



Intelligenza artificiale – Corso base sull'Intelligenza Artificiale

***Analisi di contesto, struttura delle base informativa e strumenti di
prototipizzazione rapida***

Dott. Ing. Valerio Morfino
Ordine degli Ingegneri della Provincia di Roma

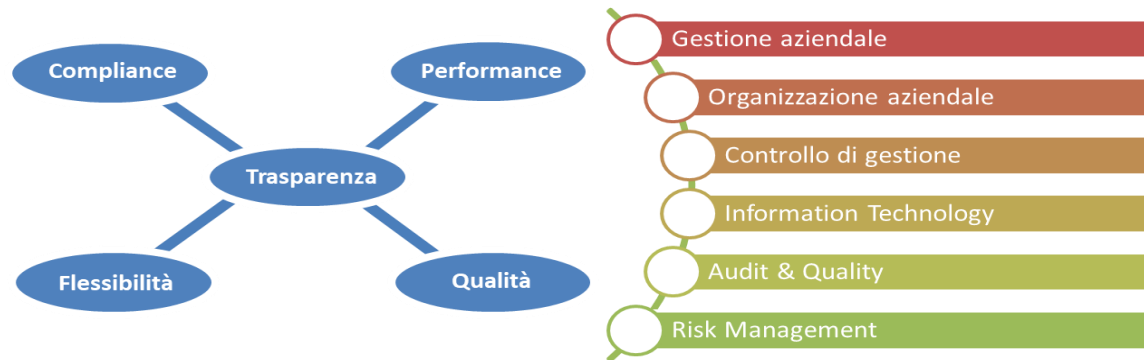
valerio.morfino75@gmail.com

7 Febbraio 2026



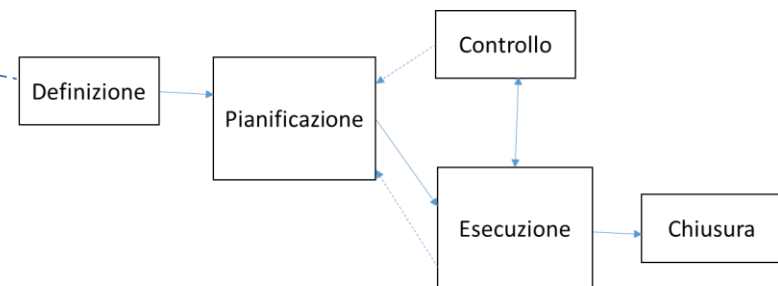
Analisi di Contesto – Approccio olistico ai progetti AI

Ottimizzazione di Processo – Cinque Dimensioni



E' fondamentale che ogni **"Innovazione"** introdotta in un Sistema venga preceduta da un'accurata **Analisi** (processo, tecnologica, normativa, ecc..) e da una adeguata **Pianificazione**, coinvolgendo tutti gli attori interessati al cambiamento (Shareholders e Stakeholders), ognuno per proprio ruolo e competenza.

Per il buon governo del cambiamento è necessario individuare e formalizzare opportunamente le necessità, i benefici e gli obiettivi verso cui indirizzare i programmi di innovazione, ottenendo il *commitment* di tutti gli *Shareholders* e *Stakeholder* coinvolti.



Per supportare il coinvolgimento di tutti gli attori, può essere utile fare ricorso a: **Comunicare con il giusto linguaggio con ogni stakeholder, Proof of Concept (POC), Prototipi, MVP.**



Cos'è l'Intelligenza Artificiale?

L'**intelligenza artificiale** è un insieme di tecnologie differenti che interagiscono per consentire alle macchine di percepire, comprendere, agire e apprendere con livelli di intelligenza «simili» a quelli umani.

AI generale o «forte»

- L'AI generale è ciò che si vede nei film di fantascienza, dove **macchine senzienti emulano l'intelligenza umana**, pensando in modo strategico, astratto e creativo, con la capacità di gestire una serie di compiti complessi.
- Al momento **non abbiamo ancora sviluppato un'intelligenza artificiale di questo tipo.**

AI ristretta o «debole»

- La **maggior parte delle applicazioni AI** che sperimentiamo nella vita quotidiana rientrano nell'AI ristretta.
- **Esegue un singolo compito o una serie di attività strettamente correlate.**
- Sono sistemi potenti, ma **l'ambito di azione è limitato.**
- Con la giusta applicazione, l'AI ristretta ha un **immenso potenziale di trasformazione** e continua a influenzare su scala globale il modo in cui viviamo.
- In questa categoria rientra il **Machine Learning** e la **GenAI**.



Machine Learning Supervisionato

Che lo si faccia con algoritmi bayesiani, alberi decisionali o deep learning, il machine learning è qualcosa che, se non compreso bene, può portare a diverse interpretazioni (spesso fantasiose) o ad aspettative fuori dalla fattibilità.

Velocità	Azione
50	Frena
40	Accelera
30	Accelera
100	Frena

Velocità	Azione
10	?

Quale azione dovrà essere intrapresa?



Training set	Velocità	Azione
	50	Frena
	40	Accelera
	30	Accelera
New Data	100	Frena

Trained Model: Decision tree	If Velocità \leq 40 \rightarrow Accelera Else \rightarrow Frena
---------------------------------	--

New Data	Velocità	Azione
	10	?

- Il Machine learning supervisionato ricava un modello dai dati di addestramento (training set)
- Il modello potrà essere applicato su nuovi dati per fare predizioni
- Il modello può essere creato con diversi algoritmi (deep learning, bayesiani, alberi, ensemble, ecc.)



Training set	Velocità	Azione
	50	Frena
	40	Accelera
	30	Accelera
New Data	100	Frena
	Trained Model: Decision tree <div>If Velocità \leq 40 \rightarrow Accelera Else \rightarrow Frena</div>	
	Velocità	Azione
	10	?

- In base all'algoritmo scelto, potrebbe essere possibile avere un modello di funzionamento esplicito.
- In molti casi (es. reti neurali) il modello non è facilmente comprensibile per l'uomo
- E' possibile ricorrere a strumenti di «explainability»



Machine Learning Supervisionato

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	Azione
12	44	A	1	Y	0	1	0.2	Y	100	0	C	11	Accelera
22	43	A	1	Y	1	1	0.33	N	200	0	C	11	Accelera
11	23	B	1	Y	1	2	0.3	N	150	0	C	12	Frena
12	43	A	1	N	1	2	0.4	N	100	0	C	13	Accelera
3	33	C	1	Y	1	1	0.1	N	150	0	C	13	Frena
2	54	B	2	Y	0	3	0.11	N	300	0	C	13	Frena
32	54	C	2	N	0	3	0.12	N	300	0	C	11	Accelera
12	33	B	3	N	0	3	0.12	N	100	1	A	12	Frena
3	45	C	1	Y	1	2	0.11	N	150	1	A	13	Accelera
2	56	A	2	Y	1	2	0.2	Y	150	0	A	12	Frena

- Anche in caso di modelli espliciti (es. Decision Tree) può essere difficile comprendere il modello esplicito
- Anche in questi casi, possono essere utili tecniche di explainability

Machine Learning Non Supervisionato

A volte non è immediato capire come può aiutare e come funziona, eppure talvolta può essere di grande utilità ed efficacia.



sample



Cluster/group

Suddivisione di un gruppo di informazioni in categorie omogenee, calcolate in base a distanze matematiche

<https://www.guru99.com/unsupervised-machine-learning.html>

Come può essere utilizzato?



sample



Cluster/group

- Suddividendo i contribuenti in classi omogenee... si possono scoprire frodi, anomalie, evasioni fiscali
- Identificazione pratiche o produzioni con anomalie
- Classi di client o utenti con caratteristiche simili
- Identificazioni di criteri di similitudine non evidenti

Nota:

Il colore è solo una delle dimensioni di analisi. Nel caso specifico, oltre al colore potremmo avere: grandezza, peso, costo, provenienza, qualità, stato di conservazione, operatore,...

Che cos'è l'AI Generativa?

Gli strumenti software di Generative AI creano contenuti (es. immagini, testi, stringhe di codice, video, musica) a partire da uno specifico tipo di prompt utilizzando algoritmi addestrati su un set di dati selezionati. Ad esempio:



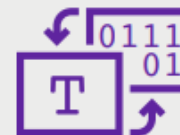
Generatori text-to-text

(es. OpenAI ChatGPT, Google Bard)



Generatori text-to-image

(es. Gemini Nano Banana, DALL-E, Stability AI, Midjourney)



Generatori text-to-code

(es. GitHub Copilot, Claude, ChatGPT)

E l'attualità ci racconta come siamo già oltre: nell'ultima versione, ChatGPT potrà vedere, leggere e parlare



Che cos'è un LLM – Large Language Model

Un **Large Language Model (LLM)** è un sistema di intelligenza artificiale avanzato, progettato per comprendere, generare e manipolare il linguaggio umano in forma scritta. Viene chiamato "grande" perché è addestrato su miliardi di parole e testi, coprendo una vasta gamma di argomenti, stili e contesti.

01

Reti Neurali

Gli LLM usano reti neurali ispirate al cervello umano, in particolare i transformer, capaci di 'comprendere' il significato delle parole analizzando il contesto in cui sono inserite.

02

Capacità Predittive

Questo consente al modello di rispondere a domande, sintetizzare testi lunghi e persino simulare conversazioni coerenti, più in generale di compiere elaborazioni testuali complesse.

03

Apprendimento

Un LLM viene "addestrato" su grandi raccolte di testi: libri, articoli, siti web, documenti pubblici, ecc. Non impara regole fisse, ma identifica schemi ricorrenti nel linguaggio. A differenza di una banca dati, non si limita a recuperare informazioni, ma è in grado di costruire risposte nuove partendo dalle conoscenze apprese.

04

Modello probabilistico

A differenza di un programma informatico classico, che esegue istruzioni predefinite, un LLM è probabilistico e adattivo: genera contenuti basandosi su modelli linguistici, non su regole rigide. Questo lo rende flessibile, ma anche meno deterministico nei risultati.

05

Alcuni LLM

- GPT di OpenAI,
- Claude di Anthropic,
- LLAMA di Meta,
- DeepSeek V3
- Minerva della Sapienza



Transfer Learning

Il **Transfer Learning** è la tecnica alla base della GenAI, che consente di utilizzare un modello LLM preaddestrato su un set molto ampio di informazioni, per eseguire compiti per i quali il modello non è stato specificamente addestrato.

Come funziona?

Un neolaureato ha studiato molti argomenti all'università, tuttavia, entrato nel mondo del lavoro, è consapevole di dover ancora imparare molto. Tuttavia, impara molto più velocemente: con una formazione breve o un affiancamento è in grado di svolgere compiti di crescente complessità. Durante il percorso di studi, ha imparato a imparare!

