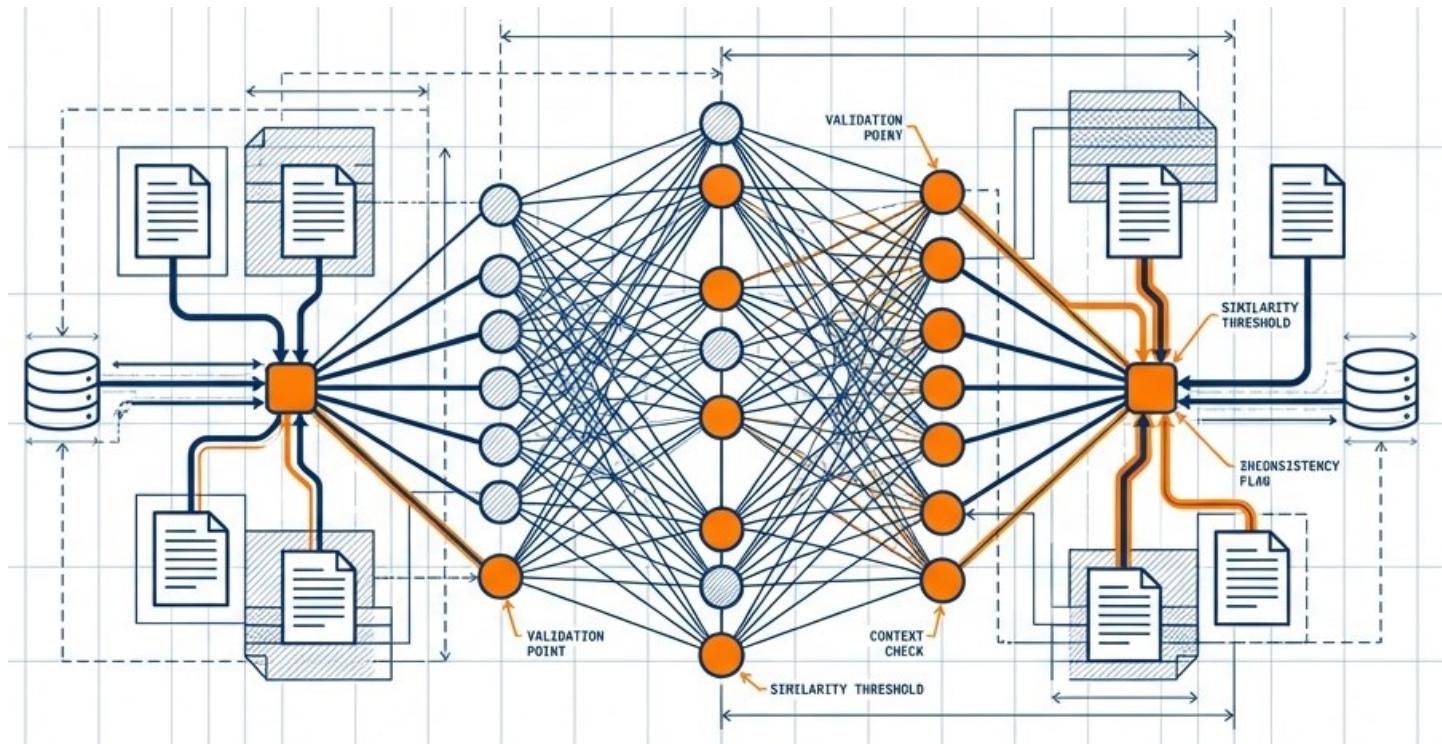


NeuroSymbolic Artificial Intelligence at Scale

Parte: 5.1 (2025-26) from LLM RAG to Agentic LLM

Paolo Nesi, paolo.nesi@unifi.it

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

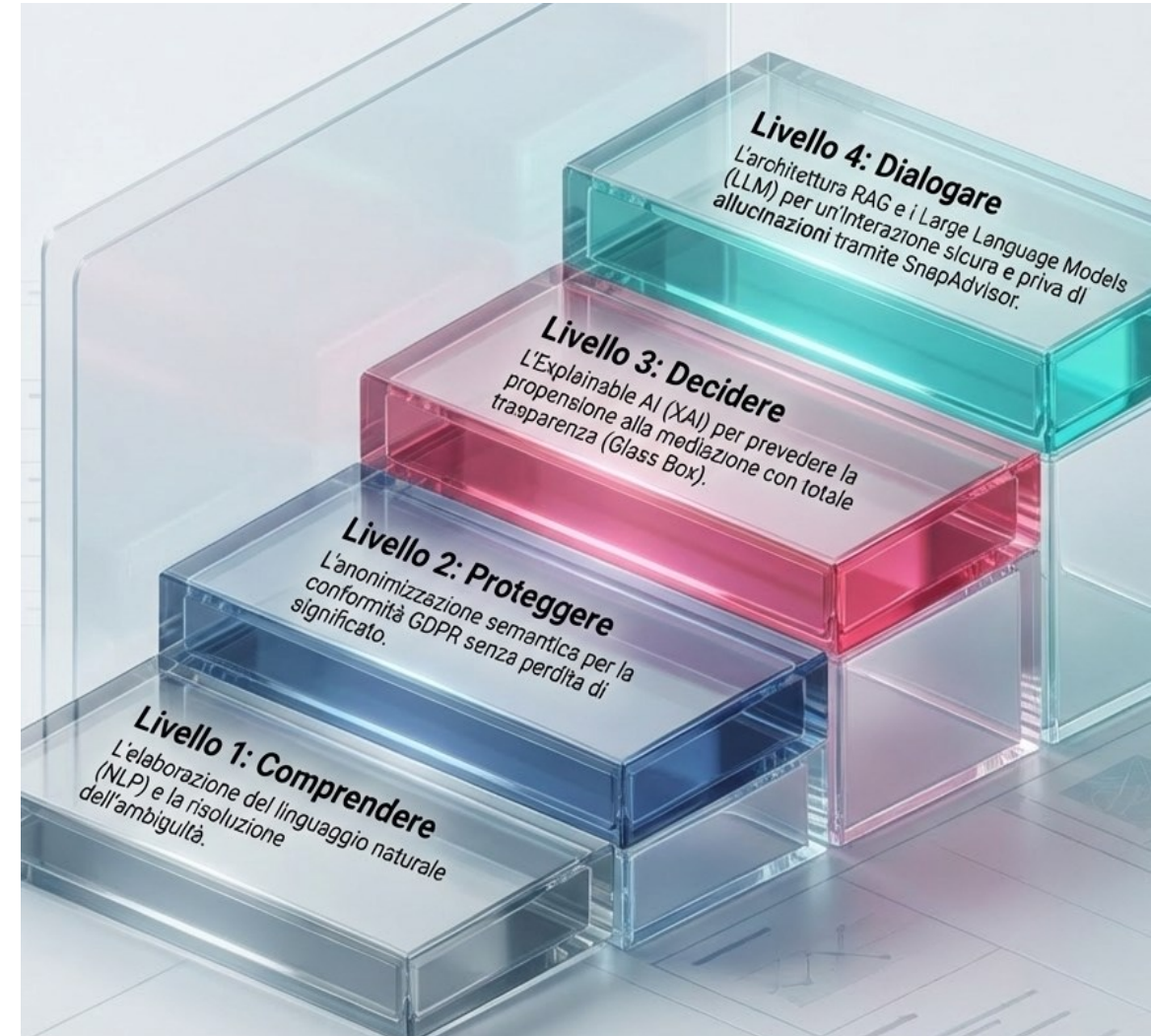


Aspettative

rispondere a domande come un esperto che *studiando il caso* è in grado di fornire suggerimenti

- **referiti** ad affermazioni localizzate nei documenti e fornire spiegazione
