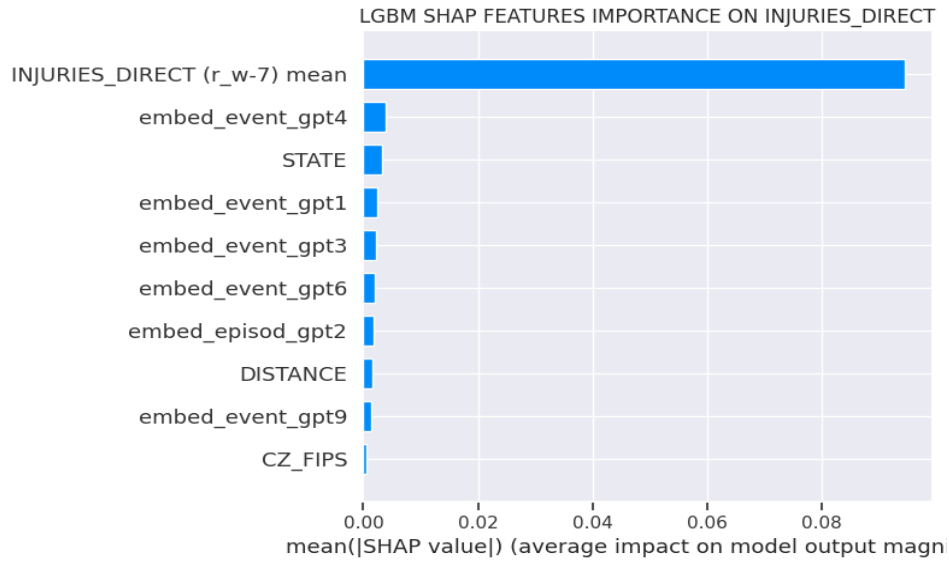




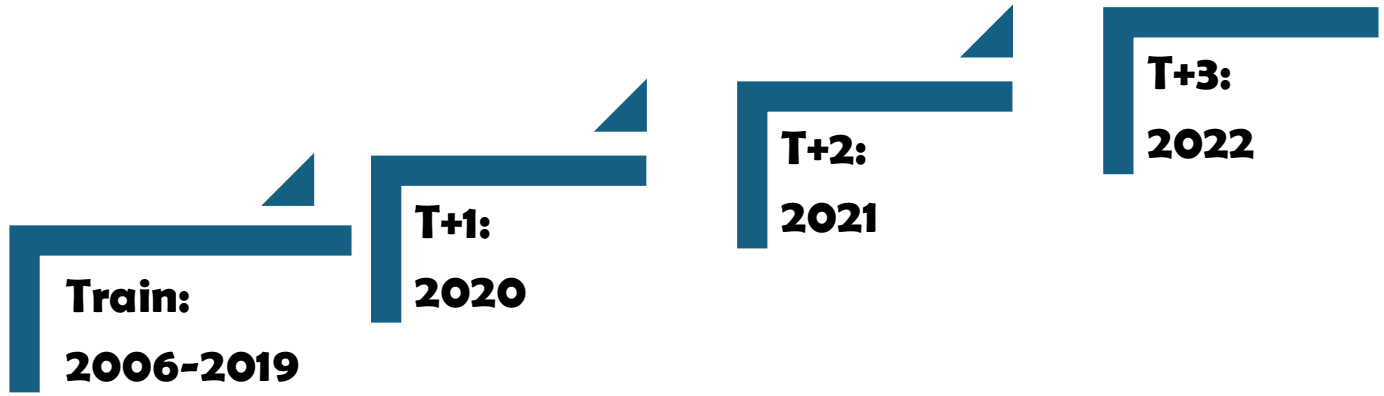
Applicazione ML e LLM: Influenza del cambiamento climatico sugli eventi naturali







