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

L'innovazione attuariale attraverso l'intelligenza artificiale generativa e l'ingegneria dei prompt

28 Marzo 2025

Claudio G. Giancaterino

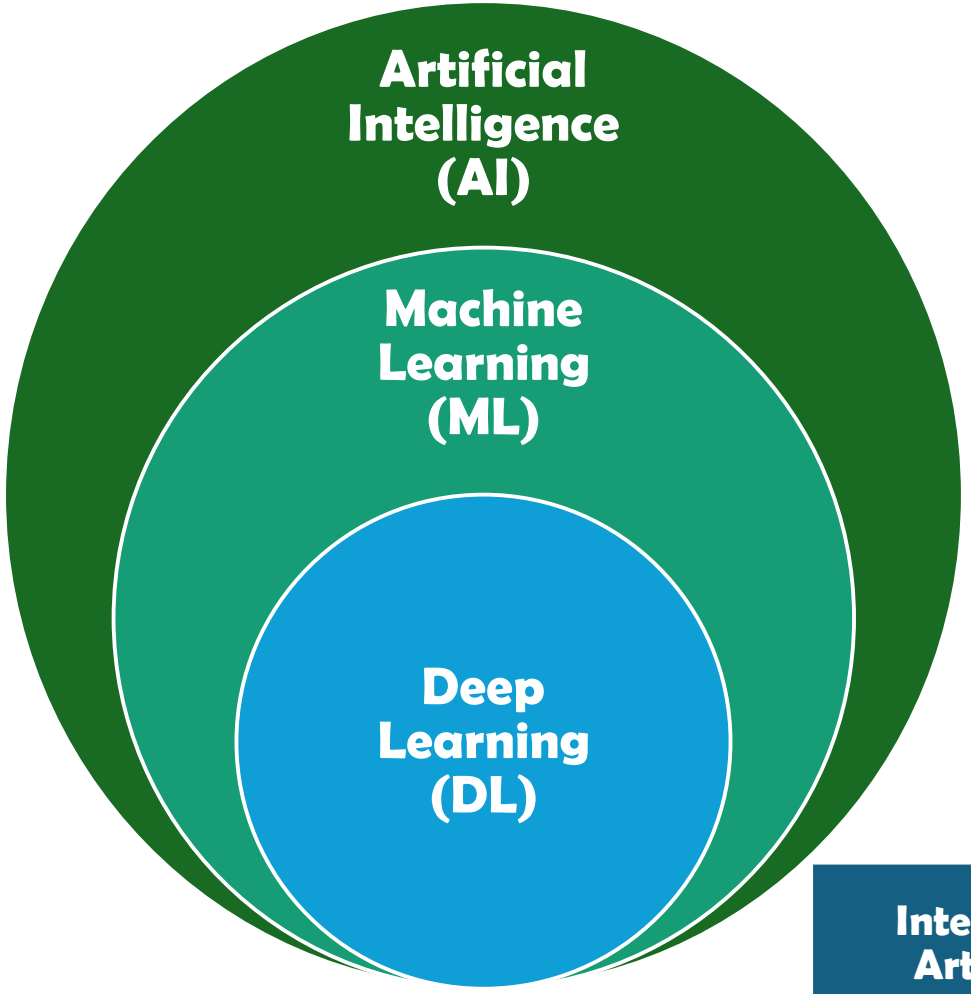
Agenda

-Introduzione all'intelligenza artificiale generativa
-I modelli linguistici e l'ingegneria dei prompt
-Tecniche di ingegneria dei prompt
-Ingegneria dei prompt applicata alla programmazione a supporto degli attuari
-Prospettive future

Introduzione all'intelligenza artificiale generativa

Le professioni attorno all'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 dimensioni 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 decisioni strategiche.

IA Generativa => Crea nuovi contenuti come testi, immagini suoni o dati, simulando il processo creativo umano.

IA Causale => Identifica e modella relazioni causa-effetto per andare oltre le semplici correlazioni statistiche.

IA Spiegabile => Fornisce motivazioni trasparenti e interpretabili sui risultati forniti dai modelli di apprendimento automatico.

IA Adattiva => E' focalizzata sull'apprendimento continuo, adattandosi a contesti e dati in costante mutamento.

Evoluzione storica dei Modelli Generativi

L'Alba della Modellazione Generativa - Processi Stocastici e Primi Approcci Statistici: Catene di Markov (1906) e Catene di Markov Nascoste (HMM) ('60-'70)

L'Ascesa dei Modelli Probabilistici - Rappresentare Dipendenze Complesse: Boltzman Machines (1985) & Restricted Boltzman Machines (1986); Gaussian Mixture Models (GMM) ('70), applicazione Copule (teoria risale agli anni '50)

L'Approccio Bayesiano alla Modellazione Generativa: Bayesian Networks (1985) e Latent Dirichlet Allocation (LDA) (2003)

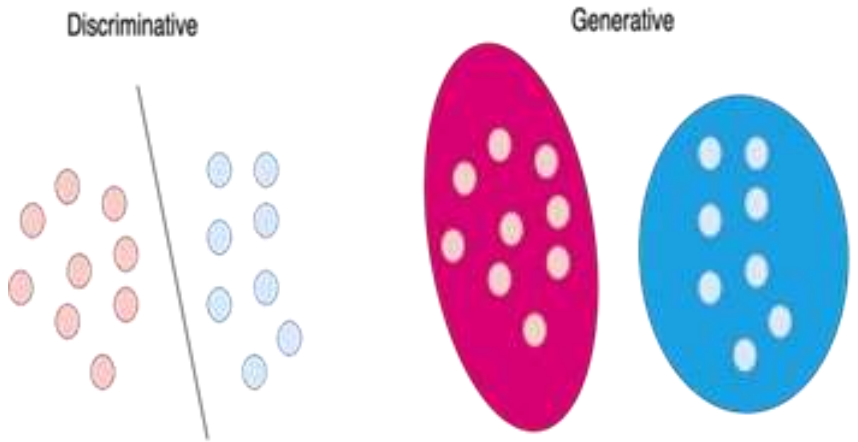
L'Era del Deep Learning - Modelli Generativi Potenziati dalle Reti Neurali: Deep Belief Networks (2006)

L'Era Moderna dei Modelli Generativi - Il Deep Learning al Centro della Scena: Variational Autoencoders (VAEs) (2013). Dal 2014 Generative Adversarial Networks (GANs) ed evoluzioni. Dal 2017 Transformer ed evoluzione dei modelli linguistici. Modelli di Diffusione, introdotti nel 2015 e poi diffusi a partire del 2020 con varie evoluzioni.

Due approcci nell'Apprendimento Automatico a confronto

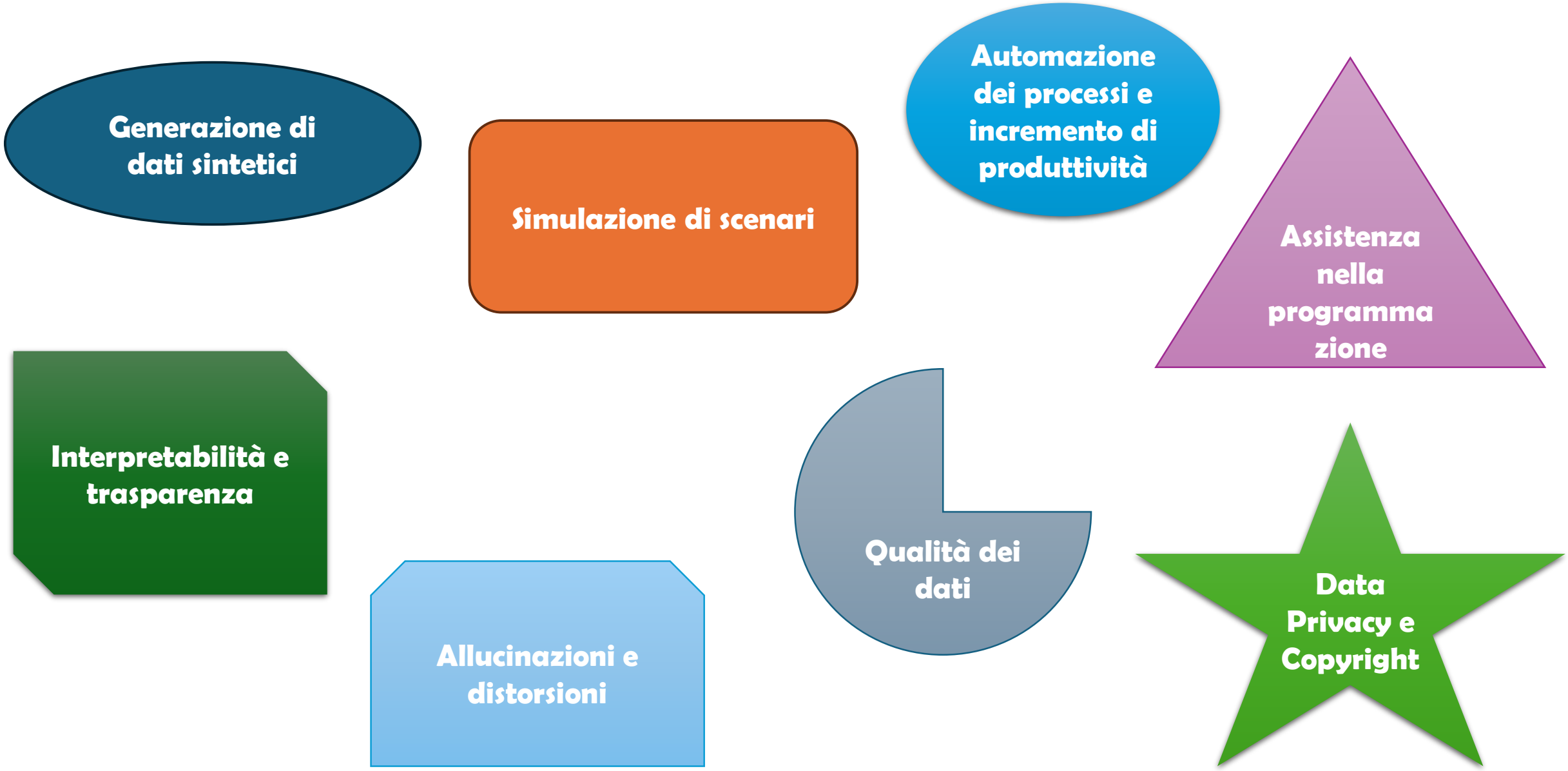
Modelli discriminativi: questi modelli si concentrano sull'apprendimento della distribuzione di probabilità condizionata P(Y|X). Apprendono una funzione per mappare la relazione tra la variabile risposta e le variabili esplicative per effettuare classificazioni o previsioni.