- con una **riduzione dei tempi** di analisi di una vertenza,
 - dal punto vista computazionale è instancabile
- **non influenzate** da precedenti documenti,
- **non affetti** da allucinazioni (deduzioni inventate)
- **Con minore varianza** dovuta alle variazioni di contesto, al quale gli umani sono sensibili.

→ *Decision Support System*



La Sfida

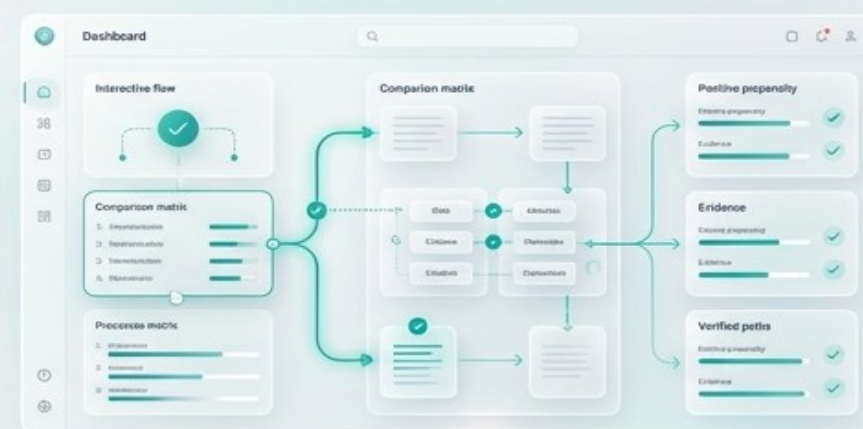


Analisi manuale di voluminosi documenti relativi al contenzioso.