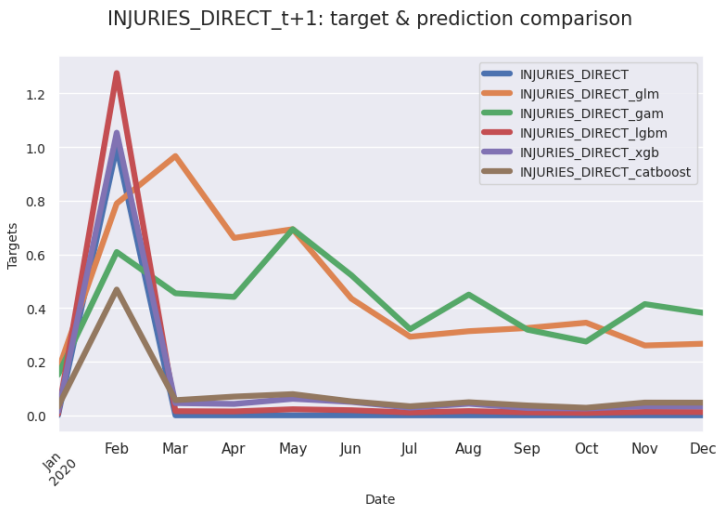
Applicazione ML e LLM: Influenza del cambiamento climatico sugli eventi naturali