Modelli generativi: questi modelli mirano ad apprendere la distribuzione di probabilità congiunta sui dati P(X,Y). Si concentrano sulla generazione di dati che siano simili ai dati usati per l'addestramento.



fonte

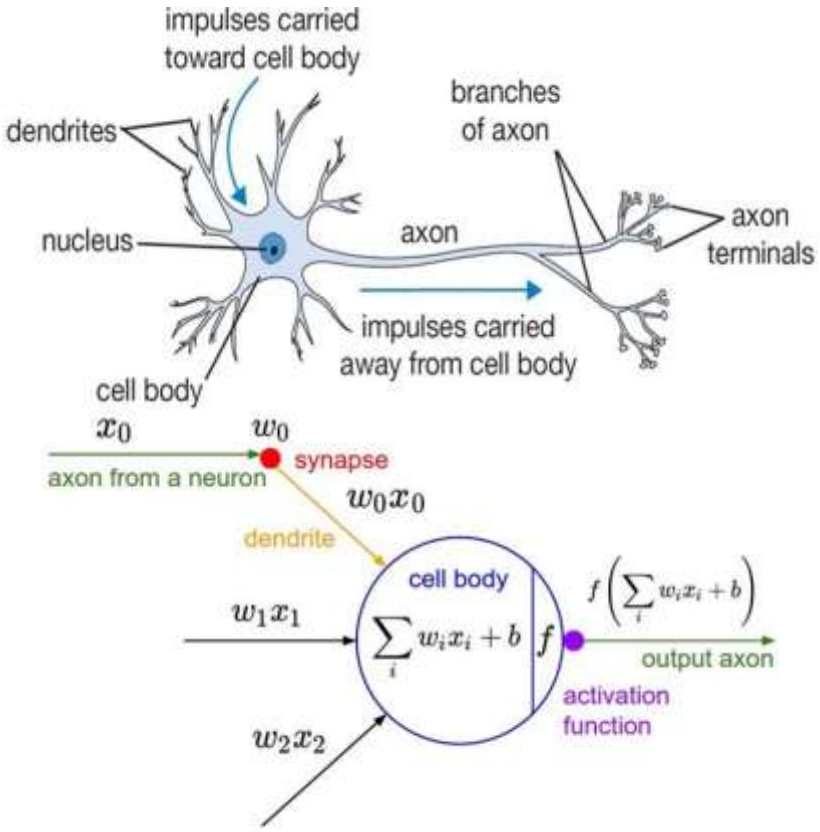
Intelligenza Artificiale Generativa: applicazioni e sfide in ambito attuariale



I modelli linguistici e l'ingegneria dei prompt

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

Le reti neurali artificiali, ispirate alle neuroscienze, sono costituite da unità di calcolo, chiamate neuroni, collegate tra loro in maniera articolata. Ogni neurone riceve un segnale di input da tutti i neuroni dello strato precedente e invia il proprio segnale di output a quelli dello strato successivo. Queste reti combinano funzioni diverse; infatti, ampliano i modelli lineari applicando una trasformazione non lineare, la funzione di attivazione, alla combinazione lineare degli input provenienti dagli strati precedenti. Costruire una rete neurale è simile a realizzare una struttura con i mattoncini Lego: si assemblano singoli pezzi (neuroni) per realizzare sistemi sempre più complessi.



Neurone Artificiale

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

Rete Neurale a strati (feedforward)

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

fonte

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).

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

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

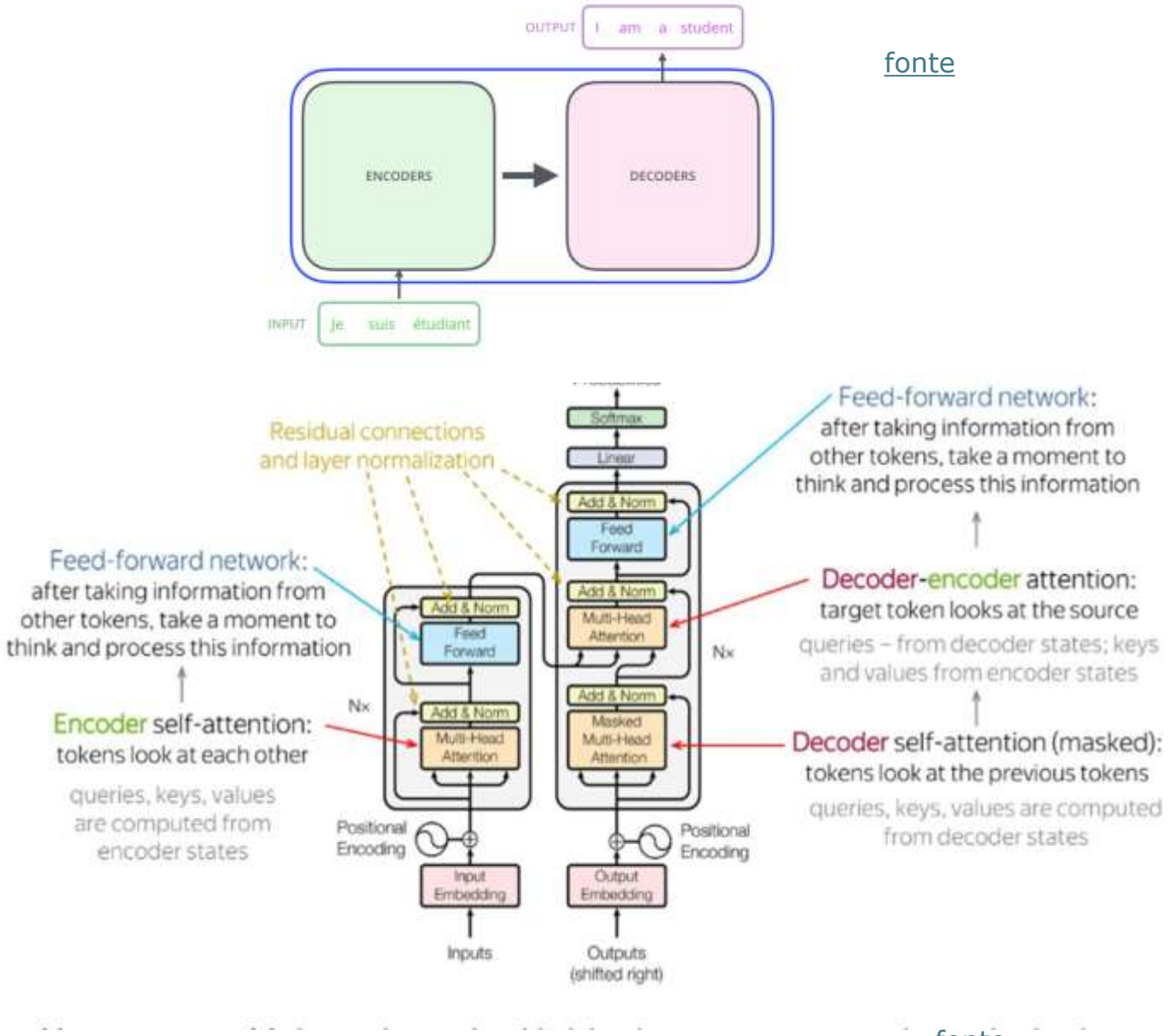
Łukasz Kaiser* Google Brain lukaszkaiser@google.com

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.