Analogamente, un **LLM addestrato su grandissimi quantità di dati** è in grado con poco o nessun adattamento di **svolgere compiti specifici complessi**.

Perché è utile?

Perché permette di sviluppare applicazioni AI complesse con limitata necessità di dati e risorse computazionali di addestramento, abilitando casi d'utilizzo avanzati in molti campi quali medicina, industria, finanza, formazione e tanto altro.

Terminologia essenziale della GenAI

LLM



Un **Large Language Model (LLM)** è un modello di grandi dimensioni addestrato su grandi quantità di dati testuali per comprendere e generare linguaggio naturale. I modelli multimodali, possono anche elaborare immagini, video e audio, oltre al testo. Sono alla base di applicazioni come ChatGPT, generatori di immagini (es. DALL·E) e assistenti AI evoluti.

Il **Prompt** è al testo di input che viene fornito a un modello. Esso definisce la natura della richiesta e orienta il modello nella generazione della risposta. Con il **prompt engineering** il modello viene addestrato per rispondere a compiti specifici.



Prompt

L'**Embedding** è una tecnica che consente una rappresentazione numerica del testo (vettorizzazione) per consentire al sistema di manipolarlo e comprenderlo in maniera più efficiente conservando la semantica e la sintassi del concetto espresso.



Embedding

Vector
Store



Il **Vector Store** è il contenitore in cui vengono storicizzate le rappresentazioni numeriche dei testi (vettori). Questa rappresentazione vettoriale permette al sistema di elaborare, confrontare e ricercare il testo in modi che sarebbero complessi utilizzando il testo nella sua forma originale.



LRM – Large Reasoning Model

Un **Large Reasoning Model (LRM)** è un'evoluzione dei modelli linguistici (LLM) che, oltre a comprendere e generare testo, è progettato per simulare processi di ragionamento esplicito. Questi modelli non si limitano a fornire una risposta, ma mostrano i passaggi logici che conducono a una conclusione.

- Utilizzano tecniche come il Chain-of-Thought, in cui il modello esplicita il proprio percorso logico passo dopo passo. Questo li rende capaci di affrontare problemi complessi rendendo visibili le inferenze intermedie e facilitando la verifica del ragionamento.
- Modelli come GPT-o1 o DeepSeek R1 rientrano tra gli LRM: mostrano una capacità avanzata di analizzare un problema, strutturare un ragionamento e fornire una risposta motivata, piuttosto che semplicemente plausibile.

Esempio: Log di Ragionamento di un LRM

Domanda

Un contratto prevede una clausola penale per inadempimento, ma il creditore ha comunque accettato la prestazione in ritardo senza riserve. Può ancora pretendere la penale?

Ragionamento esplicito (Chain-of-Thought)

1. La clausola penale è una previsione contrattuale che stabilisce una somma dovuta in caso di inadempimento o ritardo.
2. Il creditore ha accettato la prestazione **in ritardo**, ma **non ha espresso riserve** nel momento dell'adempimento.
3. Ai sensi dell'art. 1462 c.c. (diritto italiano), se il creditore accetta volontariamente l'adempimento tardivo **senza riserve**, si presume che **abbia rinunciato alla penale**.
4. Quindi, anche se la clausola è formalmente valida, il **comportamento successivo del creditore ha effetti modificativi**.
5. Il modello riconosce qui una **rinuncia tacita**, fondata sul comportamento concludente.

Risposta finale

No, se il creditore ha accettato la prestazione in ritardo **senza formulare riserve**, **non può più esigere la penale**, in quanto ha **rinunciato tacitamente** a farla valere.

Allucinazioni 1/2

- Si parla di allucinazioni quando un modello genera un contenuto (es. testo, immagine, video) che contiene informazioni false o fuorvianti, pur sembrando plausibili
- La GenAI "allucina" quando inventa cose che non sono vere, ma verosimili.
- Immagini fotografiche apparentemente realistiche, ma che possono contenere elementi irreali nei particolari (es. Volto, scritte sul giornale, prospettiva della panchina).
- Testi apparentemente validi, comprensivi anche di argomentazioni, ma in realtà falsi.
- Traduzioni che stravolgono il senso del testo da tradurre.



Generata da ChatGPT, prompt: "uomo seduto al parco che parla al cellulare con un giornale in mano"



Perché si generano le allucinazioni?

- **Mancanza di dati:** LLM addestrato su un set di dati incompleto o di bassa qualità.
- **Pregiudizi nei dati:** Se i dati di addestramento contengono pregiudizi(*) o informazioni errate, l'LLM potrebbe replicarli nelle sue generazioni.
- **Modello:** Gli LLM possono essere imperfetti e generare output errati, anche con dati di alta qualità.

Le allucinazioni possono creare problemi:

- **Informazioni errate:** Le informazioni generate possono essere errate (es. assistente virtuale che fornisce info errate ad un turista, ad un medico, ad un cliente).
- **Disinformazione:** Le informazioni false generate dalla GenAI possono essere diffuse online e ingannare le persone.
- **Sfiducia:** Se le persone non possono fidarsi dell'output della GenAI, potrebbero essere meno propense a utilizzarla in futuro.

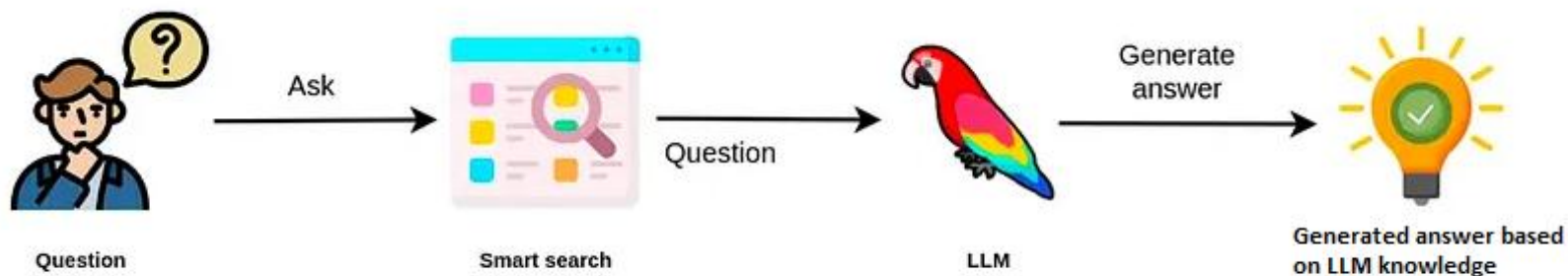
Consapevolezza dell'utente e RAG (Retrieval-Augmented Generation)

Questo approccio mitiga le allucinazioni nei testi generati con un **sistema di recupero di informazioni pertinenti da un corpus di testi verificato**, per guidare il processo di generazione in modo da produrre testi più coerenti e pertinenti.

In sostanza, l'uso del recupero delle informazioni consente al modello di avere una base di conoscenza più solida su cui basare la generazione del testo, riducendo così la probabilità di produrre allucinazioni.

Perché usare il RAG? - Cos'è l'Architettura RAG

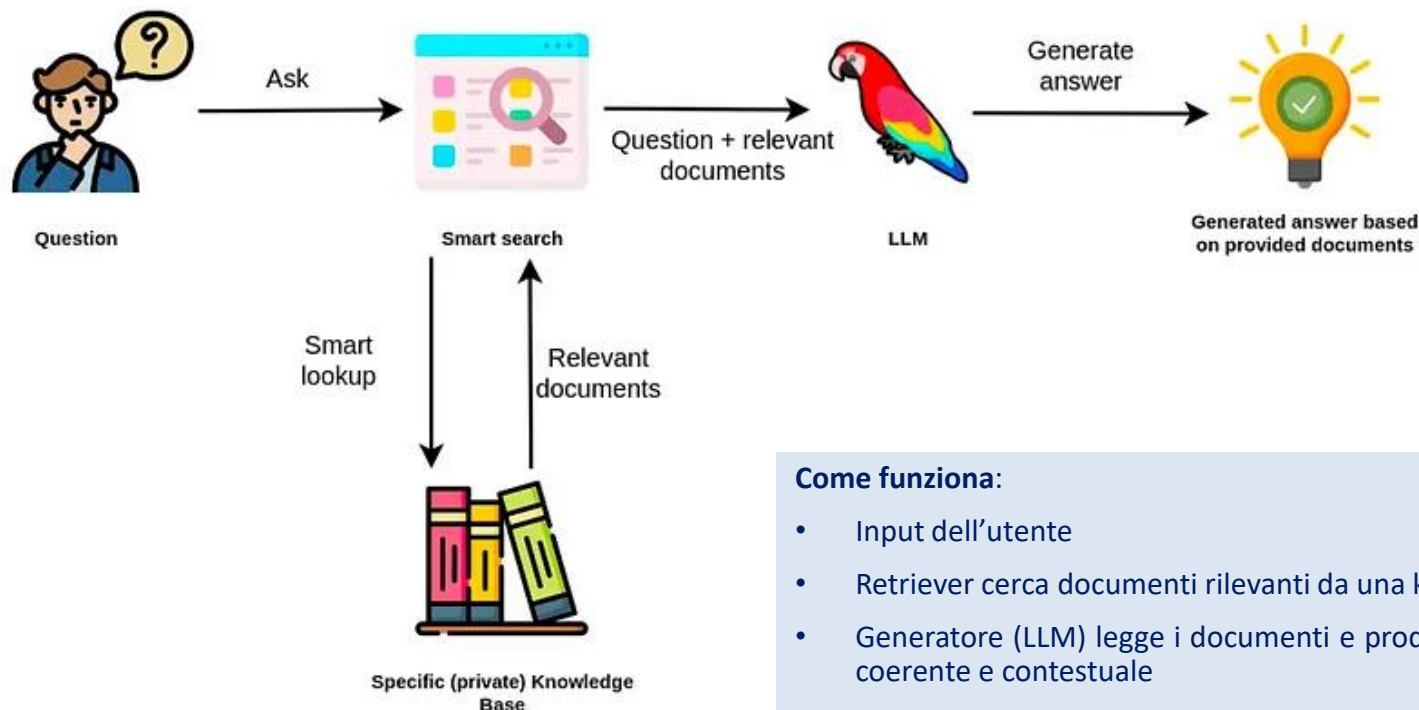
Architettura non RAG



**Quanto è aggiornata la risposta?
Su quale base di conoscenza risponde?**

Perché usare il RAG? - Cos'è l'Architettura RAG

Architettura LLM RAG



Come funziona:

- Input dell'utente
- Retriever cerca documenti rilevanti da una knowledge base
- Generatore (LLM) legge i documenti e produce una risposta coerente e contestuale

Obiettivo: Fornire risposte accurate, aggiornate e affidabili, anche su argomenti non inclusi nel training del modello



Perché usare il RAG?