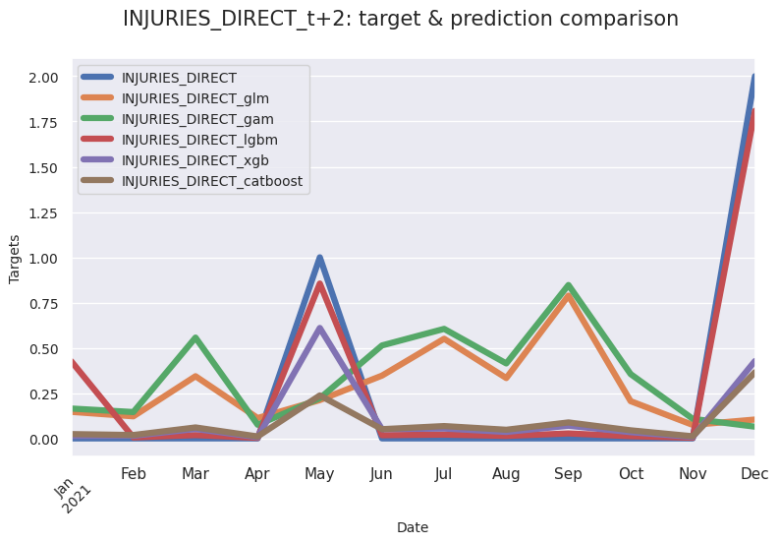


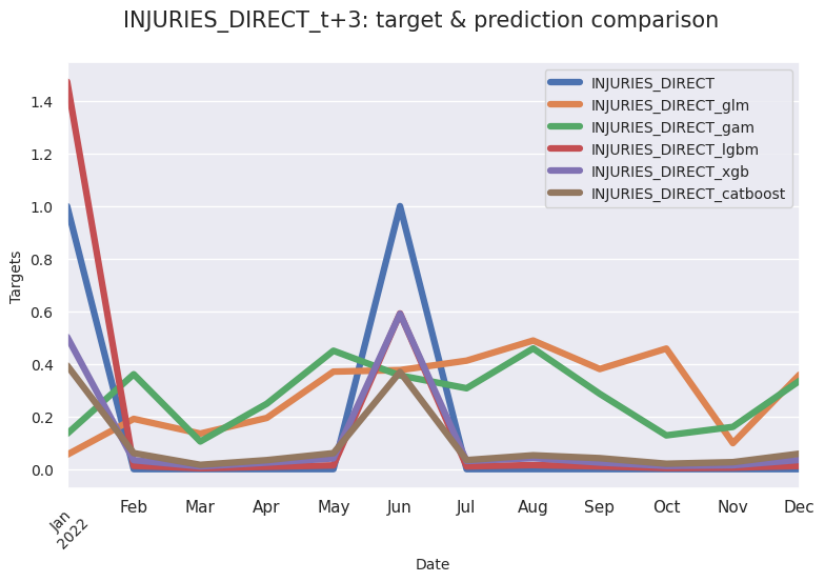
T+1

T+2

T+3

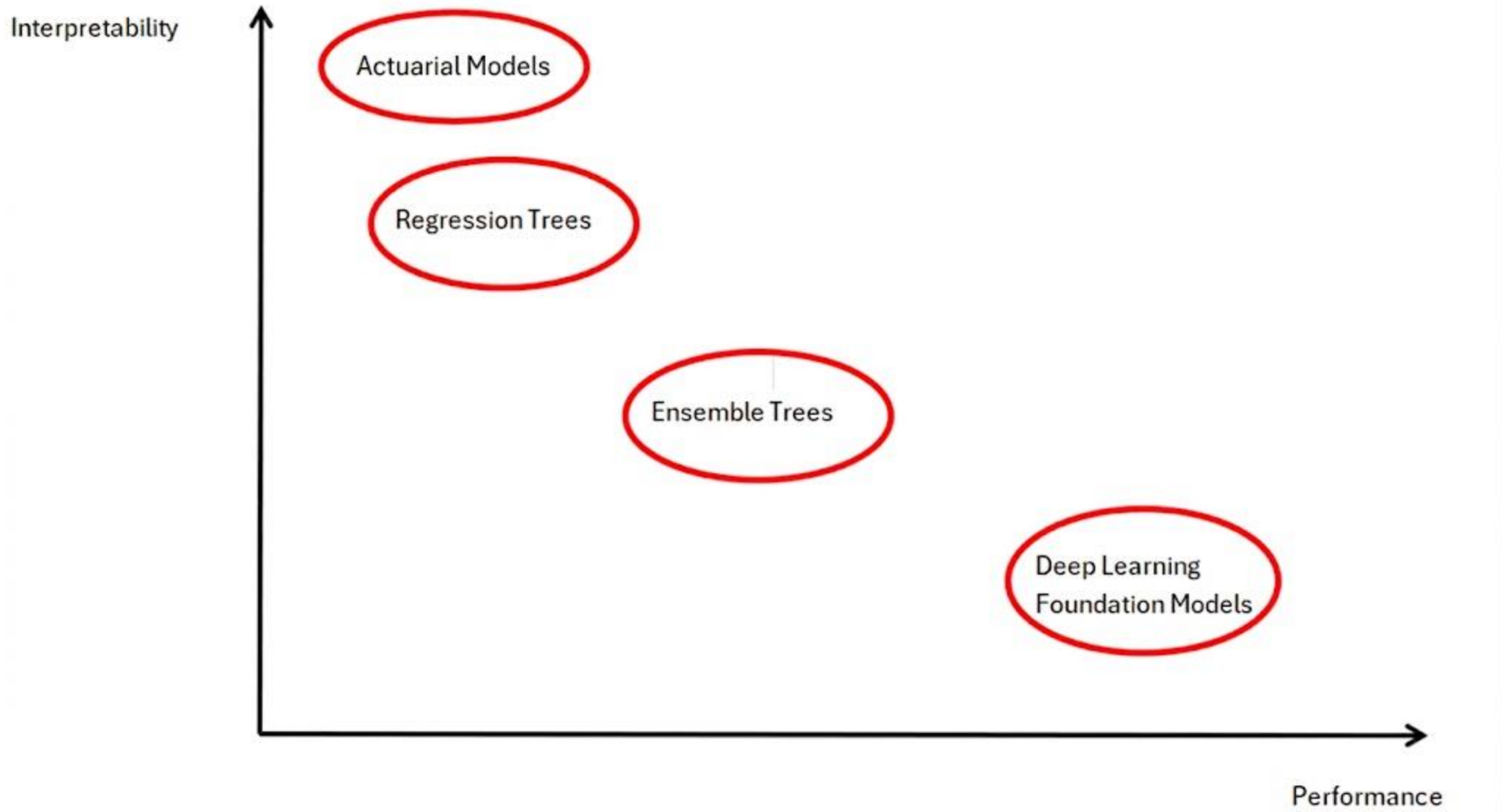






Interpretabilità e gestione del rischio di modello per la validazione attuariale

Interpretabilità vs Prestazione



SHAP: un metodo di riferimento per l'interpretabilità

SHapley · Additive · eXplanations

欄 Cos'è

Spiega *quanto* ogni variabile ha contribuito alla singola previsione del modello — positivamente o negativamente.