L'intelligenza artificiale è

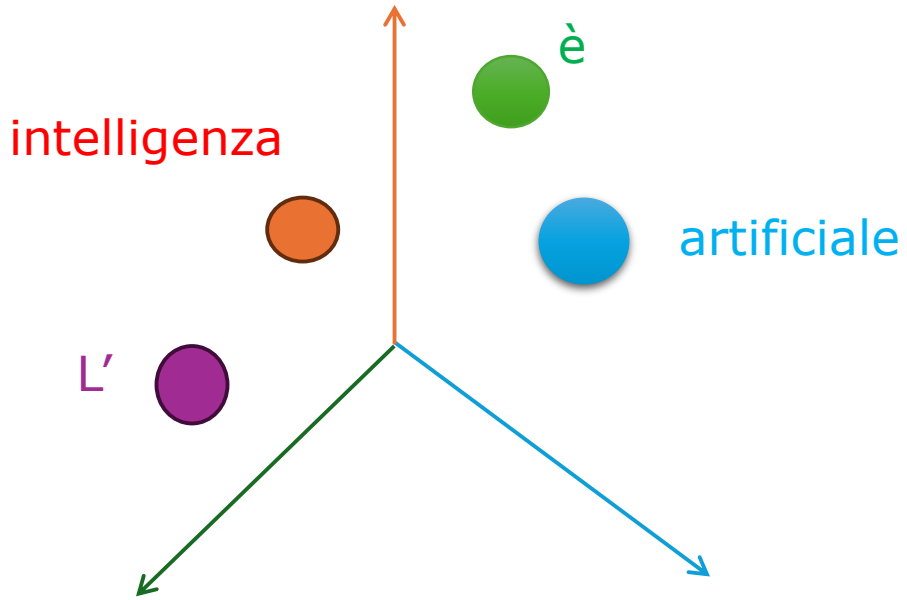
Dizionario delle unità:

«L'»: 1349
«intelligenza»: 1543
«artificiale»: 12456
«è»: 218

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:
«L'» (1349):[0.2,0.5,0.7]
«intelligenza» (1543): [0.3,0.6,0.4]
«artificiale» (12456): [0.7,0.5,0.2]
«è» (218): [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

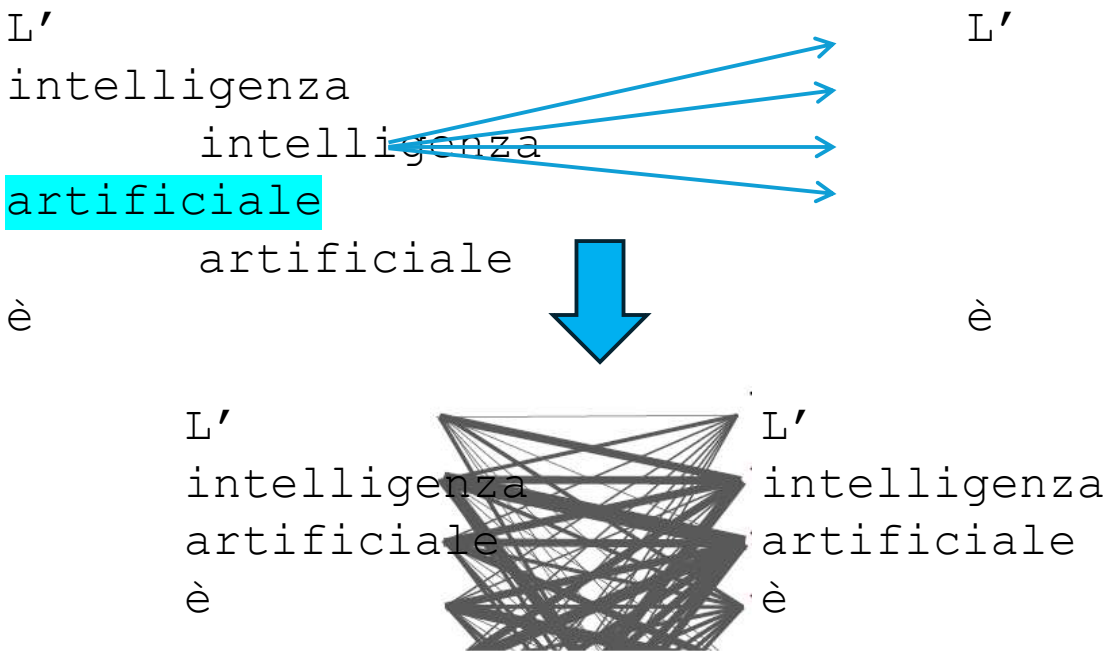
La rappresentazione vettoriale delle unità e del posizionamento sono aggiunte per preservare l'informazione sull'ordine delle unità nel testo

Word	token IDS	token embedding	positional embedding	final input embedding	
L'	1349	[0.2,0.5,0.7]	[0.0,0.0,0.0]	[0.2,0.5,0.7]	position:0
intelligenza	1543	[0.3,0.6,0.4]	[0.1,0.1,0.1]	[0.4,0.7,0.5]	position:1
artificiale	12456	[0.7,0.5,0.2]	[0.2,0.2,0.2]	[0.9,0.7,0.3]	position:2
è	218	[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



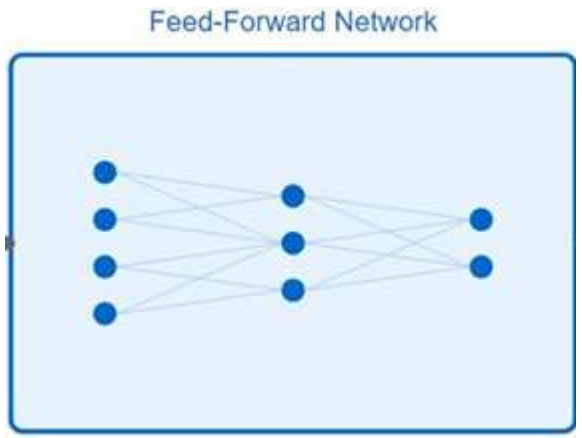
Funzionamento del trasformatore

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".

Pesi dal processo di attenzione

L’	[1349]	: 0.56
intelligenza	[1543]	: 0.83
artificiale	[12456]	: 0.92
è	[218]	: 0.62

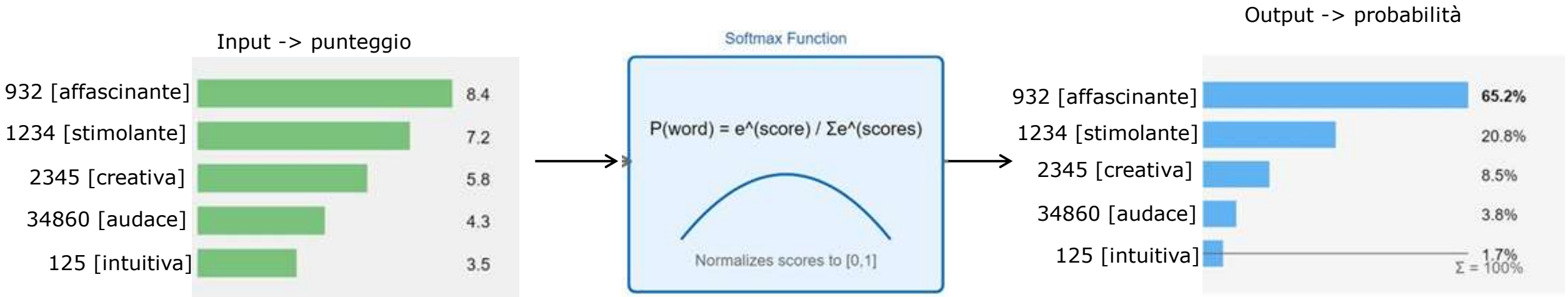


Vettore di «logits»
Punteggio per la prossima parola

Funzionamento del trasformatore

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

La parola con la probabilità più alta viene scelta come previsione



Funzionamento del trasformatore

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

-Token IDS

1349 1543 12456 218 932

-Decodifica del testo

L'intelligenza artificiale è affascinante

-Prossima parola prevista: affascinante (probabilità 65.2%)

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 3.5 turbo

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

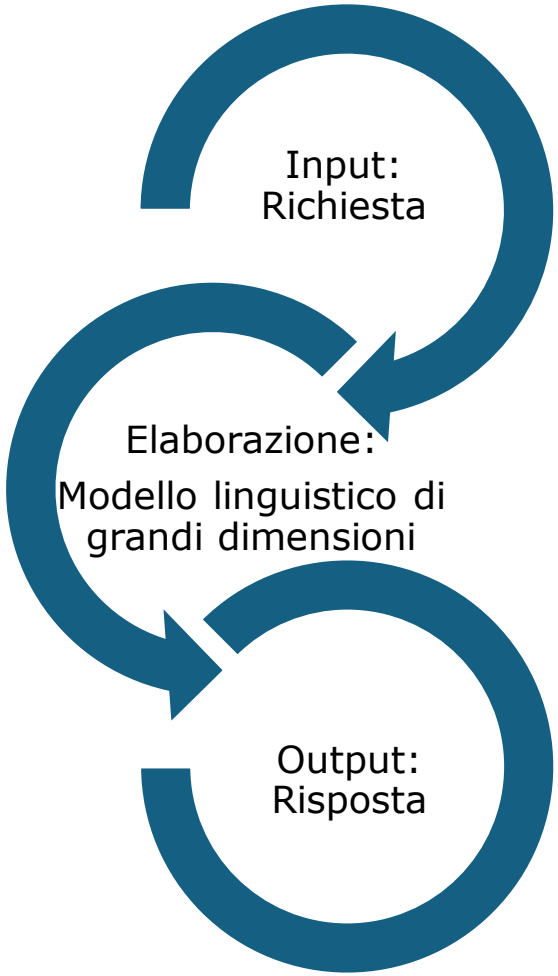
LMSYS Chatbot Arena Leaderboard

Questa classifica ordina i modelli linguistici in base alle prestazioni raggiunte in compiti conversazionali.
Chatbot Arena Leaderboard - a Hugging Face Space by lmarena-ai