Comprensione profonda del linguaggio naturale e del contesto giuridico.

Vulnerabilità umane: affaticamento e varianza dovuta alle variazioni di contesto.

L'Obiettivo



Creare un assistente computazionale instancabile.

Rispondere a domande mirate (es. Che tipo di rischio si ha?, Vi sono altre strutture coinvolte?)

Fornire suggerimenti basati su affermazioni localizzate nei documenti.

Requisito Fondamentale: Non influenzato da documenti precedenti, non affetto da allucinazioni (deduzioni inventate).



Ambiguità dei linguaggi naturali (1)

Ambiguità Lessicale

Stessa parola, funzione diversa.
«attacco» (verbo vs. sostantivo)



Ambiguità Strutturale

La sintassi crea doppi sensi.
«Ieri ho visto l'uomo col telescopio»
(Chi ha il telescopio?)



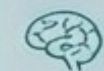
Ambiguità Semantica

Stessa parola, significato divergente.
«acuto» (persona intelligente vs. tipo di suono)



Ambiguità Pragmatica

L'intenzione comunicativa dipende dal ricevente.
«Se Buffon non gioca contro la Spagna, l'Italia perderà.»



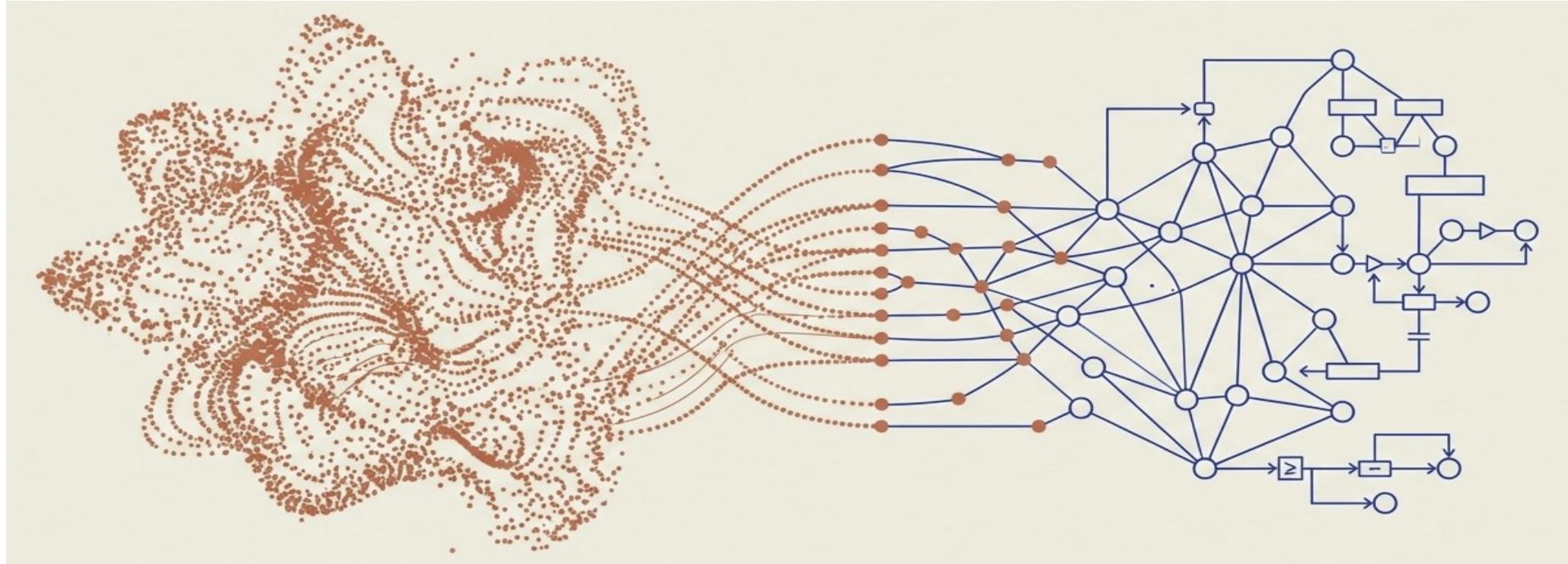
Interpretazione **Emotiva** (assenza psicologicamente fondamentale) vs. Interpretazione **Referenziale** (squadra fisicamente più debole).

Syntax Tree: Parsing del Linguaggio Naturale



L'approccio moderno (BERT) unisce questa rigida struttura sintattica al Machine Learning per comprendere il dominio specifico di conoscenza.