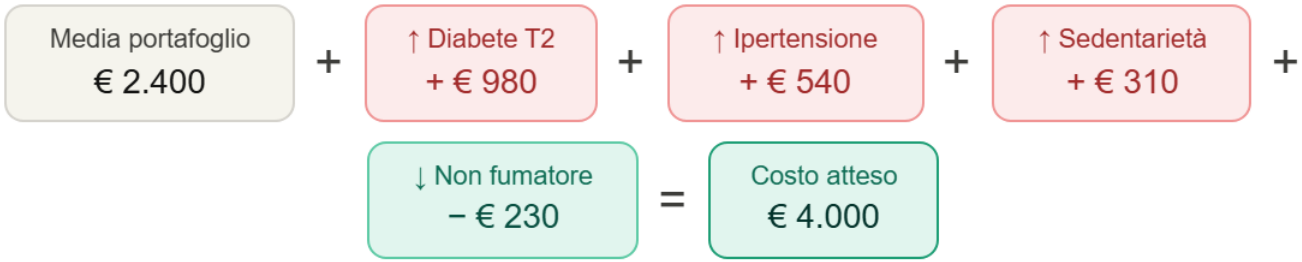
□ Da dove viene

Nasce dalla teoria dei giochi: come distribuire equamente il "merito" del risultato tra più fattori che cooperano.

□ Proprietà chiave

La somma dei contributi SHAP, più la media del modello, ricostruisce esattamente la previsione finale.

□ Esempio: un modello stima il costo sanitario atteso nell'anno per un assicurato di 58 anni, con diabete di tipo 2, ipertensione e sedentario. La media del portafoglio è €2.400. SHAP risponde: «Quanto ha *pesato* ciascuna caratteristica sullo scostamento dalla media?»



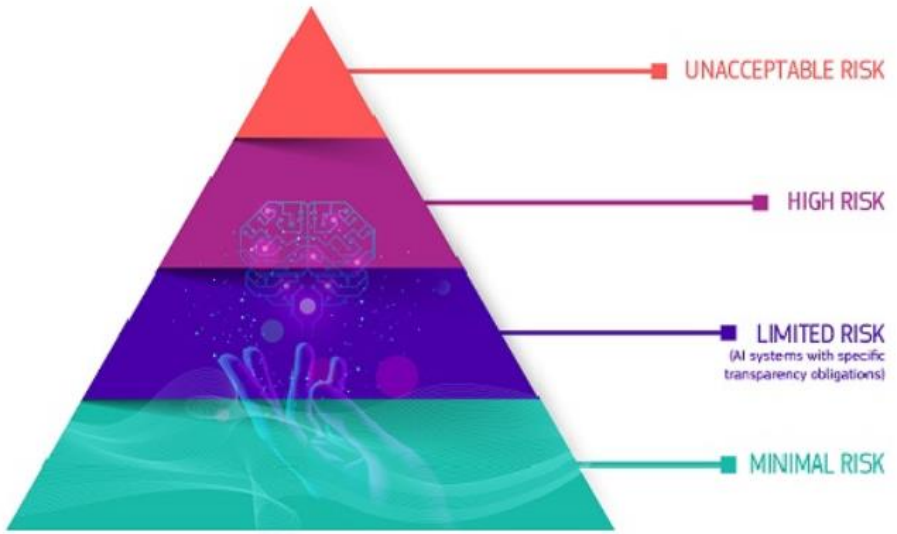
□ Analisi locale — singolo assicurato

□ Analisi globale — intero portafoglio

L’approccio dell’EU AI Act nell’ambito assicurativo

Un approccio basato sul rischio

La legge sull'IA definisce 4 livelli di rischio per i sistemi di IA:



fonte

Ramo assicurativo	Livello di rischio secondo AI Act	Riferimento normativo
Vita e salute	**Alto rischio**	Reg. 2024/1689, Art. 6 & Allegato III, punto 5
Danni (auto, RC, ecc.)	**Non automaticamente alto rischio** (obblighi generali di trasparenza e governance)	Reg. 2024/1689, Artt. 13–14 & Art. 6

Model Risk Management

Checklist per la validazione di un modello di apprendimento automatico

- Separazione dei dataset in set di training, di validazione e di test.
- Benchmark contro modello semplice.
- Analisi di stabilità.
- Audit delle variabili.
- Documentazione delle assunzioni.
- Piano di monitoraggio.