Rank* (UB)	Rank (StyleCtrl)	Model	Arena Score	95% CI	Votes	Organization	License	Knowledge Cutoff
1	1	Gemini-2.5-Pro-Exp-03-25	1443	+8/-13	2540	Google	Proprietary	Unknown
2	4	Grok-3-Preview-02-24	1404	+5/-6	10398	xAI	Proprietary	Unknown
2	3	chocolate (Early Grok-3)	1402	+6/-6	13854	xAI	Proprietary	Unknown
2	2	GPT-4.5-Preview	1398	+7/-6	10615	OpenAI	Proprietary	Unknown
5	8	Gemini-2.0-Flash-Thinking-Exp-01-21	1381	+4/-3	22659	Google	Proprietary	Unknown
5	5	Gemini-2.0-Pro-Exp-02-05	1380	+5/-5	20293	Google	Proprietary	Unknown
5	3	ChatGPT-4o-latest (2025-01-29)	1374	+5/-4	22517	OpenAI	Proprietary	Unknown
8	6	DeepSeek-R1	1360	+5/-5	12772	DeepSeek	MIT	Unknown
8	13	Gemini-2.0-Flash-001	1355	+4/-4	18327	Google	Proprietary	Unknown
8	5	o1-2024-12-17	1351	+4/-4	25044	OpenAI	Proprietary	Unknown
11	13	Qwen2.5-Max	1340	+5/-3	17124	Alibaba	Proprietary	Unknown

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

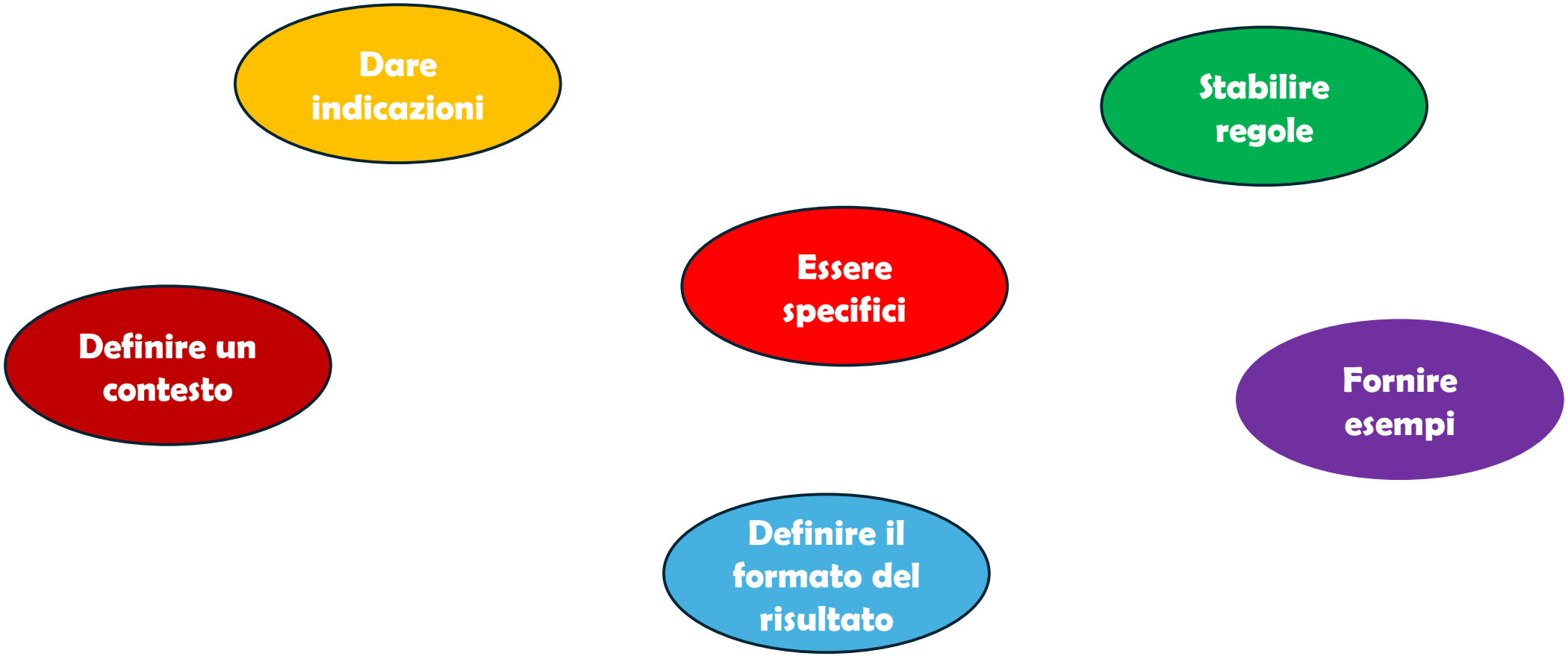
- Puoi scrivermi le condizioni di assicurazione di una polizza vita?

Richieste efficaci

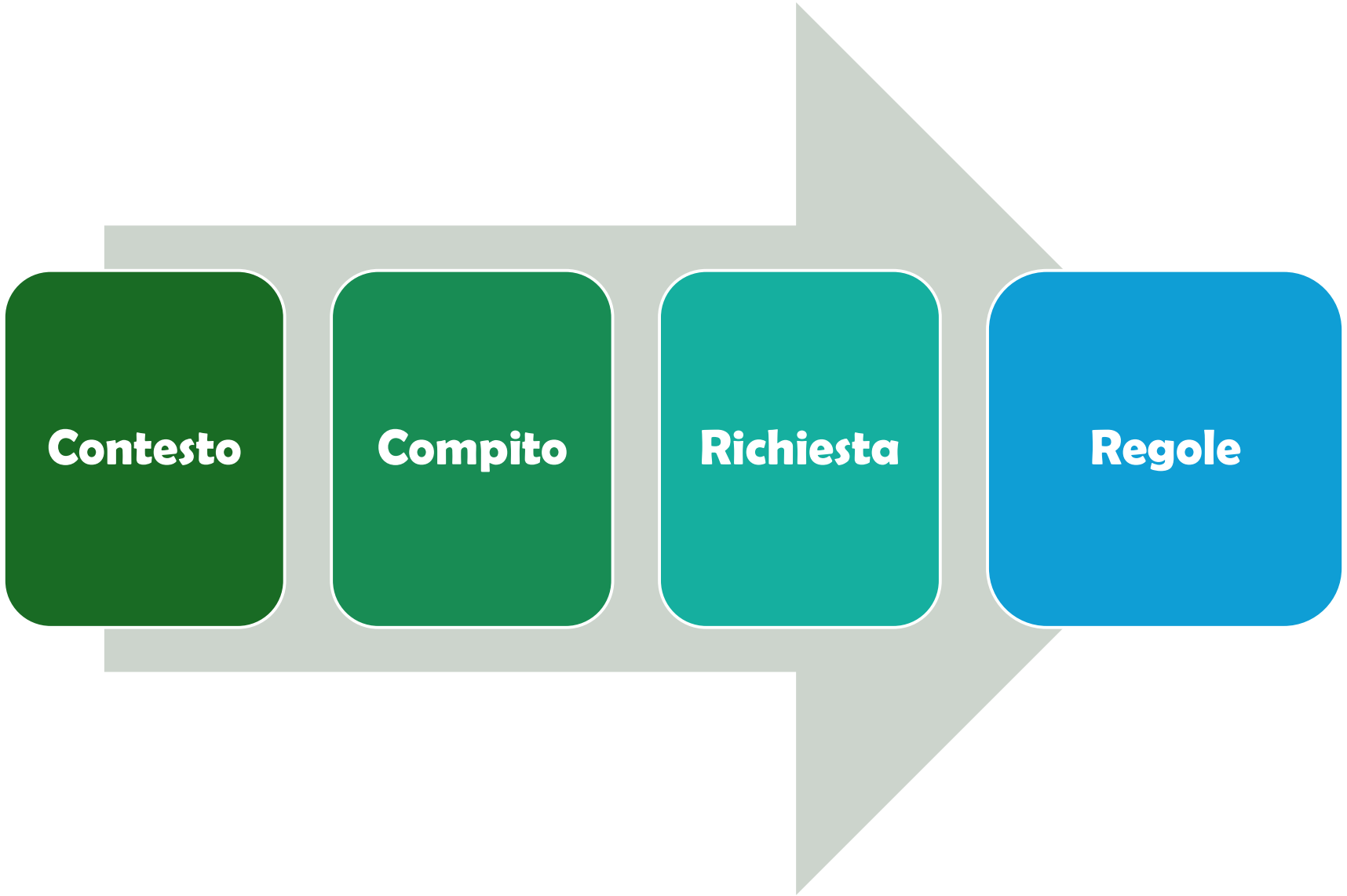
- Puoi scrivermi le condizioni di assicurazione di una polizza vita, temporanea caso morte a premio annuo e capitale costante, con tasso tecnico 1%, tavola Istat 2014 (maschi 60%, femmine 40%), durata 10 anni, capitale assicurato pari a 100.000 € e caricamenti del 15% .....?