Obiettivi



Soluzioni AI sul linguaggio naturale

- **Creazione di report di sintesi su Vertenze, Documentazione privata**
 - Evoluzione temporale degli accadimenti, Causalità degli eventi, ordinamento e causa/effetto
 - Riferimenti ad articoli o a documenti
 - Ricerca delle evidenze su materiale multimodale
 - Etc.
- **Valutazione**
 - della propenzione alla mediazione, dai documenti di vertenze
 - delle richieste di risarcimento
 - della documentazione da analizzare per una perizia

Per esempio: Propensione alla Mediazione

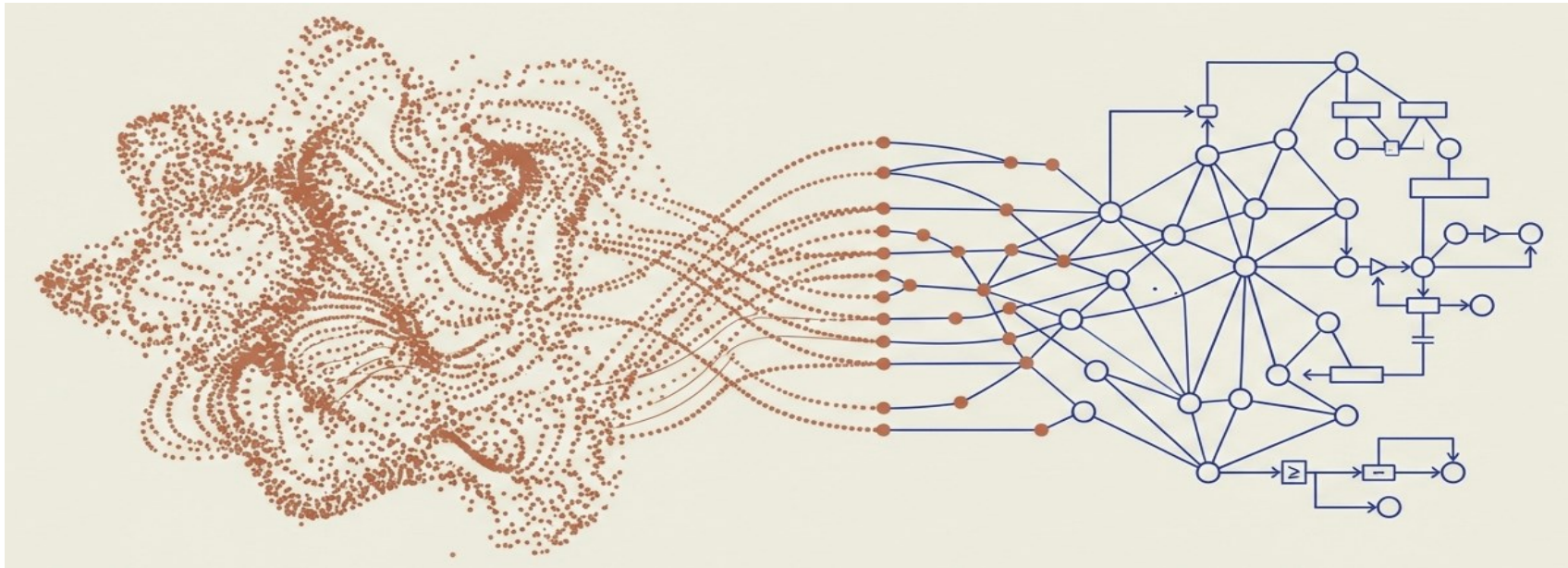
- **Probabilità che una vertenza**
 - possa essere effettivamente mandata in mediazione
 - *Vi sono delle condizioni specifiche per quali la mediazione viene esclusa*
 - sia mandata in mediazione da parte del giudice
 - *Il giudice valuta sulla base della sua esperienza*
 - Se mandata in mediazione venga risolta con successo: quando lo sappiamo?
 - *Se le parti effettivamente trovano un accordo*
 - *Non sempre si è riesce a conoscere il risultato se positivo*

Per esempio: stima della Propensione alla Mediazione

- **Passi:**

- **Modelli di Anonimizzazione** orientati alla comprensione
- **Analisi dei documenti** del fascicolo digitale
- **Stima della propensione** alla mediazione delle parti sulla base del contenuto del testo, come suggerimento per la procedura
- **Supporto al procedimento** per decidere o meno l'invio in mediazione
- **Presentazione delle motivazioni alla base della stima:** porzioni di testo rilevanti per la stima

Anonimizzazione necessaria o meno ?