Ricerca, casi d'uso e prospettive future

Contributi ricerca in ambito IA

Advanced Applications of Generative AI in Actuarial Science: Case Studies Beyond ChatGPT: Esplora quattro case study implementati, tra cui l'uso di LLM per estrarre feature da dati testuali non strutturati, migliorando la previsione dei costi dei sinistri; l'automazione delle comparazioni di mercato con Retrieval-Augmented Generation; la classificazione dei danni auto con vision-LLM; e sistemi multi-agente per l'analisi dei dati.

A Primer on Generative AI for Actuaries (Society of Actuaries, 2024): Fornisce un'introduzione tecnica alle applicazioni di GenAI per gli attuari, coprendo la produttività generale (ad es. sommario di documenti, redazione), coding, documentazione dei modelli, arricchimento dei dati sintetici, analisi degli scenari, claims e underwriting, con enfasi su benefici, limitazioni e checklist pratiche.

From Traditional AI to Generative AI: Implications for the Insurance Sector (EIOPA, 2025): Il documento di Petra Hielkema (EIOPA) esplora il passaggio dall'IA tradizionale all'IA generativa nel settore assicurativo, evidenziando opportunità in termini di efficienza e personalizzazione. Tuttavia, mette in guardia contro rischi quali i bias discriminatori, le "allucinazioni" e la dipendenza dai grandi fornitori tecnologici. La strategia europea, guidata dall'AI Act, punta a una governance rigorosa e a una supervisione umana per bilanciare l'innovazione tecnologica con la tutela dei consumatori.

Generative AI in the Actuarial Profession: Survey Insights (IFoA, 2025): Il rapporto descrive una professione in una fase di "ottimismo pragmatico": gli attuari non sono semplici osservatori, ma stanno attivamente integrando questi strumenti per migliorare l'efficienza, pur mantenendo il rigore analitico e la prudenza tipici della categoria. La sfida futura sarà integrare le competenze di data science con il giudizio attuariale tradizionale.

Contributi ricerca in ambito IA

Time-Series Forecasting of Mortality Rates using Deep Learning: Il paper propone un modello di Deep Learning basato su reti convoluzionali (CNN) per prevedere i tassi di mortalità, superando i limiti del classico metodo Lee-Carter. Gli autori dimostrano che un'architettura relativamente semplice, arricchita da *embedding* per genere e area geografica, offre una precisione superiore e una capacità di generalizzazione migliore. Testato sui dati dell'Human Mortality Database, il modello automatizza la previsione senza richiedere interventi manuali né calibrazioni specifiche per ogni popolazione.

AI Tools for Actuaries: Il documento è un set di dispense didattiche progettato per aggiornare gli attuari sulle tecniche di AI e Data Science, dai modelli di regressione classica al Deep Learning e agli LLM. Il testo integra rigore statistico e applicazioni pratiche, trattando temi quali la spiegabilità dei modelli (XAI), l'apprendimento per rinforzo e la visualizzazione dei dati. L'obiettivo è fornire una solida base tecnica per automatizzare i processi attuariali e gestire i dati complessi in modo etico e responsabile.