Tecniche di ingegneria dei prompt

Ingredienti per una istruzione efficace



Flusso per una istruzione efficace



Flusso per una istruzione efficace nel dettaglio

- **Contesto:** stabilisce l'ambientazione per l'interazione con il modello linguistico. Include tutti i dettagli di sfondo, le impostazioni o i parametri che definiscono l'ambiente attorno al compito. Il contesto aiuta il modello a comprendere la rilevanza e il quadro del compito in questione.
- **Compito:** rappresenta l'obiettivo che si desidera venga raggiunto dal 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 e per minimizzare 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.

Tipologie di istruzioni

Zero-Shot

One-Shot

Few-Shot

Chain-of-Thought (CoT)

Tree-of-Thoughts (ToT)

ReAct (reason & act)

Automatic Prompt Engineering

Istruzioni senza esempi (Zero-Shot Prompting)

- La strategia zero-shot prevede che il modello linguistico di grandi dimensioni generi una risposta senza alcun esempio fornito dall'utente, basandosi esclusivamente sulla sua conoscenza pre-addestrata.

«Che competenze deve avere un attuario?»

Istruzioni con un esempio (One-Shot Prompting)

- La strategia one-shot prevede che il modello linguistico di grandi dimensioni generi una risposta a partire da un singolo esempio o pezzo di contesto fornito dall'utente.

Effettua la classificazione del testo fornito come disastro, oppure no disastro. Di seguito un esempio:
«13,000 people receive #wildfires evacuation orders in California.» Classificazione: disastro.

fonte

Testo: «#flood #disaster Heavy rain causes flash flooding of streets in Manitou, Colorado Springs areas»

Istruzioni con pochi esempi (Few-Shot Prompting)

- La strategia few-shot prevede che il modello linguistico di grandi dimensioni generi una risposta a partire da diversi esempi (con input e output) o pezzi di contesto. In questo modo, il modello può comprendere meglio l'intenzione umana e i criteri per generare risposte accurate.

Effettua la classificazione del testo fornito come disastro, oppure no disastro. Di seguito due esempi:
«Damage to school bus on 80 in multi car crash #BREAKING» Classificazione: disastro.

fonte

«I love fruits» Classificazione: no disastro.

Testo: «#Flood in Bago Myanmar #We arrived Bago»

Chain-of-Thought

- La strategia chain-of-thought prevede che il modello linguistico di grandi dimensioni generi dei passaggi intermedi di ragionamento. Questo processo aiuta il modello linguistico di grandi dimensioni a giungere a risposte più accurate, in particolare per compiti complessi che richiedono un ragionamento preliminare prima della formulazione della risposta.

«Analizza gli impatti di una nuova normativa su un portafoglio assicurativo. Procedi passo dopo passo, considerando prima i requisiti principali della normativa, poi identificando i prodotti coinvolti, le modifiche necessarie nelle politiche di underwriting, e concludi stimando gli effetti sui premi e sulle riserve tecniche. Proponi strategie per adattare il portafoglio»

Tree-of-Thoughts

- La strategia tree-of-thoughts prevede che il modello linguistico di grandi dimensioni generi dei passaggi intermedi di ragionamento. A differenza del CoT, che solitamente segue una singola catena lineare di ragionamento, il ToT consente ai modelli linguistici di esplorare simultaneamente molteplici percorsi di ragionamento.

«Valuta se introdurre un nuovo prodotto assicurativo. Esamina diverse possibilità del problema: (1) la profittabilità; (2) i rischi associati, con analisi dei rischi emergenti e delle loro mitigazioni; (3) l'aderenza normativa, valutando conformità legale. Fornisci una conclusione comparativa basata sulle tue analisi.»

Tree-of-Thoughts

- La strategia tree-of-thoughts prevede che il modello linguistico di grandi dimensioni generi dei passaggi intermedi di ragionamento. A differenza del CoT, che solitamente segue una singola catena lineare di ragionamento, il ToT consente ai modelli linguistici di esplorare simultaneamente molteplici percorsi di ragionamento.

«Valuta se introdurre un nuovo prodotto assicurativo. Esamina diverse possibilità del problema: (1) la profittabilità; (2) i rischi associati, con analisi dei rischi emergenti e delle loro mitigazioni; (3) l'aderenza normativa, valutando conformità legale. Fornisci una conclusione comparativa basata sulle tue analisi.»

ReAct (reason & act)

- La strategia ReAct consente al modello linguistico di grandi dimensioni non solo di ragionare su un compito, ma anche di compiere azioni per raccogliere le informazioni necessarie tramite strumenti esterni, imitando un processo di risoluzione dei problemi più interattivo e informato, simile a quello umano.

«Ottimizza il mix di investimenti di un fondo pensionistico. Alterna ragionamento e azione: identifica un obiettivo principale per il portafoglio, esplora possibili strumenti finanziari, costruisci un portafoglio ipotetico, simula il suo andamento sotto diversi scenari economici e rivedilo. Fornisci una soluzione iterativa e ben motivata.»

Automatic Prompt Engineering (APE)

- La strategia APE è un metodo che mira ad automatizzare il processo di creazione di prompt, spesso complesso e iterativo, sfruttando le capacità degli stessi modelli linguistici di generare e valutare potenziali prompt.

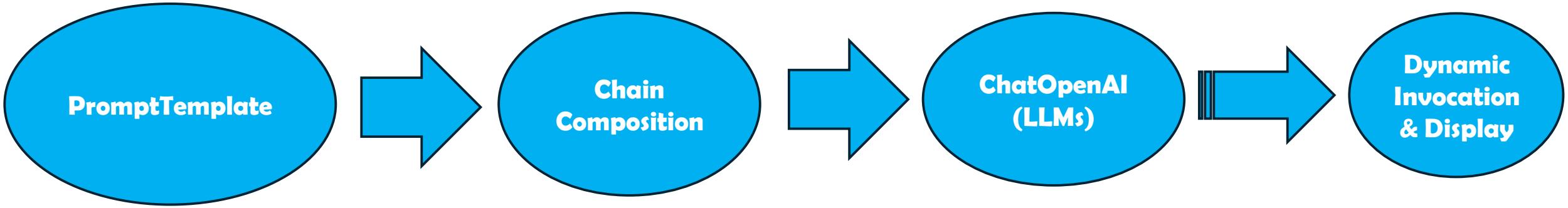
«Valutare l'adeguatezza delle riserve tecniche in un portafoglio assicurativo e segnalare aree di rischio potenziale.
Il modello potrebbe generare prompt intermedi come:
-Quali sono i parametri di riferimento per valutare l'adeguatezza delle riserve?
-Quali dati disponibili sono rilevanti per il calcolo?
-Quali scenari di rischio potenziale devono essere simulati?»

Ingegneria dei prompt applicata alla programmazione a supporto degli attuari

IA generativa come supporto nella programmazione

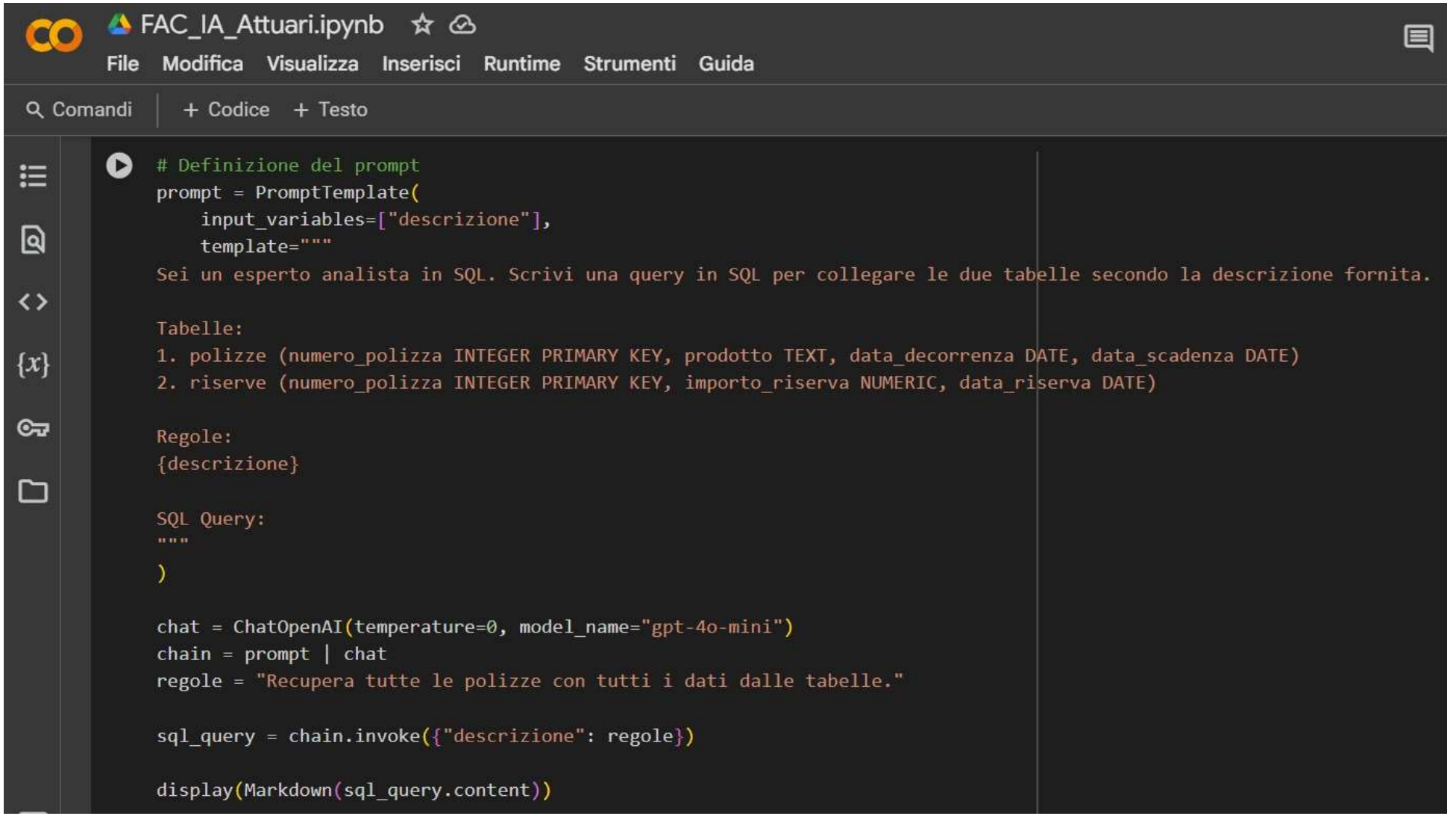
L'IA generativa rappresenta uno strumento potente per gli attuari coinvolti nella programmazione, offrendo opportunità per aumentare la produttività, migliorare le competenze di codifica, perfezionare la documentazione e semplificare diverse fasi del ciclo di vita dello sviluppo software nel settore attuariale.
L'IA generativa deve essere vista come un potenziamento delle competenze attuariali e non come un sostituto. Gli attuari devono continuare a fornire guida, interpretare i risultati e applicare il loro giudizio professionale. Inoltre, la qualità dei risultati dipende dalla qualità dei prompt e dei dati di addestramento.