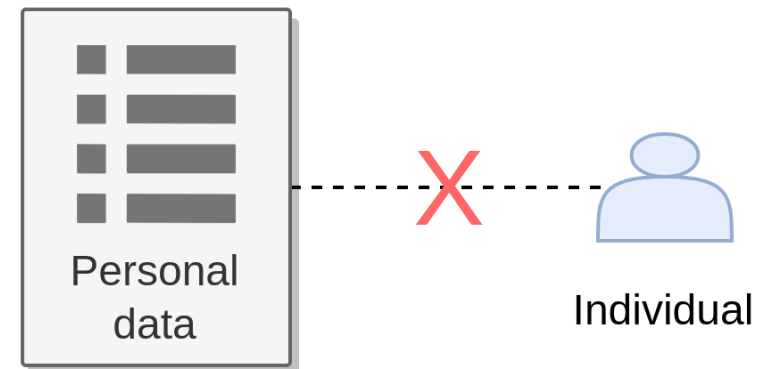


Anonimizzazione

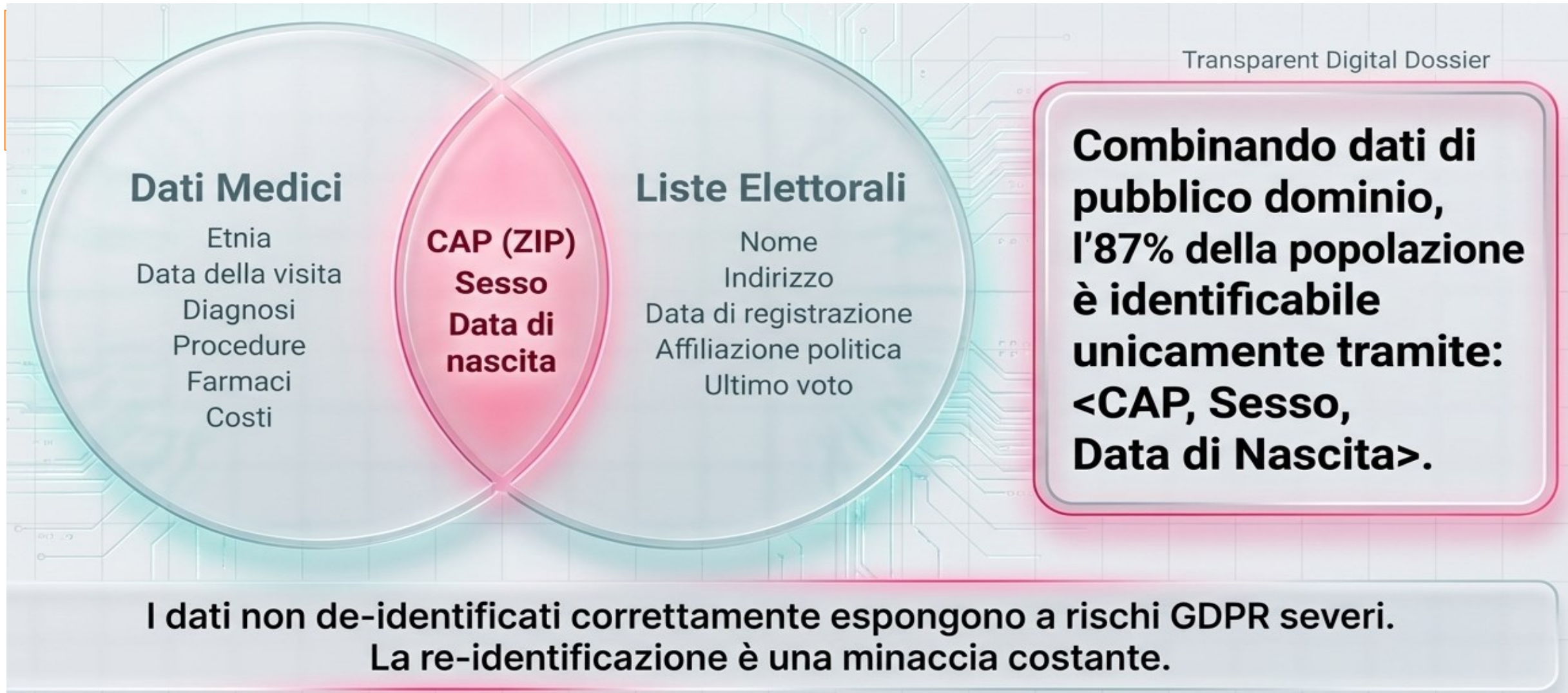
- **La rimozione dei dati personali è un processo utile per i seguenti motivi:**
 - Tutelare la privacy dei soggetti coinvolti
 - Realizzare un dataset immune ad adversarial attacks, ovvero al recupero dei dati di addestramento
 - Realizzare un dataset privo di bias, ovvero non influenzato da informazioni personali
 - GDPR che dice ?
- **Esempi**
 - **Forza bruta:**
 - Mario Rossi è figlio di Marco Rossi. Marco Rossi ha invaso il giardino di Giuseppe Bianchi.
 - Xxxx è figlio di Xxxx. Xxxx ha invaso il giardino di Xxxx.
 - **Conservazione della Semantica:**
 - Mario Rossi è figlio di Marco Rossi. Marco Rossi ha invaso il giardino di Giuseppe Bianchi.
 - Act1 è figlio di Act2. Act2 ha invaso il giardino di Act3.