- Fonti personalizzabili: permette di scegliere da quali documenti o knowledge base attingere, adattandosi al contesto.
- Maggiore trasparenza: le fonti utilizzate per generare la risposta sono note e certificate, utile per controllo e fiducia.
- Risposte più pertinenti: recuperando solo le informazioni rilevanti, migliora la qualità delle risposte in domande complesse.
- Meno allucinazioni: le risposte sono basate su documenti concreti, riducendo errori o invenzioni.
- Modularità Retriever e generatore sono separati e aggiornabili indipendentemente.



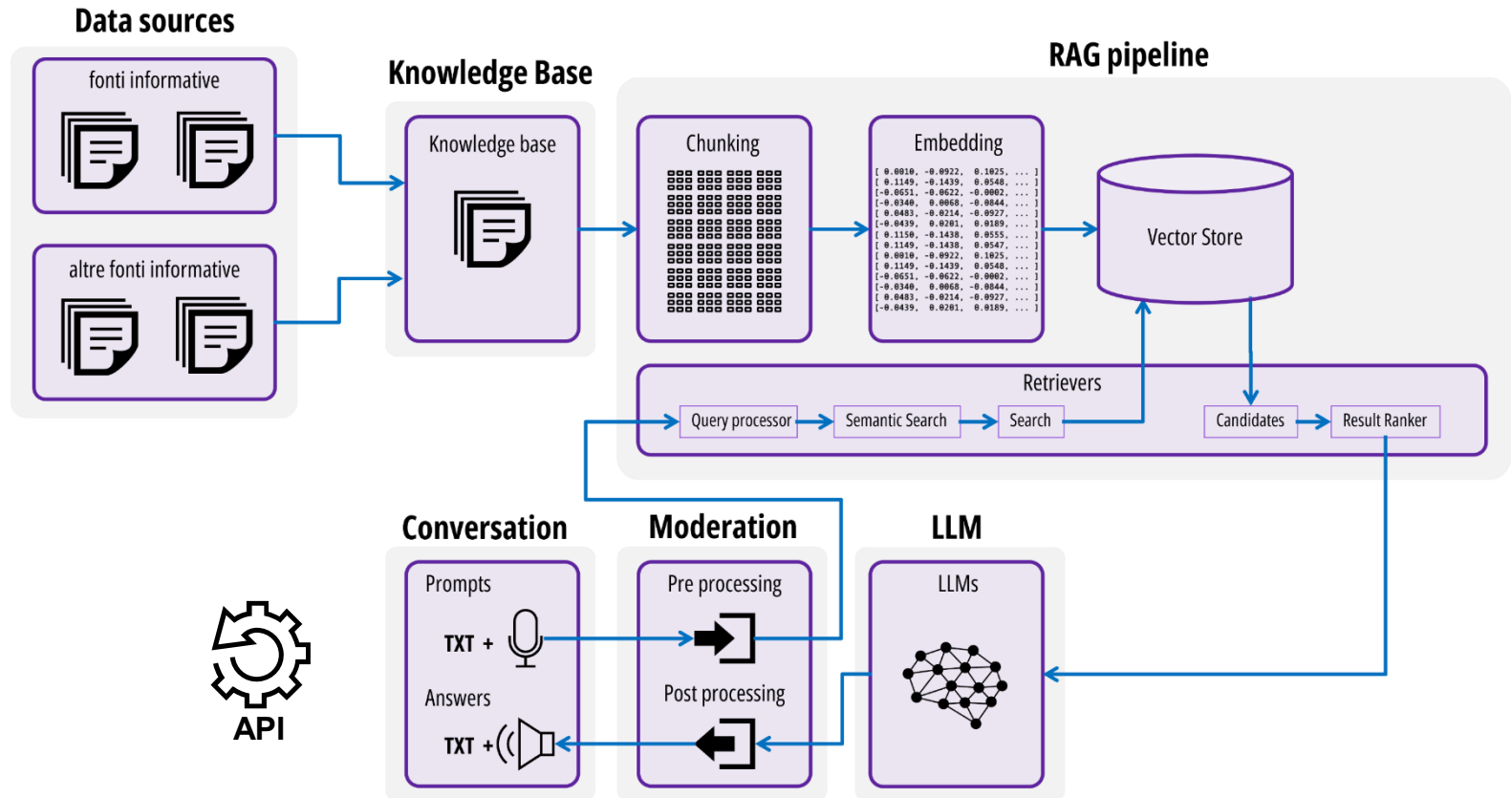
Perché usare il RAG?

- Un LLM è un modello addestrato su Miliardi di dati, che possiamo adattare a molti compiti grazie al transfer learning
- Architettura RAG è: LLM + Archivio dati da interrogare prima di rispondere
- Una metafora: “ Parlami del ruolo di Robespierre nella Rivoluzione Francese”
 - LLM senza RAG è come “rispondere a memoria”
 - LLM+RAG è come leggere da un libro, e poi rispondere

Chi è più affidabile?

Rag Augmented Applications

La **retrieval-augmented generation** è una tecnica che permette ai modelli linguistici di grandi dimensioni (LLM) di recuperare e incorporare informazioni nuove da fonti di dati esterne





La questione dei dati

I sistemi di intelligenza artificiale hanno bisogno di dati

- Per l'addestramento
- Per restituire un risultato (es. predizione)

Quando si sviluppa una POC, si tende:

- Ad avere un perimetro circoscritto
- A fare delle semplificazioni
- Ad avere un dataset ristretto e ben qualificato

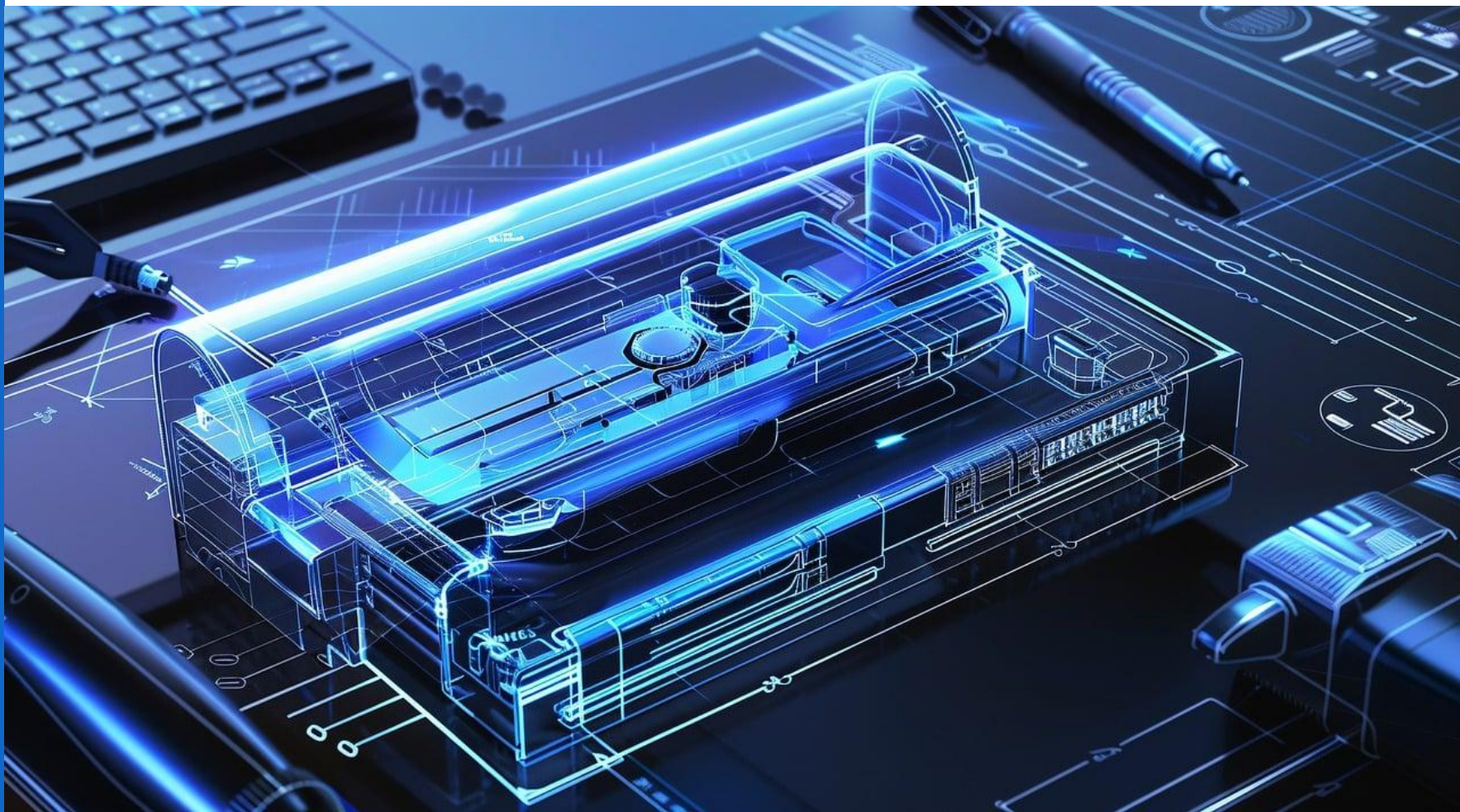
Bisogna essere consapevoli e comunicare ai nostri Stakeholders che in fase progettuale sarà necessario:

- Reperire dati di qualità per il training ed etichettarli
- Verificare il Bias
- Gestire il bilanciamento del Dataset
- Avere dati testuali in formato digitalizzato (per LLM). Nel caso di PDF potrà essere necessario OCR e preservare lo schema dei documenti
- Integrare i sistemi decisionali con processi e software già in uso e nella architetture

Questi aspetti della progettualità possono avere un impatto molto rilevante sui costi ed anche sulla fattibilità del progetto



Prototipi e POC



E strumenti per crearli



Ing. Valerio Morfino

Intelligenza artificiale – Comunicazione semplificata al cliente e strumenti di prototipazione rapida

7 Febbraio 2026

pag. 23



Perché un Prototipo o una POC?

1. **Per verificare i requisiti:** il prototipo consente di mitigare il rischio di fraintendimento dei requisiti e dei risultati attesi
2. **Per ridurre i rischi di progetto:** Il prototipo consente di identificare e mitigare potenziali problemi tecnici, etici o di sicurezza prima dell'implementazione completa.
3. **Per verificare la fattibilità :** Aiuta a dimostrare la fattibilità e l'efficacia dell'applicazione, garantendo che soddisfi realmente le esigenze degli utenti.
4. **Per ottimizzare le risorse:** Risparmia tempo e risorse, consentendo di apportare modifiche e miglioramenti più facilmente durante le fasi iniziali dello sviluppo.
5. **Per avere un feedback tempestivo:** Permette di raccogliere feedback dagli utenti e dagli stakeholder precocemente nel processo, facilitando iterazioni rapide e miglioramenti mirati.