Componenti di LangChain coinvolti nell'ingegneria delle richieste



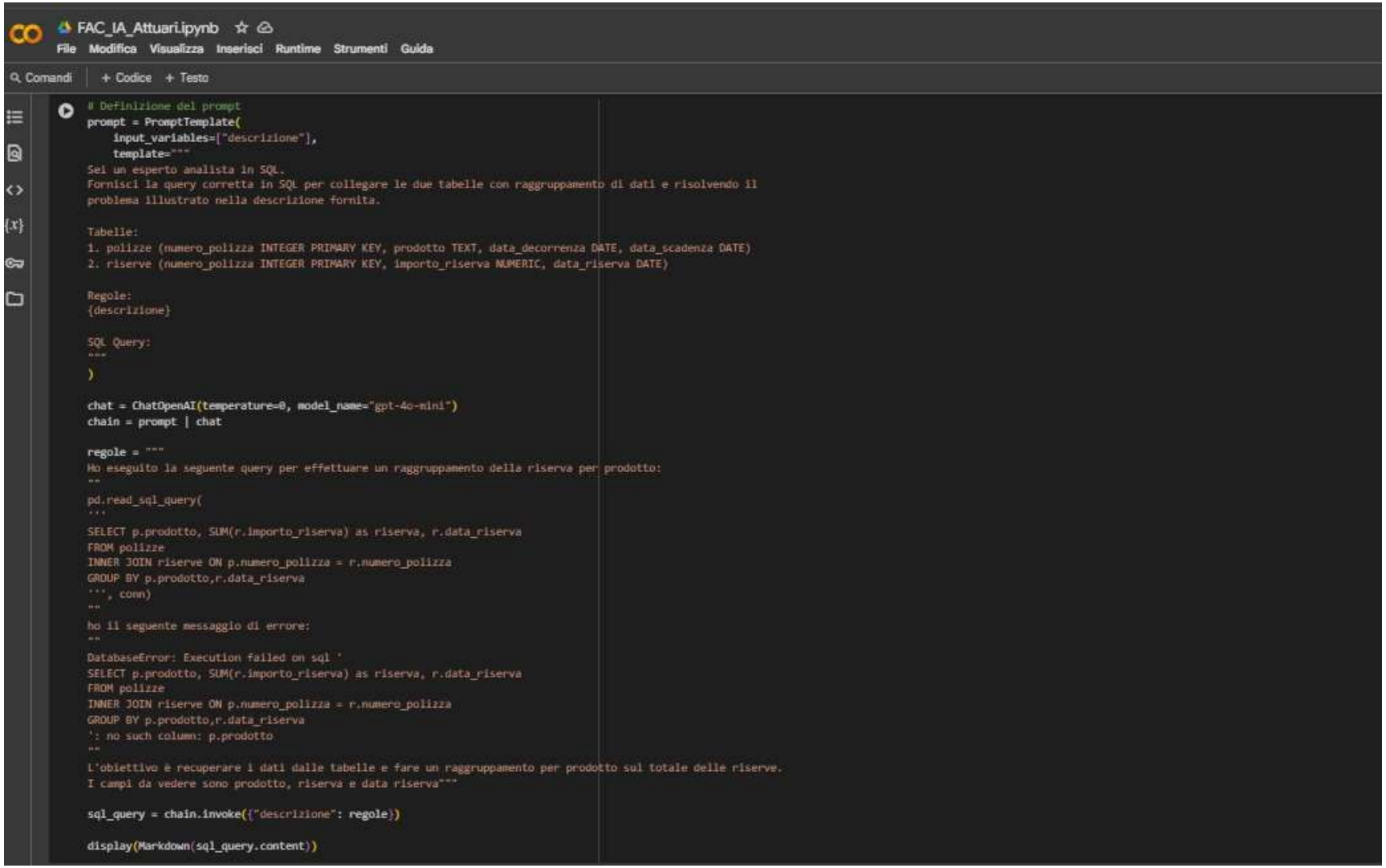
Prompt Engineering per la programmazione

Utilizzare i prompt per guidare i modelli linguistici nella generazione di codice in specifici linguaggi di programmazione



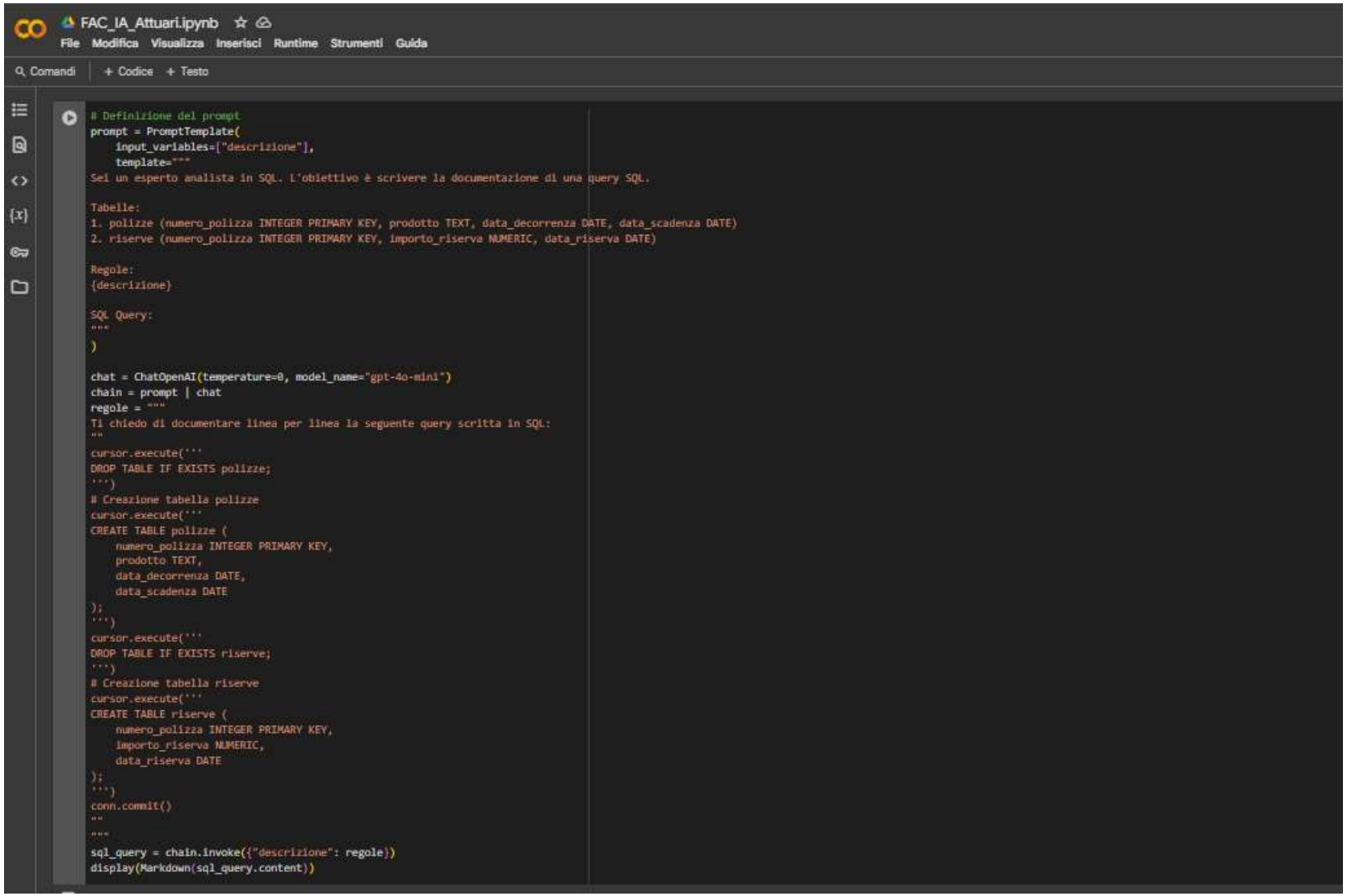
Prompt Engineering per il debugging

Il modello linguistico può suggerire correzioni immediatamente relative all'errore riscontrato e proporre miglioramenti per una maggiore robustezza e leggibilità