Anonymization methods

- Remove any link between data and individuals
- Suitable for anonymizing documents
 - The information of interest is in the context rather than in the personal data.
 - Beware of loss of semantics!
- Re-identification is **impossible**
 - Data are no longer protected by GDPR
 - Data can be published without concern



Re-identification



Evolution of Data Redaction: From Brutal Force to Semantic Preservation

Mario Rossi è figlio di Marco Rossi.
Marco Rossi ha invaso il giardino di
Giuseppe Bianchi.



Rimozione / Forza Bruta

OMISSIS è figlio di OMISSIS. OMISSIS ha invaso il giardino di OMISSIS.

Perdita totale della semantica. Inutile per l'addestramento AI.



Tagging Semplice

#PERSON è figlio di #PERSON. #PERSON ha invaso il giardino di #PERSON.

Trascura le relazioni tra le entità. Chi ha invaso il giardino?



Tagging Enumerato

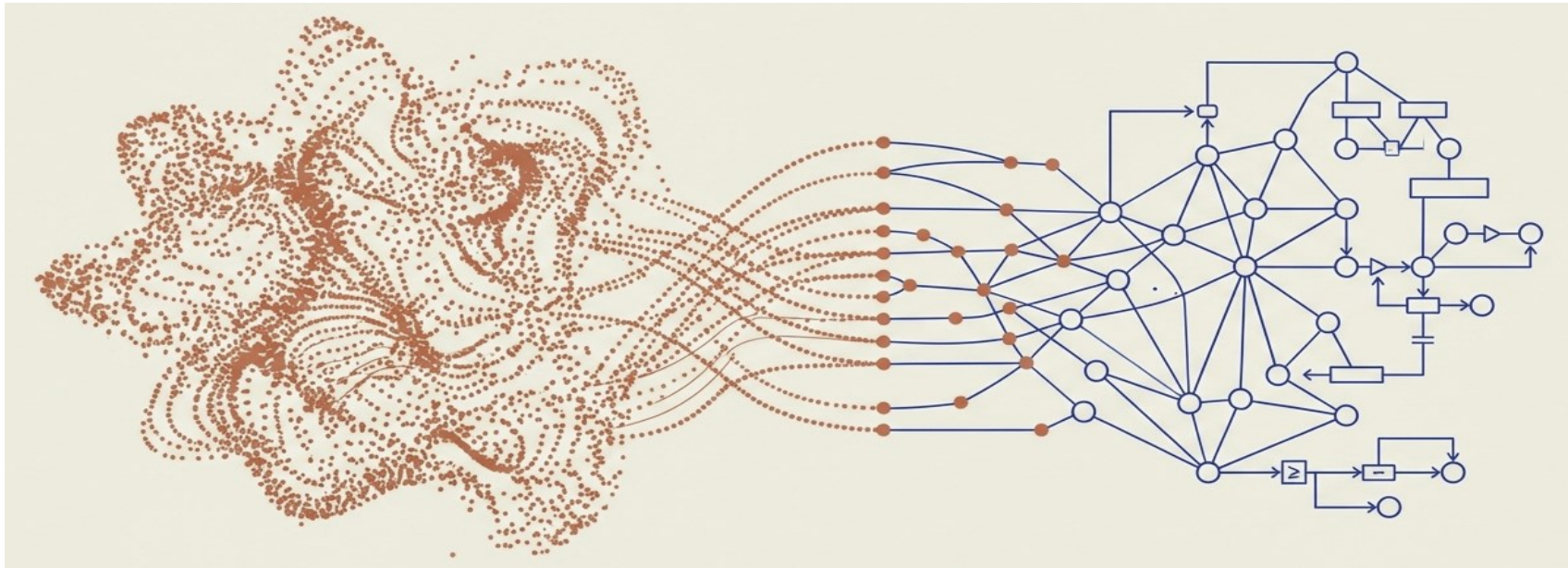
#PERSON_1 è figlio di #PERSON_2. #PERSON_2 ha invaso il giardino di #PERSON_3.

Il testo rimane comprensibile. La semantica è preservata per il Machine Learning (Identity Management).