Piattaforme low code per ML

Il Machine Learning non è più solo un ambito sperimentale, è diventato uno strumento strategico per generare valore dai dati.

Per affrontare la complessità dei progetti ML e accelerare il time-to-value, le organizzazioni si affidano sempre più a piattaforme integrate in grado di supportare l'intero ciclo di vita dei modelli.

Cos'è una piattaforma ML?

Strumenti integrati che semplificano l'intero ciclo di vita del Machine Learning:

- Ingestion e preparazione dei dati
- Sviluppo e training dei modelli
- Validazione e interpretabilità
- Deploy in produzione e monitoraggio

Perché usarle?

- Favoriscono la collaborazione tra data scientist, analisti e IT
- Offrono interfacce low-code / no-code e supporto per codice avanzato
- Gestiscono infrastruttura, scalabilità e sicurezza
- Abilitano il monitoraggio continuo dei modelli in produzione

Alcuni prodotti open-source e di mercato per ML



Commerciali

- **Dataiku:** Piattaforma completa per l'analisi dei dati con funzionalità avanzate di preparazione, sviluppo e distribuzione di modelli di machine learning.
- **Alteryx:** Piattaforma di analisi che semplifica la preparazione dei dati, l'esecuzione di analisi predittive e la distribuzione di soluzioni analitiche senza codice.
- **Databricks:** Piattaforma basata su Apache Spark per lo sviluppo e la gestione di applicazioni di big data e machine learning.
- **RapidMiner:** Suite di software per l'analisi predittiva e l'apprendimento automatico con un'interfaccia user-friendly e una vasta gamma di strumenti per la preparazione dei dati.
- **SAS Viya:** Piattaforma di analisi avanzata con funzionalità di data wrangling, machine learning e analytics visuali, progettata per la scalabilità e la collaborazione aziendale.



Open source

- **Weka:** Suite di software per l'apprendimento automatico e l'analisi dei dati con una vasta collezione di algoritmi di machine learning e un'interfaccia utente intuitiva.
- **KNIME:** Ambiente di sviluppo per l'analisi dei dati e la creazione di workflow che supporta una vasta gamma di algoritmi di machine learning e offre una forte integrazione con strumenti open source.

Dataiku è una piattaforma commerciale completa per la gestione del ciclo di vita delle applicazioni AI che semplifica lo sviluppo e l'implementazione di modelli AI.

Funzionalità chiave

- **Preparazione dei dati:** Strumenti intuitivi e visuali per l'acquisizione, la pulizia e la trasformazione dei dati da diverse fonti.
- **Sviluppo dei modelli:** Ambiente di sviluppo collaborativo per la creazione, la validazione e l'ottimizzazione di modelli di machine learning e analisi predittiva.
- **Distribuzione delle soluzioni:** Capacità di distribuire facilmente i modelli in produzione, monitorarne le prestazioni e gestirne il ciclo di vita.
- **Collaborazione:** Piattaforma che favorisce la collaborazione tra team di analisti, data scientist e sviluppatori, consentendo la condivisione dei progetti e il lavoro congiunto su analisi complesse.
- **Automazione dei processi:** Strumenti per l'automazione dei processi aziendali basati sui dati, consentendo alle organizzazioni di ottimizzare le operazioni e migliorare l'efficienza.
- **Supporto AI Generativa:** funzioni specifiche per creare applica con AI Generativa



Piattaforme low code per GenAI

L'Intelligenza Artificiale Generativa sta ridefinendo il modo in cui aziende e organizzazioni interagiscono con i dati, i contenuti e le persone.

Le piattaforme GenAI offrono gli strumenti per passare dall'esplorazione all'impatto reale.

Le piattaforme low code permettono di costruire applicazioni basate su Large Language Models (LLM) come GPT-4 o Claude senza scrivere codice.

Cos'è una piattaforma ML?

Strumenti integrati che semplificano lo sviluppo di applicazioni GenAI:

- Prompt Design, Editor visuali per creare prompt con variabili e condizioni
- LLM Integration, Supporto modelli cloud e modelli self-hosted
- Memory & Tools, Memoria conversazionale, function calling, plugin, strumenti personalizzati
- Deployment, Interfacce web, embed in app, API REST
- Data Connector, Retrieval da documenti, database, Notion, Google Docs, ecc.
- Auth & Logging, Utenti, logging conversazioni, limiti, controllo accessi

Perché usarle?

- Creazione di chatbot, assistenti AI, generatori di contenuti
- Interfacce visuali per progettare prompt e flussi logici
- Integrazione semplice con API esterne o dati interni
- Deployment rapido come app web o API



Alcuni prodotti open-source e di mercato per GenAI

Commerciali

- **Dust.tt:** Soluzione per la creazione di assistenti AI per team aziendali, con integrazioni a Notion, Slack e database esterni.
- **Builder.ai:** Piattaforma per creare app AI e soluzioni digitali personalizzate senza scrivere codice, orientata a PMI e startup.
- **Poe (by Quora):** Costruttore di bot AI basati su modelli LLM come GPT e Claude, con interfaccia semplice e pubblicazione rapida.

Open source

- **Langflow:** Interfaccia visuale per costruire flussi AI modulari tramite nodi, basata su LangChain, perfetta per prototipi e esperimenti.
- **Flowise:** Piattaforma visuale open-source per creare agenti AI e flussi LLM, con supporto per document retriever e API REST.
- **Superagent:** Strumento open-source per creare agenti AI personalizzati, schedulabili, con API access e supporto a memory e tool.
- **OpenPipe:** Sistema open-source per la gestione e l'ottimizzazione dei prompt, utile per fine-tuning e A/B test sui modelli LLM.
- **CrewAI:** Framework Python per costruire team di agenti AI autonomi, utilizzabile in progetti LLM complessi.



Langflow è una piattaforma open-source per costruire e testare agenti AI usando una UI visuale basata su nodi connessi (flowchart). Lavora in stretta sinergia con LangChain.

Funzionalità chiave

- **UI drag & drop** per creare flussi LLM
- **Supporto a modelli:** OpenAI, Cohere, Hugging Face, ecc.
- **Componenti modulari:** prompt, retriever, output parser, ecc.
- **Esportazione del flow in codice Python** (LangChain)
- **Ideale per prototipi e sperimentazione rapida**

Use case tipici:

- Chatbot AI con logica condizionale
- Retrieval-augmented generation (RAG)
- Flussi di domande/risposte complesse



Dify



<https://dify.ai/>

Dify è una piattaforma no-code e open-source per creare applicazioni AI basate su modelli linguistici di grandi dimensioni (LLM) come GPT-4, Claude o modelli self-hosted. Offre un ambiente visuale per progettare, testare e distribuire agenti AI personalizzati.

Funzionalità chiave

- **Prompt engineering visuale** con variabili, campi dinamici e modelli multipli
- **Supporto multi-LLM**: OpenAI, Anthropic, Azure, Hugging Face, e modelli locali
- **Knowledge base integrata**: upload documenti e interrogazione tramite RAG
- **Tools e plugins** per espandere le capacità dell'AI (API calls, DB access, ecc.)
- **Gestione utenti e logging**: ruoli, tracciamento conversazioni e limiti d'uso
- **SaaS o self-hosted**

Use case:

- Chatbot AI per siti e app
- Assistenti virtuali personalizzati per knowledge base aziendali
- Generatori di contenuti dinamici
- Co-pilot interni per team di supporto o marketing

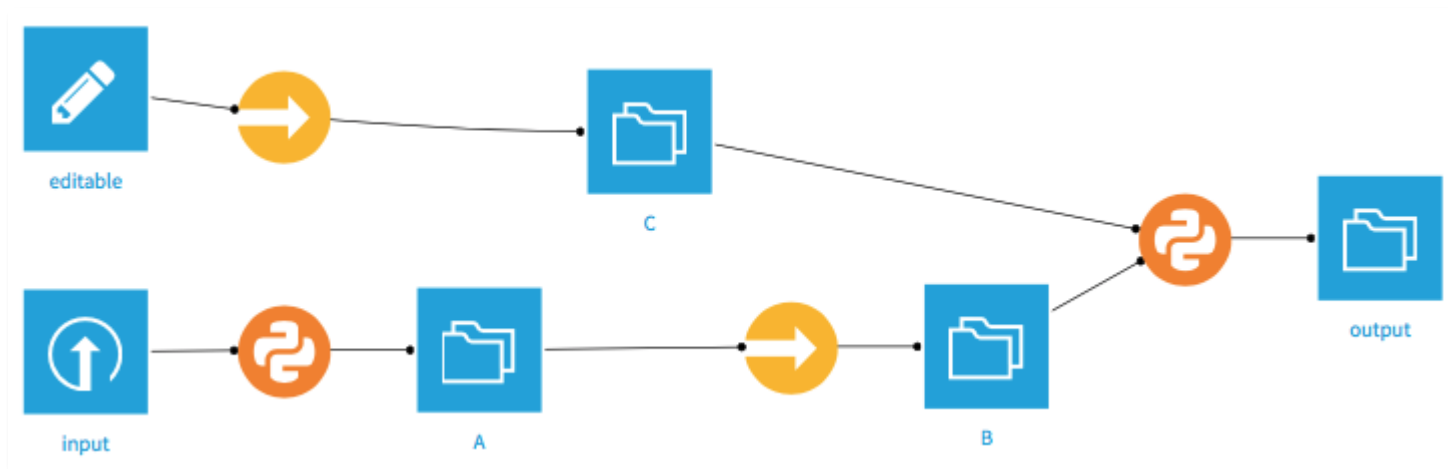


6.Domande e risposte





Addestramento di un modello AI/ML con uno strumento no-code – esempio pratico





Addestramento di un modello AI/ML con uno strumento no-code – esempio pratico

Supporto alla diagnosi precoce del Morbo di Parkinson dall'analisi del parlato





Descrizione del caso

Scopo del Prototipo

Verificare e condividere le potenzialità di utilizzo di algoritmi di AI per supportare la diagnosi precoce del Morbo di Parkinson

Strumenti di Prototipazione -> Dataiku in versione SaaS e Weka

Dataset -> Dataset creato da Max Little dell'Università di Oxford, in collaborazione con il National Centre for Voice and Speech di Denver. Include una serie di misurazioni biomediche della voce di 31 persone, di cui 23 affette da morbo di Parkinson (PD).