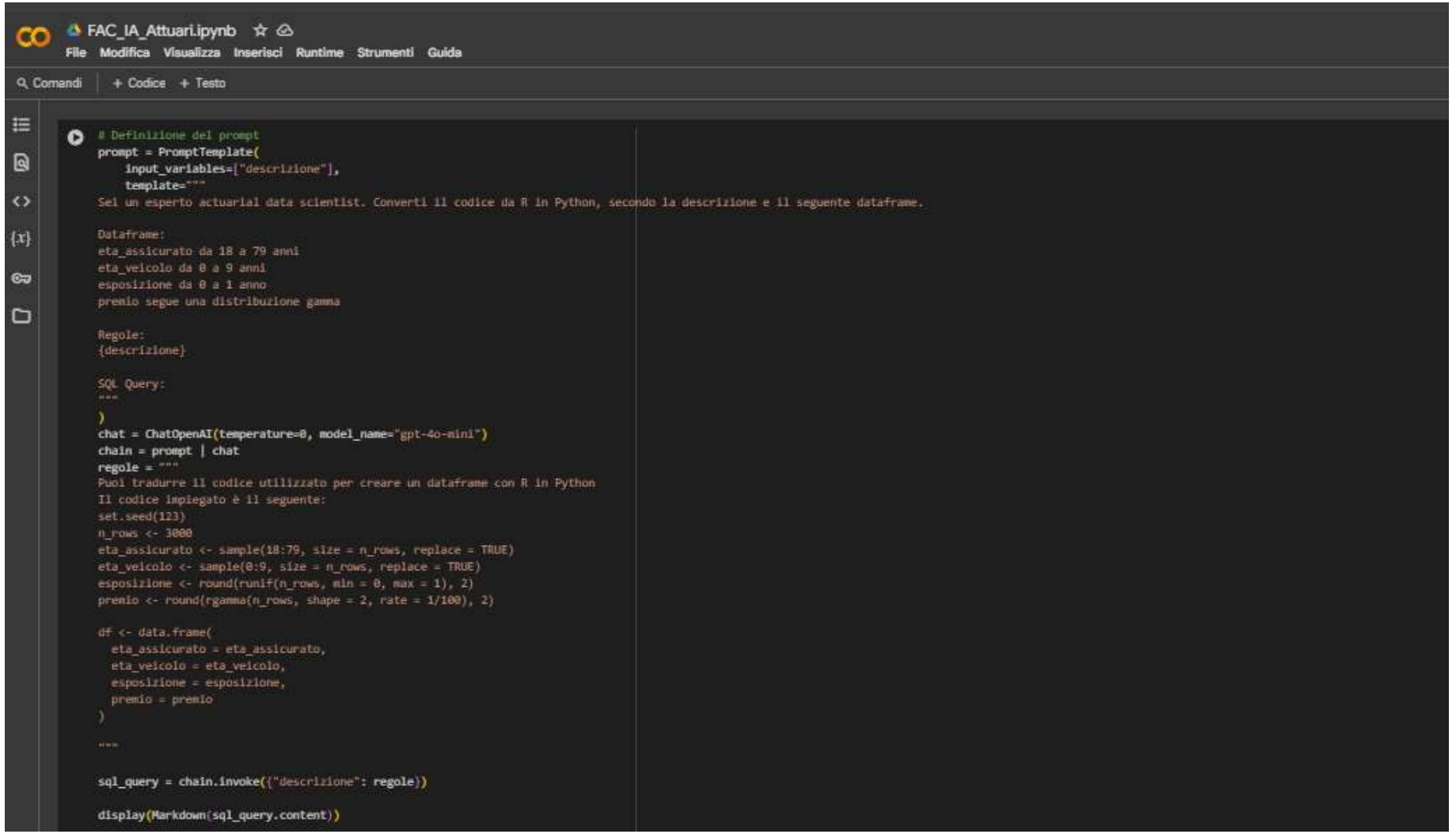
Prompt Engineering per la documentazione

Porre domande sui frammenti di codice per ricevere una spiegazione dettagliata delle funzioni e del flusso.



Prompt Engineering per la traduzione

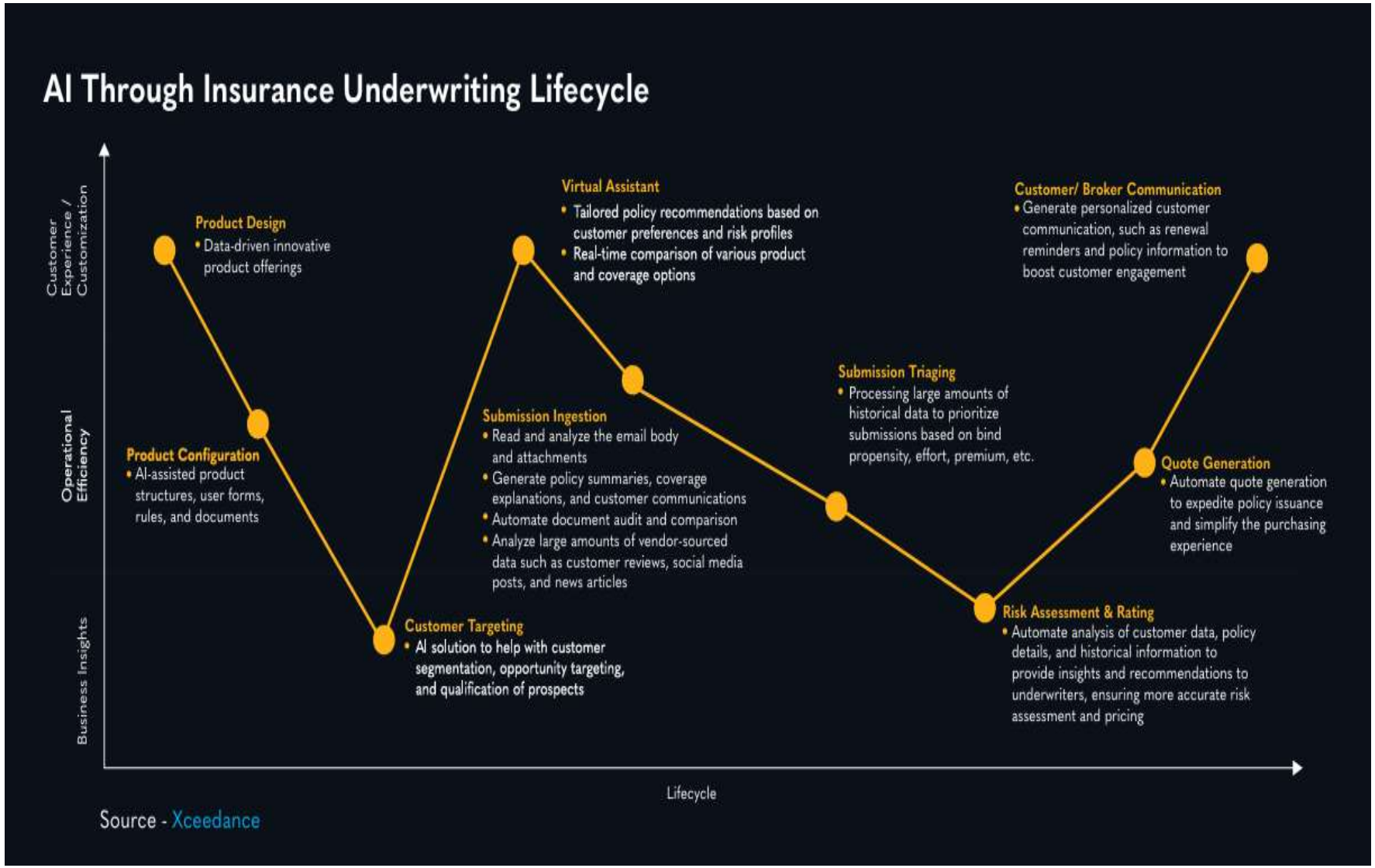
Usare i prompt per convertire il codice da un linguaggio di programmazione a un altro.