La Diagnostica dell'Anonimizzazione

	Rimozione Dati (Brute Force)	Tagging Standard	Tagging Enumerato
Conformità GDPR	Sì	Sì	Sì (Immune ad adversarial attacks)
Preservazione della Semantica	Persa	Parziale	Eccellente (Unbiased dataset)
Utilità per il Machine Learning	Nulla	Basso	Ideale (Mantiene relazioni logiche)
Livello di Complessità	Bassa	Media	Alta (Richiede modelli NLP avanzati per la classificazione)

Apprendimento



Anonimizzazione per l'apprendimento

- **Anonimizzare in modo che sia preservata la semantica**
 - Si usano soluzioni di identity management
 - processo focalizzato a rimuovere le informazioni dei soggetti coinvolti (nomi, indirizzi email, date) per ottenere un dataset unbiased e normalizzato
 - un documento anonimizzato può non essere utile per altri scopi (es: pubblicazione)
 - Produzione di una data set di vertenze che abbiano:
 - Testi anonimizzati in modo semantico
 - Risultati noti della propensione alla Mediazione
- **Procedimento per anonimizzazione semantica**
 - modello di analisi del testo con cui poter anonimizzare automaticamente i documenti
 - Il modello identifica le occorrenze di dati da rimuovere (es. Nome e Cognome, Codice Fiscale, Luogo, ...)
 - occorrenze individuate sono sostituite con una etichetta anonima (es. #NOME, #CF, #LUOGO, ...)
- **Esempio:**
 - Mario Rossi è nato il 10/04/1990 a Firenze >>>> #NOME è nato il #DATA a #LUOGO

Modello AI vs Training

1. Ingestion (Documenti Fascicolo)

L'input di N documenti complessi e multimodali.



2. Preparazione (Anonimizzazione Semantica)

Conversione in #PERSON_1, #ORGANIZATION_1 per garantire la privacy (GDPR).



3. Inferenza (Rete Neurale)

Il modello BERT/LLM valuta le singole frasi, il contesto e la causalità degli eventi.



4. Output (DSS Report)

Predizione: Propensione (es. 0.00%), Non Propensione (es. 0.01%), Neutro (es. 99.99%).

Spiegazione (XAI): Evidenza delle parole influenti e riferimenti.



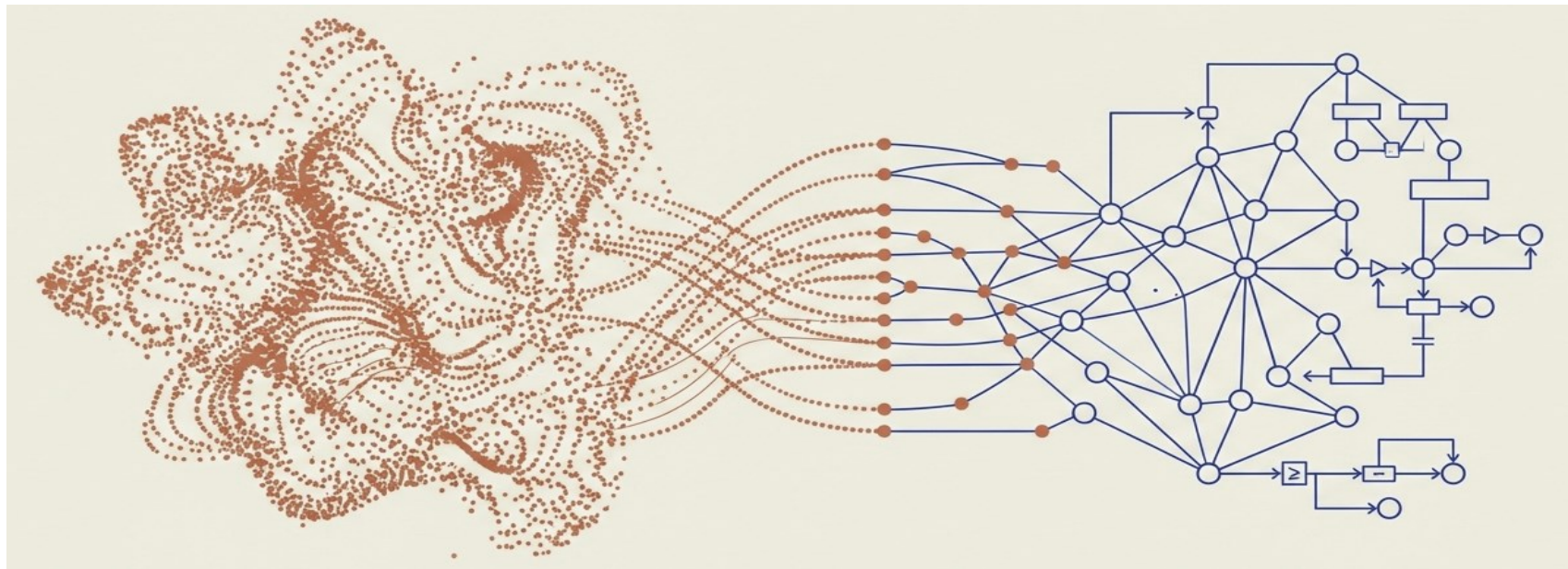
XAI Explained

Sample text: **influential words** want imere
influential words and semse negative words
negative next in the **negative words** ...