<https://www.kaggle.com/datasets/itsmesunil/parkinsons>

Descrizione del Dataset

Campo	Significato
name	Nome e n° di registrazione del paziente in codice ASCII
MDVP:Fo(Hz)	Media della frequenza fondamentale della voce
MDVP:Fhi(Hz)	Massima frequenza fondamentale della voce
MDVP:Flo(Hz)	Minima frequenza fondamentale della voce
MDVP:Jitter(%)	Misure di variazione della frequenza fondamentale
MDVP:Jitter(Abs)	Misure di variazione della frequenza fondamentale
MDVP:RAP	Misure di variazione della frequenza fondamentale
MDVP:PPQ	Misure di variazione della frequenza fondamentale
Jitter:DDP	Misure di variazione della frequenza fondamentale
MDVP:Shimmer	Misure di variazione della ampiezza
MDVP:Shimmer(dB)	Misure di variazione della ampiezza
Shimmer:APQ3	Misure di variazione della ampiezza
Shimmer:APQ5	Misure di variazione della ampiezza
MDVP:APQ	Misure di variazione della ampiezza
Shimmer:DDA	Misure di variazione della ampiezza
NHR	Misure del rapporto tra rumore e componenti tonali della voce
HNR	Misure del rapporto tra rumore e componenti tonali della voce
status	Stato di salute del soggetto (0: in salute, 1: affetto da Parkinson)
RPDE	Due misure di complessità dinamica non lineare
DFA	Esponente di scalatura frattale del segnale
spread1	Tre misure non lineari della variazione della frequenza fondamentale
spread2	Tre misure non lineari della variazione della frequenza fondamentale
D2	Due misure di complessità dinamica non lineare
PPE	Tre misure non lineari della variazione della frequenza fondamentale

La colonna "status" è impostata a 0 per le persone sane e a 1 per le persone affette da Morbo di Parkinson.




Training dell'algoritmo con Dataiku

Random forest (s1) - v1



Model

Model ID	S-PARKINSONANALYSIS-BR3VLJMu-initial 
Model type	Two-class classification
Target	status
Classes	0 1
Backend	Python (in memory)
Algorithm	Random forest classification
Trained on	2024/01/24 10:28
Columns	27
Train set rows	158
Test set rows	37
Calibration method	No calibration
Code Env	DSS builtin env
Python version	3.7.13

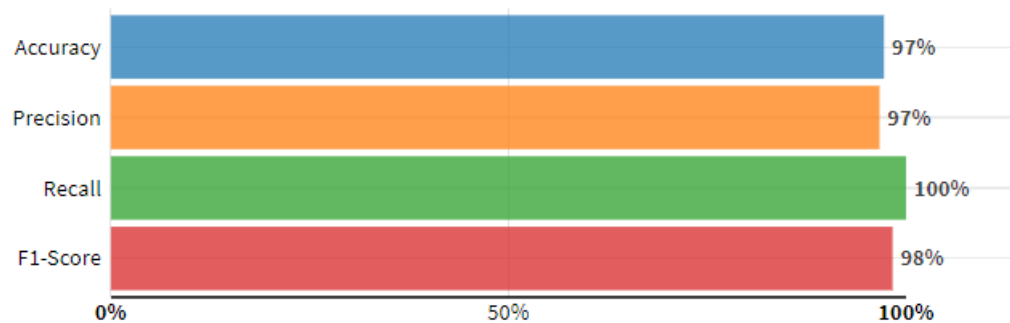


Matrice di Confusione

Confusion matrix

Display: Record count

	Predicted <input type="text" value="1"/>	Predicted <input type="text" value="0"/>	Total
Actually <input type="text" value="1"/>	29	0	29
Actually <input type="text" value="0"/>	1	6	7
Total	30	6	36

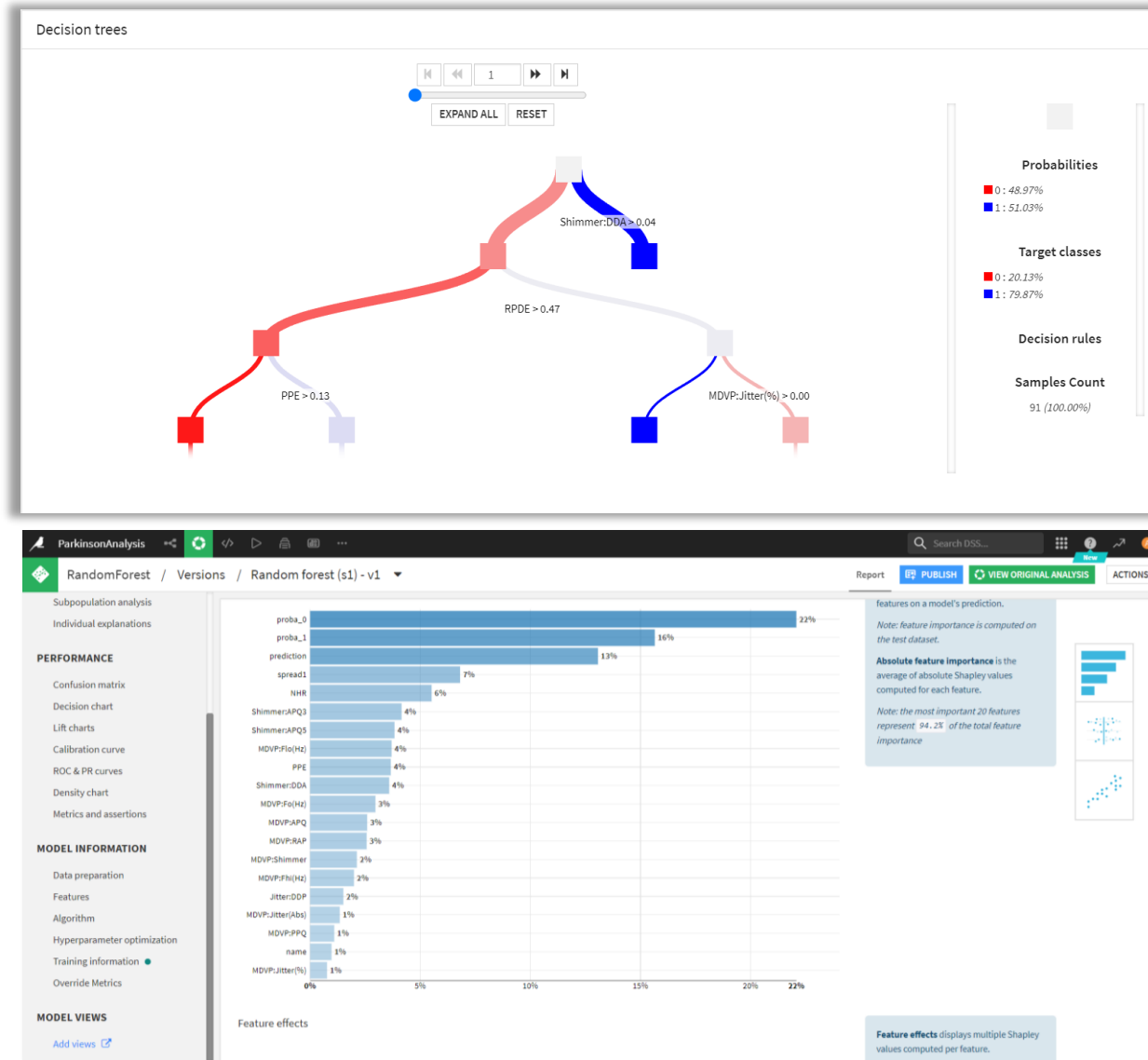


Matrice di Confusione

- **Accuracy:** (Veri Positivi + Veri Negativi) / Totali. Quanti valori complessivamente (tra positivi e negativi) l'algoritmo ha predetto correttamente?
- **Precision:** Veri Positivi / (Veri Positivi + Falsi Positivi) cioè quanti positivi esatti tra quelli predetti? (es. l'algoritmo ha predetto che 100 sono positivi ma in realtà 5 sono dei falsi positivi)
- **Recall:** Veri positivi / (Veri positivi + Falsi Negativi) cioè quanti positivi esatti tra tutti i positivi totali? L'algoritmo ha predetto 93 dei 100 positivi effettivamente nel dataset
- **F1 Score:** Media armonica tra Precision e Recall. Molto utile nel caso di dataset sbilanciati (es. pochissimi casi negativi)

$$FS = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}$$

Explainability



Training Information

The screenshot displays the Dataiku DSS interface for a project named "pakinso's disease prediction". The breadcrumb navigation shows the path: "Choose Algorithms for status on Parkinson_disease / Models / Random forest". The left sidebar contains a navigation menu with sections: "Decision trees", "Variables importance", "Partial dependence", "Subpopulation analysis", "Individual explanations", "Interactive scoring", "PERFORMANCE", "MODEL INFORMATION", and "Training Information" (which is currently selected and highlighted in green).

The main content area is divided into several panels:

- Diagnostics:** Contains a "Dataset sanity checks" section with a warning icon and text: "Training set might be too small (158 rows) for robust training" and "Test set might be too small (37 rows) for reliable performance estimation". Below this are sections for "Modeling parameters", "Training speed", "Overfit detection", "Leakage detection", "Model check", and "ML assertions", all of which report "Nothing to report".
- Steps:** A table showing the execution steps and their durations:

Step	Duration
Loading train set	0.0s
Loading test set	0.0s
Collecting statistics	0.1s
Preprocessing train set	0.1s
Preprocessing test set	0.1s
Hyperparameter searching	3.1s
Fitting model	0.5s
Saving model	0.2s
Scoring model	1.6s
- Train & validation policy:** A table with the following data:

Policy	Split the dataset
Sampling method	First records
Partitions	All partitions
Record limit	100000
Split mode	Randomly
Train ratio	0.8
Random seed	1337
- Train & test sets:** A table with the following data:

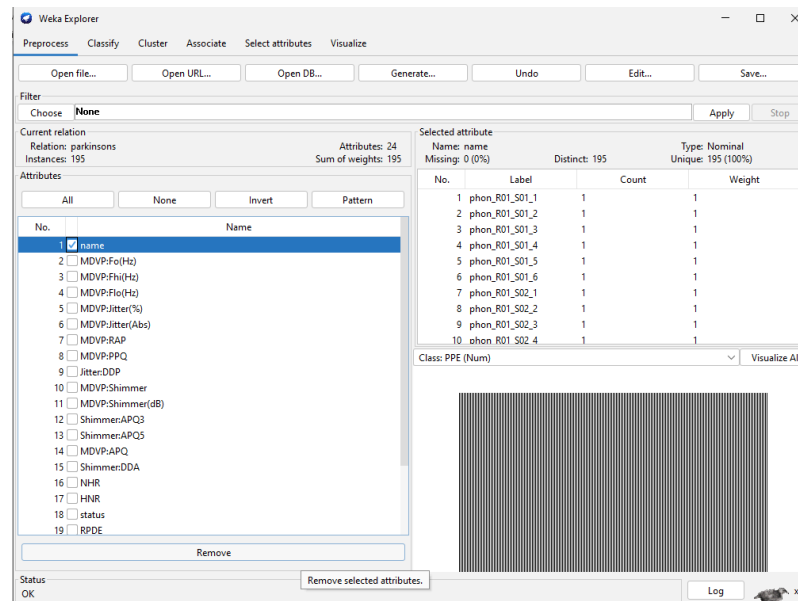
Generated on	2022/02/04 15:46:04
Train set rows	158
Test set rows	37
- Hyper parameters:** A table with the following data:

Mode	5-fold cross-test
Stratified	Yes
- Runtime configuration:** A table with the following data:

Skip expensive reports	No
Enabled Diagnostics	Yes
- Timings:** A table with the following data:

Preprocessed in	0.3s
Trained in	5.4s

Utilizzo dello strumento Weka



Classificazione e stampa output con Weka

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose **RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1**

Test options

☐ Use training set

☐ Supplied test set

☐ Cross-validation Folds

☒ Percentage split %

(Nom) status

Result list (right-click for options)

- 16:55:26 - trees.RandomForest
- 16:56:05 - trees.RandomForest
- 16:58:18 - trees.RandomForest
- 16:58:42 - trees.RandomForest
- 16:58:05 - trees.RandomForest**

Classifier output

Time taken to build model: 0.03 seconds

=== Predictions on test split ===

inst#	actual	predicted	error	prediction
1	1:T	1:T	0.97	
2	1:T	2:F	0.52	
3	1:T	1:T	0.94	
4	1:T	1:T	0.84	
5	1:T	1:T	0.82	
6	1:T	1:T	1	
7	1:T	1:T	0.99	
8	1:T	1:T	0.96	
9	1:T	1:T	0.71	
10	1:T	1:T	0.92	
11	2:F	2:F	0.86	
12	1:T	1:T	1	
13	1:T	1:T	0.93	
14	2:F	1:T	0.68	
15	1:T	1:T	0.99	
16	2:F	2:F	0.84	
17	1:T	2:F	0.82	
18	1:T	1:T	0.98	
19	2:F	2:F	0.6	
20	1:T	1:T	0.83	
21	1:T	1:T	0.98	
22	1:T	1:T	0.82	
23	1:T	1:T	0.78	
24	2:F	2:F	0.92	
25	1:T	1:T	0.91	
26	1:T	1:T	1	
27	1:T	1:T	0.97	
28	1:T	1:T	0.87	
29	1:T	1:T	1	
30	1:T	1:T	0.96	
31	1:T	1:T	0.63	
32	1:T	1:T	0.73	
33	1:T	1:T	0.94	
34	2:F	2:F	0.69	
35	1:T	1:T	0.93	
36	1:T	1:T	0.99	
37	1:T	1:T	0.87	
38	1:T	1:T	0.78	
39	2:F	2:F	0.8	

=== Evaluation on test split ===

Time taken to test model on test split: 0.05 seconds

=== Summary ===

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose **RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1**

Test options

☐ Use training set

☐ Supplied test set

☒ Cross-validation Folds

☐ Percentage split %

(Nom) status

Result list (right-click for options)

- 16:55:26 - trees.RandomForest
- 16:56:05 - trees.RandomForest**

Classifier output

Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===

=== Summary ===

Total Number of Instances 195

=== Detailed Accuracy By Class ===

	Precision	Recall	F-Measure	MCC	Class
	0.921	0.952	0.936	0.730	T
	0.837	0.750	0.791	0.730	F
Weighted Avg.	0.900	0.903	0.901	0.730	

=== Confusion Matrix ===

	a	b	<-- classified as
140	7	1	a = T
12	36	1	b = F

Status OK

x 0



Explainability con Weka

weka.gui.GenericObjectEditor

weka.classifiers.trees.RandomForest

About

Class for constructing a forest of random trees.

More

Capabilities

bagSizePercent 100

batchSize 100

breakTiesRandomly False

calcOutOfBag False

computeAttributeImportance True

debug False

doNotCheckCapabilities False

maxDepth 0

numDecimalPlaces 2

numExecutionSlots 1

numFeatures 0

numIterations 100

outputOutOfBagComplexityStatistics False

printClassifiers True

seed 1

storeOutOfBagPredictions False

Open... Save... OK Cancel

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose RandomForest - P 100 - print-attribute-importance - I 100 - num-slots 1 - K 0 - M 1.0 - V 0.001 - S 1

Test options

Use training set

Supplied test set

Cross-validation Folds 10

Percentage split % 80

More options...

Classifier output

```

I | I | I | MDVP:F0 (Hz) >= 127.8 : T (2/0)
I | I | MDVP:Shimmer (dB) >= 0.19
I | I | D2 < 2.74 : T (38/0)
I | I | D2 >= 2.74
I | I | MDVP:Jitter (Abs) < 0 : F (1/0)
I | I | MDVP:Jitter (Abs) >= 0 : T (1/0)
I | Shimmer:D2A >= 0.05 : T (80/0)

```

(Nom) status

Start Stop

Result list (right-click for options)

17.01.52 - trees.RandomForest

17.07.38 - bayes.NaiveBayes

17.08.31 - trees.HoeffdingTree

17.09.22 - Functions.MultilayerPerceptron

17.10.14 - trees.RandomForest

Attribute Importance based on average impurity decrease (and number of nodes using that attribute)

0.5	(144)	MDVP:F0 (Hz)
0.47	(103)	MDVP:F0 (Hz)
0.46	(40)	MDVP:F0 (Hz)
0.45	(73)	MDVP:Jitter (k)
0.45	(70)	MDVP:RAP
0.43	(49)	Jitter:DDP
0.43	(85)	MDVP:F0 (Hz)
0.42	(27)	MDVP:Jitter (Abs)
0.42	(38)	RNR
0.41	(44)	MDVP:Shimmer (dB)
0.37	(36)	Shimmer:D2A
0.37	(56)	RNR
0.36	(42)	DFA
0.36	(62)	spread2
0.35	(43)	Shimmer:APQ5
0.35	(57)	RSGC
0.34	(41)	Shimmer:APQ3
0.33	(63)	MDVP:Shimmer
0.3	(58)	D2
0.28	(91)	PPE
0.28	(48)	spread1
0.25	(54)	MDVP:APQ

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose RandomForest - P 100 - print-attribute-importance - I 100 - num-slots 1 - K 0 - M 1.0 - V 0.001 - S 1

Test options

Use training set

Supplied test set

Cross-validation Folds 10

Percentage split % 80

More options...

(Nom) status

Start Stop

Result list (right-click for options)

17.01.52 - trees.RandomForest

17.07.38 - bayes.NaiveBayes

17.08.31 - trees.HoeffdingTree

17.09.22 - Functions.MultilayerPerceptron

17.10.14 - trees.RandomForest

Classifier output

```

RandomTree
=====
spread1 < -4.76
I | MDVP:F0 (Hz) < 185.32
I | MDVP:Shimmer < 0.02 : T (1/0)
I | MDVP:F0 (Hz) >= 185.32 : F (26/0)
spread1 >= -4.76
I | MDVP:APQ < 0.02
I | MDVP:APQ < 0.01
I | RNR < 0 : T (10/0)
I | RNR >= 0
I | MDVP:F0 (Hz) < 117.72
I | MDVP:Jitter (k) < 0 : F (10/0)
I | MDVP:F0 (Hz) >= 117.72
I | MDVP:APQ < 0.01 : T (18/0)
I | MDVP:APQ >= 0.01 : F (8/0)
I | MDVP:APQ >= 0.01
I | DFA < 0.41
I | MDVP:F0 (Hz) < 88.56 : T (4/0)
I | MDVP:F0 (Hz) >= 88.56
I | RNR < 0.03 : F (4/0)
I | RNR >= 0.03 : T (12/0)
I | DFA < 0.41 : T (28/0)
MDVP:APQ >= 0.02 : T (10/0)
Size of the tree : 23
RandomTree
=====
Shimmer:APQ3 < 0.01
I | spread1 < -4.59 : F (29/0)
spread1 >= -4.93
I | RNR < 26.45
I | D2 < 2.15
I | Jitter:DDP < 0.01
I | MDVP:F0 (Hz) < 88.04 : T (2/0)
I | MDVP:F0 (Hz) >= 88.04
I | Jitter:DDP < 0 : F (8/0)
I | Jitter:DDP >= 0 : T (12/0)
I | Jitter:DDP >= 0 : F (11/0)
I | Jitter:DDP >= 0.01
I | MDVP:F0 (Hz) < 74.27 : F (10/0)
MDVP:F0 (Hz) >= 74.27 : T (8/0)
I | RNR >= 3.18

```





Addestramento di un modello AI/ML con uno strumento no-code – esempio pratico

Predizione costo assicurazione sanitaria





Descrizione del caso

Scopo del Prototipo

Verificare e condividere le potenzialità di utilizzo di algoritmi di regressione per predire i costi sanitari

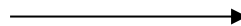
Strumenti di Prototipazione -> Dataiku in versione SaaS

Dataset -> Dataset di pubblico dominio, creato da Brett Lantz per il libro Machine Learning with R.

<https://www.kaggle.com/datasets/mirichoi0218/insurance>

Classificazione vs Regressione

Classificazione



Predire l'appartenenza ad una **classe** o ad un'altra.

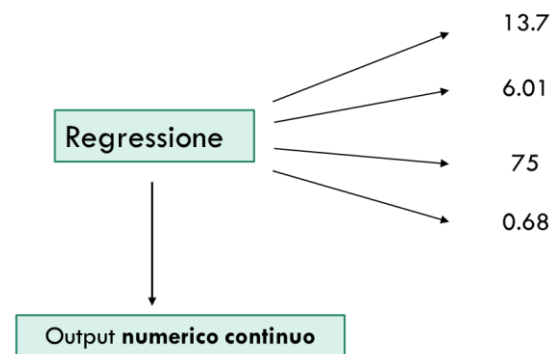
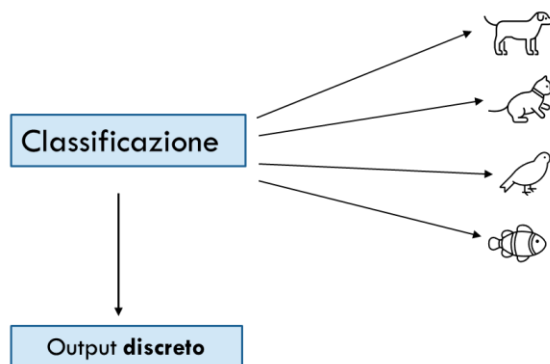
Output **discreto**

Regressione



Predire il valore numerico della variabile.