IA e mercato assicurativo

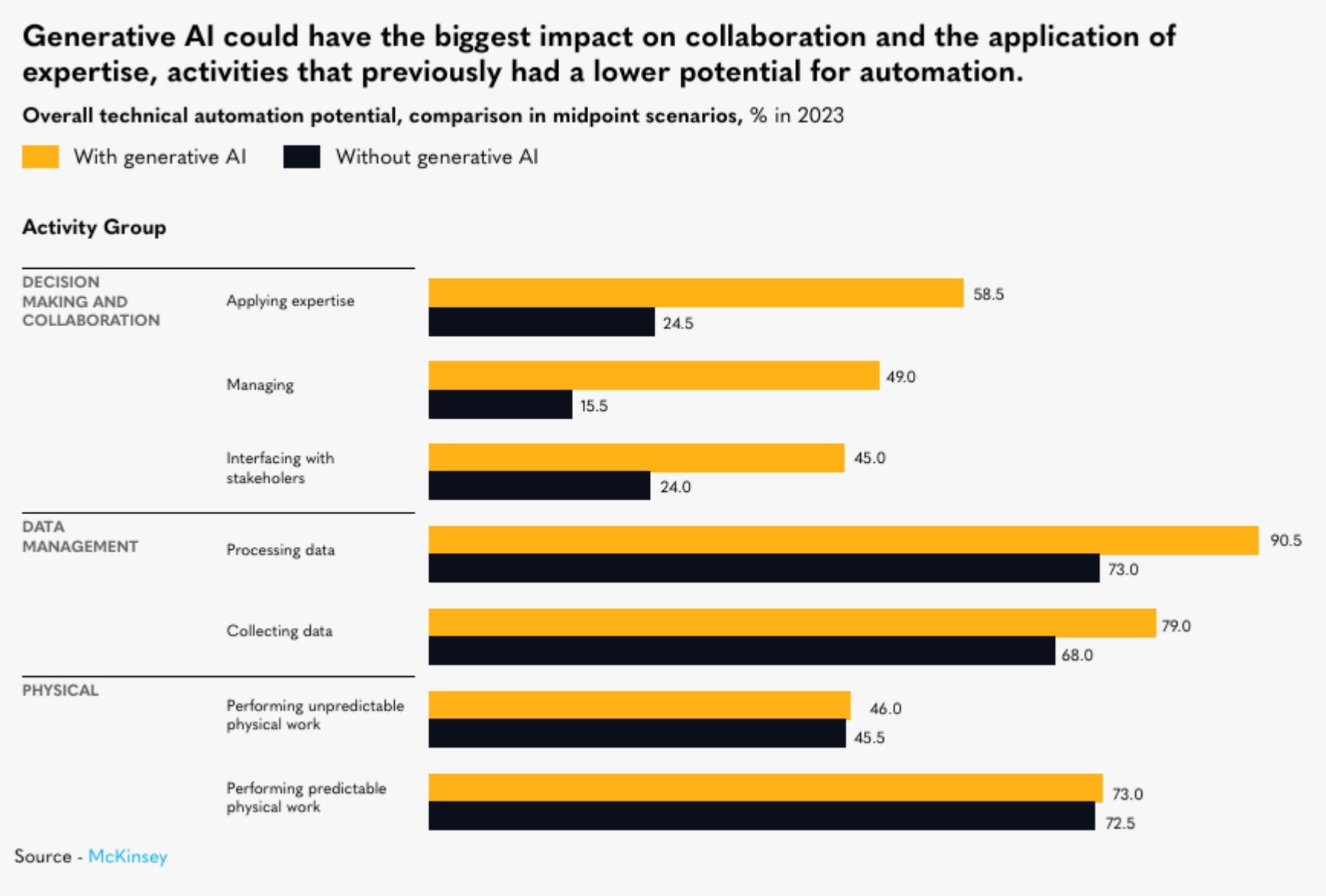
LMA: il potenziale dell'AI e del machine learning non sfruttato nella gestione attuariale e del rischio L'indagine LMA evidenzia che l'IA e il Machine Learning sono ancora sottoutilizzati nel settore attuariale, limitandosi perlopiù all'automazione di compiti semplici. Nonostante l'ottimismo degli esperti, pesano ostacoli come la scarsa fiducia nei modelli, la complessità normativa e la carenza di competenze specifiche. Per superare questa fase, le imprese devono investire in formazione e integrazione strategica, trasformando le sfide tecnologiche in un vantaggio competitivo concreto.

Artificial Intelligence at Allianz Two Use Cases Allianz utilizza l'IA per automatizzare i sinistri semplici (Project Nemo), riducendo i tempi dell'80% tramite agenti digitali, e per rilevare frodi complesse (Incognito) tramite il machine learning. In entrambi i casi, la tecnologia funge da supporto decisionale, lasciando la responsabilità finale del pagamento e dell'autorizzazione agli esperti umani. Questo approccio "human-in-the-loop" consente di gestire grandi volumi di dati, aumentando l'efficienza operativa e i risparmi economici.