Prospettive future

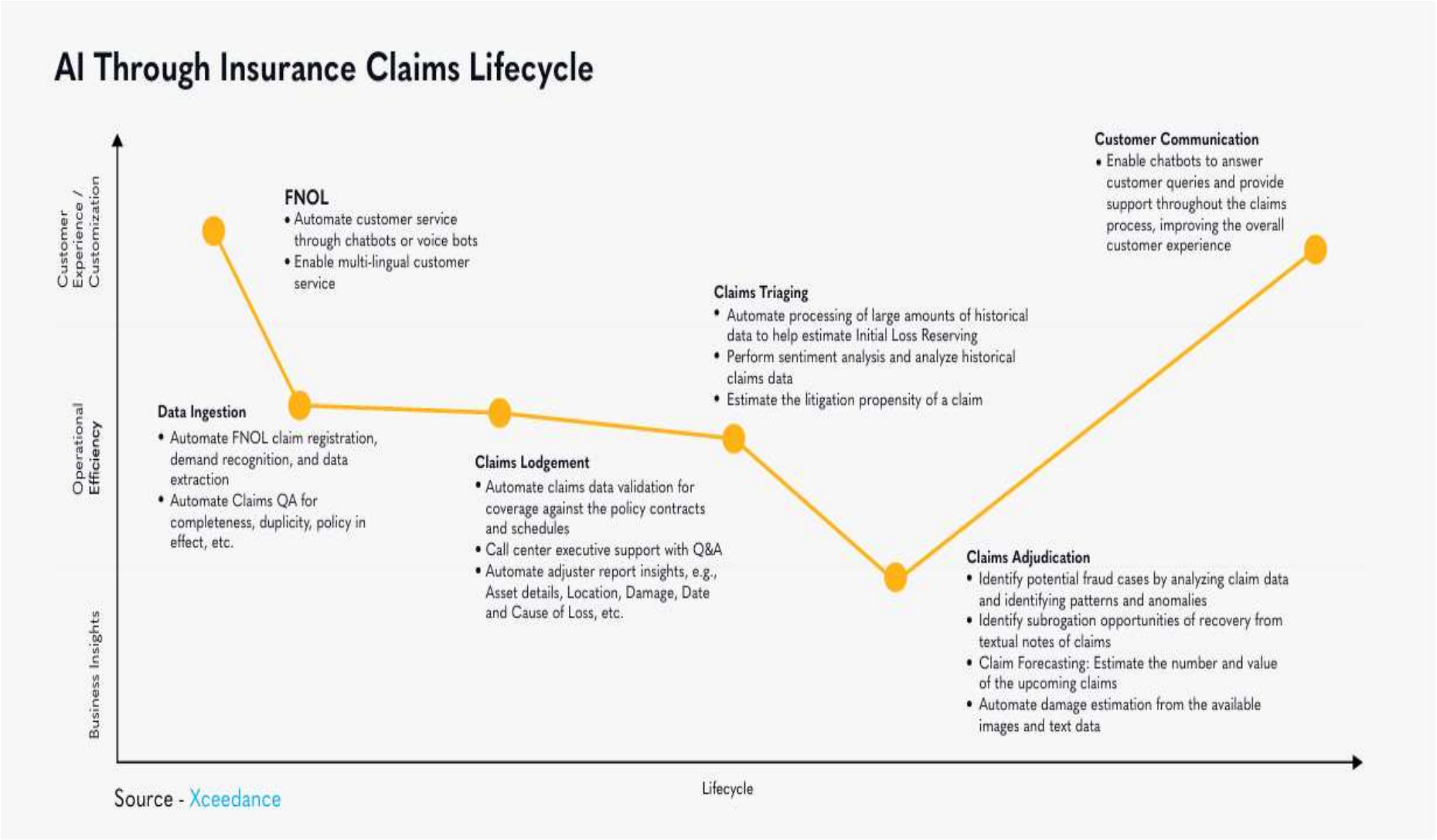
Impatti IA Generativa nel ciclo di vita della sottoscrizione

Si abbraccia l'uso dell'IA Generativa per migliorare le efficienze operative, migliorare l'esperienza del cliente e fornire efficienza operativa in ogni fase del processo di sottoscrizione. Questo include l'automazione dei compiti, l'analisi dei dati per un migliore processo decisionale, la personalizzazione delle interazioni con i clienti e, in definitiva, mira a una valutazione del rischio più accurata e a un'emissione delle polizze più rapida.



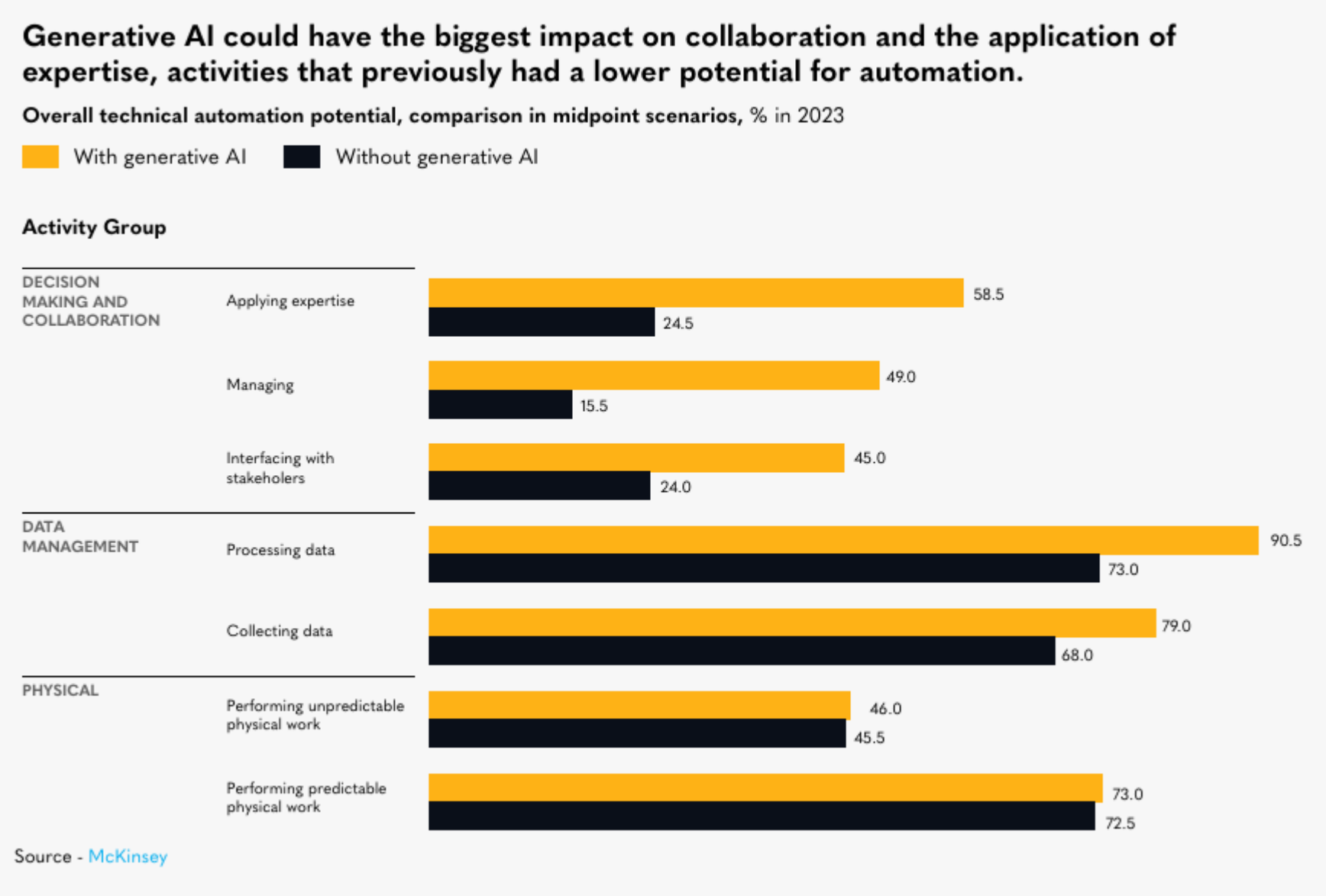
Impatti IA Generativa nel ciclo di vita dei sinistri

L'IA Generativa grazie alla sua capacità di comprendere e generare dati testuali complessi contribuisce all'estrazione automatica di dati, la convalida dei sinistri, l'identificazione delle frodi e la migliore comunicazione con i clienti tramite chatbot sofisticati.



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

-Introduzione all’intelligenza artificiale generativa

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

AI Tools for Actuaries by Mario V. Wuthrich, Ronald Richman, Benjamin Avanzi, Mathias Lindholm, Michael Mayer, Jürg Schelldorfer, Salvatore Scognamiglio :: SSRN

A comprehensive survey and analysis of generative models in machine learning – ScienceDirect

A Primer on Generative AI for Actuaries | SOA

Jobs in artificial intelligence (AI) - Intuit Blog

Statistical Modeling: The Two Cultures

-I modelli linguistici e l'ingegneria dei prompt

[1706.03762] Attention Is All You Need

[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

Guida all'ingegneria dei prompt per l'AI | Google Cloud

-Tecniche di ingegneria dei prompt

Best practices for prompt engineering with the OpenAI API | OpenAI Help Center-

Prompt engineering - OpenAI API

[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)
Prompt Engineering | Kaggle
Prompt Engineering: The Emerging Art of Coding or The Future of Human-Machine Communication?

-Ingegneria dei prompt applicata alla programmazione a supporto degli attuari
A Primer on Generative AI for Actuaries | SOA

-Prospettive future
A Primer on Generative AI for Actuaries | SOA
Generative-AI-in-Insurance-A-Game-Changer.pdf

-Notebook
FAC IA Attuari

Risorse Utili

API keys - OpenAI API
How to use Secrets in Google Colab Google Colab
Tutorial - Colab

Grazie

Contatti:
Linkedin
medium