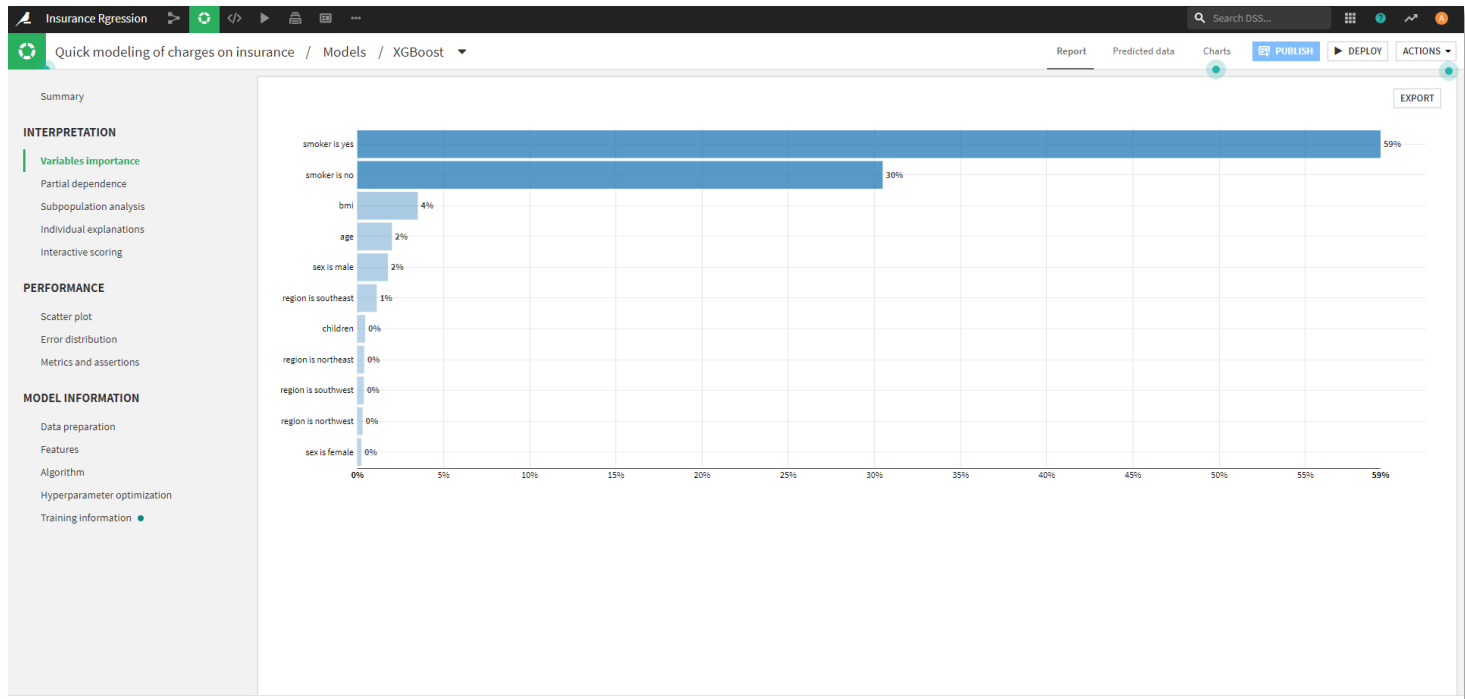
Output **numerico continuo**



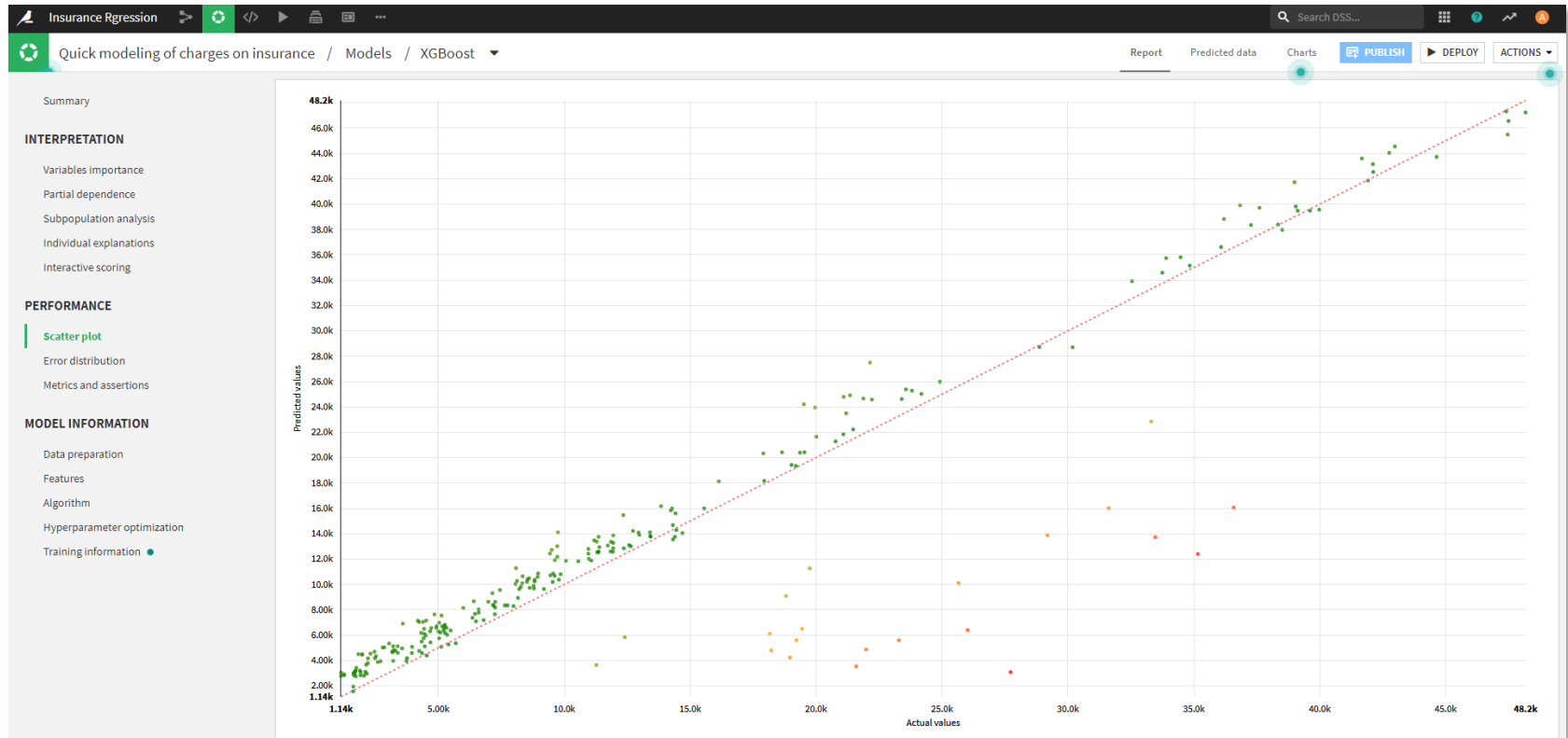
Descrizione del Dataset

Campo	Significato
age	Età
sex	Sesso
bmi	Indice di massa corporea
children	Numero di figli coperti da assicurazione sanitaria
smoker	Fumatore (0;1)
region	Area residenziale del beneficiario
charges	Spese mediche individuali fatturate dall'assicurazione sanitaria

Explainability



Prestazioni del modello





Dati di Output

Insurance Regression

Quick modeling of charges on insurance / Models / XGBoost

Report Predicted data Charts PUBLISH DEPLOY ACTIONS

252 matching rows

age	sex	bmi	children	smoker	region	charges	prediction	error	relative_error	error_decile	abs_error_decile
Integer	Gender	Decimal	Integer	Boolean	Text	Decimal	Decimal	Decimal	Decimal	Integer	Integer
28	male	33.0	3	no	southeast	4449.462	6082.5596	-1633.0975703124996	-0.3670325918757143	0	0
33	male	22.705	0	no	northwest	21984.47061	4859.435	17125.03555140625	0.7789605606248551	9	9
31	female	25.74	0	no	southeast	3756.6216	4178.558	-421.93650546875006	-0.11231807469476034	1	0
37	male	29.83	2	no	northeast	6406.4107	8664.076	-2257.6654718749996	-0.352407233566059	0	1
27	male	42.13	0	yes	southeast	39611.7577	39485.38	126.37879375000193	0.00319043640292695	1	0
30	male	35.3	0	yes	southwest	36837.467	39905.88	-3068.411906250003	-0.0832959526302393	0	1
18	male	34.1	0	no	southeast	1137.011	2775.783	-1638.771958984375	-1.441298245121969	0	0
63	female	23.085	0	no	northeast	14451.83515	14280.915	170.9201109375008	0.011826879366078348	1	0
19	male	20.425	0	no	northwest	1625.43375	1937.1266	-311.6928369140626	-0.19175979144893637	1	0
60	male	39.9	0	yes	southwest	48173.361	47233.49	939.87271787499971	0.019510216834361986	2	0
48	male	28.0	1	yes	southwest	23568.272	25380.438	-1812.1654999999992	-0.07689004522690501	0	0
58	male	36.955	2	yes	northwest	47496.49445	46557.094	939.40069999999983	0.019778316502682407	2	0
34	female	37.335	2	no	northwest	5989.52365	8131.256	-2141.732209375	-0.35757972328483917	0	1
19	female	28.9	0	no	southwest	1743.214	3412.2832	-1669.069203125	-0.9574666123178222	0	0
61	female	39.1	2	no	southwest	14235.072	15841.436	-1606.3635468749999	-0.11284548099756712	0	0
31	male	28.5	5	no	northeast	6799.458	7180.2476	-380.78955859375037	-0.056002928261892404	1	0
58	male	32.01	1	no	southeast	11946.6259	13867.335	-1920.709060937501	-0.1607741865372633	0	1
57	male	34.01	0	no	northwest	11356.6609	12532.251	-1175.5900765624992	-0.10351546875565328	1	0
29	female	29.59	1	no	southeast	3947.4131	4578.9077	-631.4946148437498	-0.15997682503605962	1	0
45	male	22.895	2	yes	northwest	21098.55405	24793.732	-3599.9996137500016	-0.1751389390532666	0	1
56	female	27.2	0	no	southwest	11073.176	11881.143	-807.9665781250005	-0.07296611000538604	1	0
59	male	29.83	3	yes	northeast	30184.9367	28707.27	1477.6671687499984	0.04895379385539671	2	0
18	male	23.75	0	no	northeast	1705.6245	3166.7944	-1461.16993359375	-0.8566773833242605	0	0
26	female	28.785	0	no	northeast	3385.39915	5115.349	-1729.9499710937498	-0.5110032508555896	0	0
22	male	25.175	0	no	northwest	2045.68525	2795.878	-750.1926796875	-0.3667195037396393	1	0
127.0.0.1:10000/projects/INSURANCEREGRESSION/analysis/ORUBMGFO/ml/p/A0bcxtdQ/A-INSURANCEREGRESSION-ORUBMGFO-A0bcxtdQ-s6-pp2-m1/table/								12058.33156640625	0.6640827135512023	7	6