Gallagher lancia uno strumento basato sull'IA per l'analisi dei profili di rischio Gallagher ha lanciato Blueprint, uno strumento basato sull'IA che analizza i profili di rischio aziendali confrontandoli con i benchmark di settore. Il sistema genera un punteggio che consente di ottimizzare le coperture assicurative, riducendo i costi e migliorando l'efficacia dei rinnovi. Grazie all'integrazione di dati proprietari e di competenze tecniche, le aziende possono allineare i propri budget alle reali priorità di rischio.

Collaborazione tra umani e IA Generativa

Incremento di una relazione sinergica in cui l'IA gestisce l'analisi dei dati e fornisce intuizioni, mentre la professionalità umana offre competenza, empatia e risoluzione creativa dei problemi per risultati complessivamente migliori nel settore assicurativo.



Riferimenti

Wolfgang Ertel, Introduction to Artificial Intelligence, 2025, Springer
Che cos'è l'intelligenza artificiale (AI)? | Google Cloud
Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 2016, MIT Press
David Foster, Generative Deep Learning, 2023, O'Reilly
Introduction to Statistical Learning
AI Tools for Actuaries by Mario V. Wuthrich, Ronald Richman, Benjamin Avanzi, Mathias Lindholm, Michael Mayer, Jürg Schelldorfer, Salvatore Scognamiglio :: SSRN
Statistical Modeling: The Two Cultures
CS231n Deep Learning for Computer Vision
CME 295 - Transformers & Large Language Models
CS230 Deep Learning
[1706.03762] Attention Is All You Need
Transformer
[1810.04805] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
The Evolution of Language Models: From GPT-1 to GPT-4 and Beyond - GeeksforGeeks
Improving Language Understanding by Generative Pre-Training
[2311.05661] Prompt Engineering a Prompt Engineer (arxiv.org)
[2402.07927] A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications (arxiv.org)
[2401.14423] Prompt Design and Engineering: Introduction and Advanced Methods (arxiv.org)
Natural Language Processing with Transformers
Agents | Kaggle
Retrieval-Augmented Generation for Large Language Models: A Survey
AGENTIC RETRIEVAL-AUGMENTED GENERATION: A SURVEY ON AGENTIC RAG
A practical guide to building agents, OpenAI
Build an AI Agent (From Scratch)
ReAct: Synergizing Reasoning and Acting in Language Models
AI Agents

Agentic AI Insurance claims processing and management: The benefits and use cases
LangChain Tutorials - Learn LangChain, RAG & AI Development Step-by-Step
The Open-Source Advantage in Large Language Models (LLMs)
The Strengths and Limitations of Large Language Models
Assessing the Strengths and Weaknesses of Large Language Models
A Primer on Generative AI for Actuaries | SOA
Generative-AI-in-Insurance-A-Game-Changer.pdf
Jobs in artificial intelligence (AI) - Intuit Blog
The Act Texts | EU Artificial Intelligence Act
High-level summary of the AI Act | EU Artificial Intelligence Act
Legge sull'IA | Plasmare il futuro digitale dell'Europa
Advanced Applications of Generative AI in Actuarial Science: Case Studies Beyond ChatGPT
From Traditional AI to Generative AI: Implications for the Insurance Sector
Generative AI in the Actuarial Profession: Survey Insights
Time-Series Forecasting of Mortality Rates using Deep Learning
LMA: il potenziale dell'AI e del machine learning non sfruttato nella gestione attuariale e del rischio
Artificial Intelligence at Allianz Two Use Cases
Gallagher lancia uno strumento basato sull'IA per l'analisi dei profili di rischio
US Injuries Flood Prediction with Large Language Models Data Augmentation
P.J. Brockwell, R.A. Davis (2016). Introduction to Time Series and Forecasting, Springer.
The Future of AI for the Insurance Industry
2022 U.S. billion-dollar weather and climate disasters in historical context | NOAA Climate.gov
NCDC Storm Events Database
Hugging Face Repository
Medium articles
Github Repository

N.B. revisione delle slides effettuata con assistente virtuale

Grazie

Contatti:

Linkedin
Medium
Hugging Face
Github
Website