Un assistente virtuale Insegnante di AI



Descrizione del caso

Scopo del Prototipo

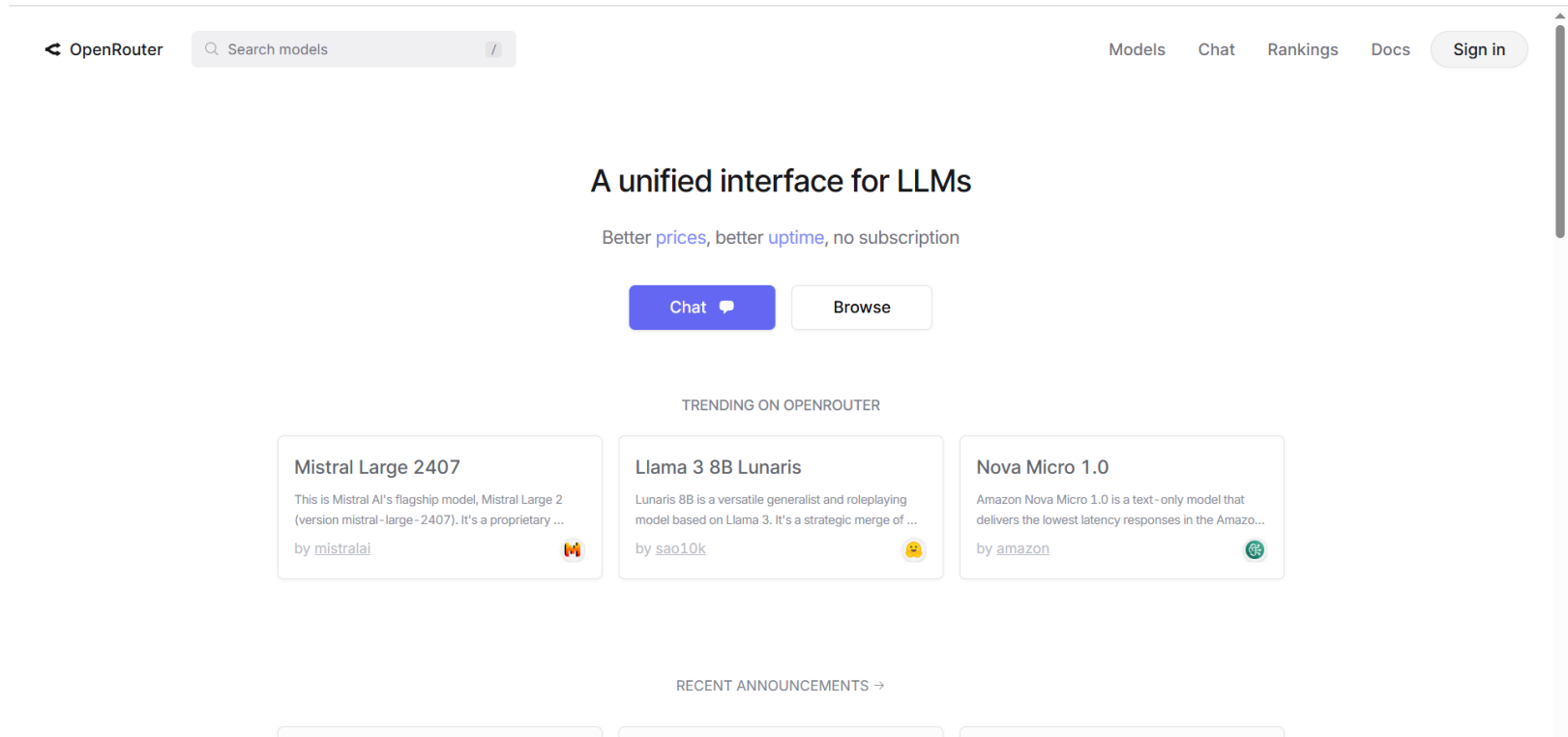
Creare un assistente virtuale specializzato nel dare risposte su AI, come foss eun insegnante

Strumenti di Prototipazione -> Langflow + OpenRouter

Step:

- Creare account su OpenRouter
- Recuperare Api Key
- Creare account su Langflow
- Creare flusso di Assistente Virtuale

Creazione account OpenRouter.ai




<https://openrouter.ai/>



Creare Api Key di openrouter.ai

Router Search models /

Models Chat Rankings Docs 

Credits
Keys
Activity
Settings
Sign out

A unified interface for LLMs

Better [prices](#), better [uptime](#), no subscription

Chat Browse

OpenRouter Search models /

Settings
Credits
API Keys

API Keys

Manage your API keys to access all models from OpenRouter ⓘ

Create Key

Create a Key

Your new key:

sk-or-v1-26d3520b6f70bf6d8370c3051bfb650t

Please copy it now and write it down somewhere safe. You will not be able to see it again.

You can use it with OpenAI-compatible apps, or [your own code](#)

Creare account LangFlow

← → ↻ 🏠 🌐 langflow.org ☆ 🗂️ 🔴 ⋮

🔗 IBM plans to Acquire DataStax, Accelerating Production AI & NoSQL Data at Scale [Read more](#)

Langflow Marketplace **Coming Soon** Docs Resources ▾ Community ▾

🔄 53k 💬 12k ✂️ 8k 📺 6k

Stop fighting your tools

Langflow is a low-code tool for developers that makes it easier to build powerful AI agents and workflows that can use any API, model, or database.

[Get Started for Free](#) [Star on GitHub](#)

Input
Type something... 🌐 ↺

Model
4o gpt-4o-mini ⌵

API Key
Type something...

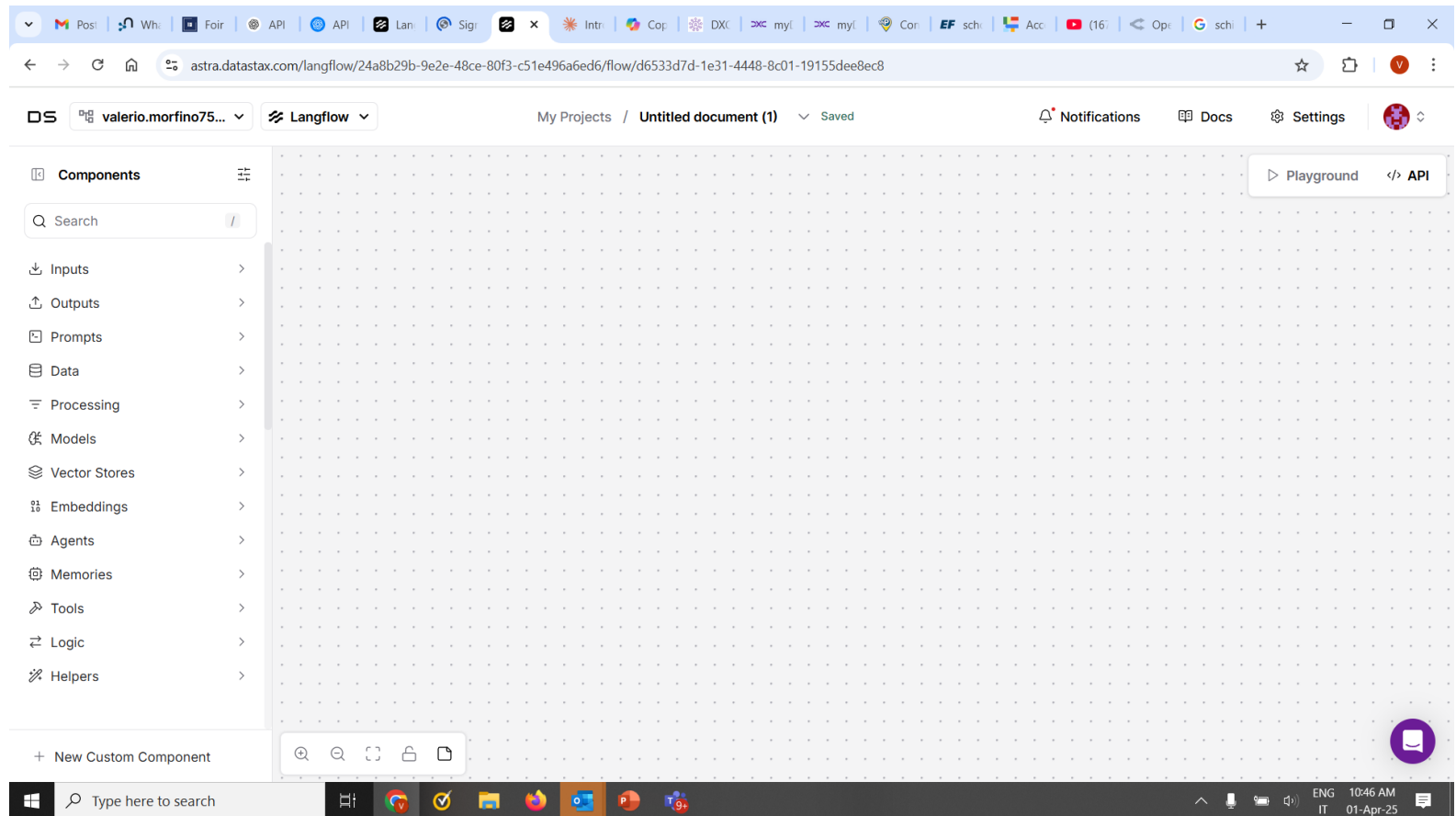
Temperature
0.50
🔧 Precise 🧠 Creative

Used by leading AI development teams

P&P IntimeGDT Global Data Tracker Fyxt SurveyHub.AI SURVEY SERVICES MARKETPLACE Athena Intelligence ecclesiastical Foqum

<https://www.langflow.org/>

Creare nuovo Flow



Creare nuovo Flow

The screenshot displays the Langflow web interface in a browser. The URL bar shows a specific flow ID. The interface includes a top navigation bar with user information, project name, and various settings. A left sidebar lists component categories like Inputs, Outputs, Prompts, Data, Processing, Models, Vector Stores, Embeddings, Agents, Memories, Tools, Logic, and Helpers. The main workspace contains a workflow diagram with three components: 'Chat Input', 'Prompt', and 'OpenRouter'. The 'Prompt' component has a template 'Rispondi come un insegnante esp...' and a 'user_input' field. The 'OpenRouter' component is configured with 'Receiving input', 'Type something...', 'Stream' toggle, 'OpenRouter API Key', 'Provider' set to 'Openai', 'Model' set to 'openai/gpt-4o-mini', 'Temperature' slider at 0.78, and 'Precise' checked. The workflow is connected to a 'Chat Output' component. A right sidebar shows 'Playground' and 'API' tabs. The bottom status bar shows system information like language (ENG), time (10:47 AM), and date (01-Apr-25).

Analisi di un flusso RAG