Classificazione delle frasi

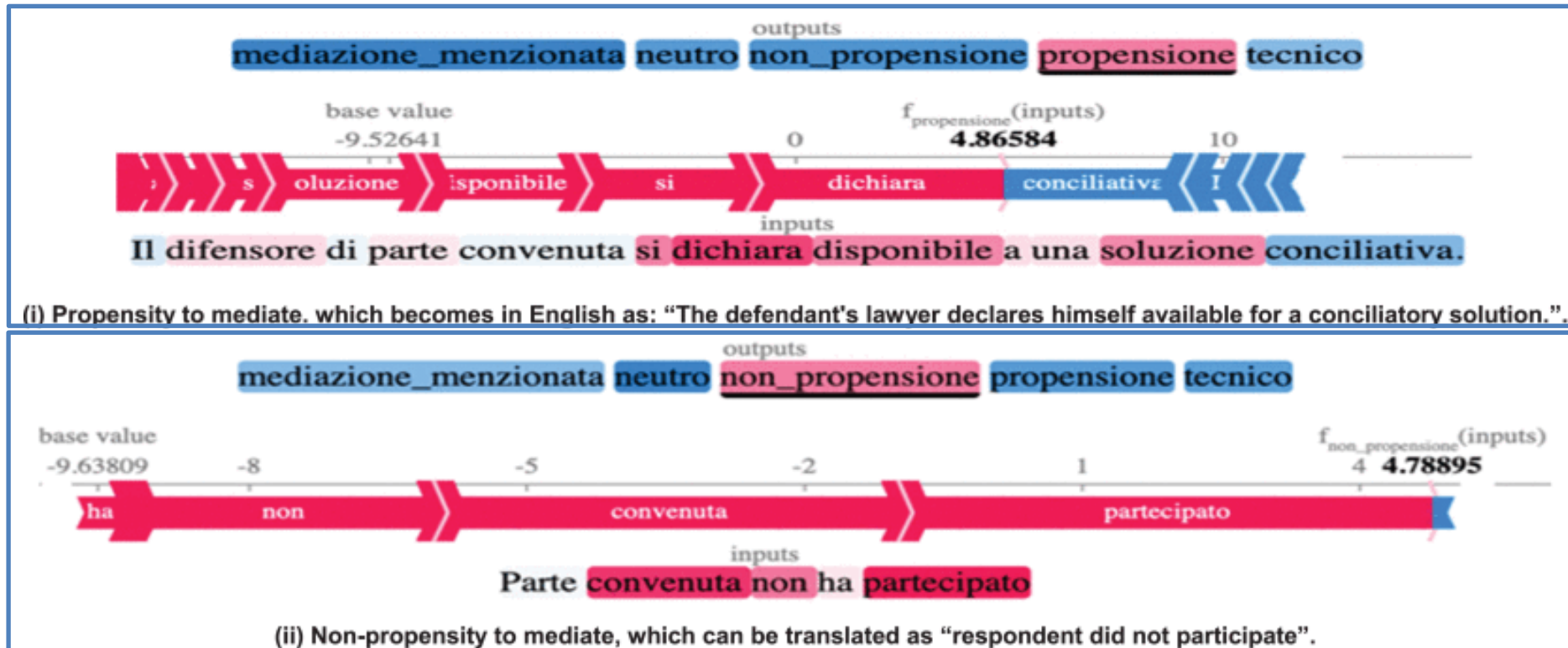
- **Mediazione menzionata:**
 - Sono esplicitamente menzionati documenti di mediazione
 - La mediazione viene disposta dal giudice
 - Viene dichiarata una mediazione in atto
 - Il giudice constata che la mediazione non è stata ancora esperita
 - Le trattative di mediazione sono fallite: non è detto che le parti non fossero propense però
 - Il giudice rileva che la materia del processo è a mediazione obbligatoria
- **Propensione:**
 - Le parti chiedono al giudice il permesso di avviare trattative
 - Entrambe le parti sono assenti al processo
 - Le parti si dichiarano disponibili a intraprendere la mediazione
 - L'attore è assente al processo: accade quando le parti stanno mediando
 - Il processo è dichiarato estinto
- **Non propensione:**
 - Indisponibilità delle parti a mediare
 - Il convenuto è assente al processo: rinuncia a difendersi e non ha interesse a seguire il processo
- **Tecnico:**
 - Presenza di un CTU
 - Richiesta di un CTU
 - Presenza di un CTP
- **Neutro:**
 - Non appartiene a nessuna delle altre classi

NLP vs XAI



Explainable AI, global/local

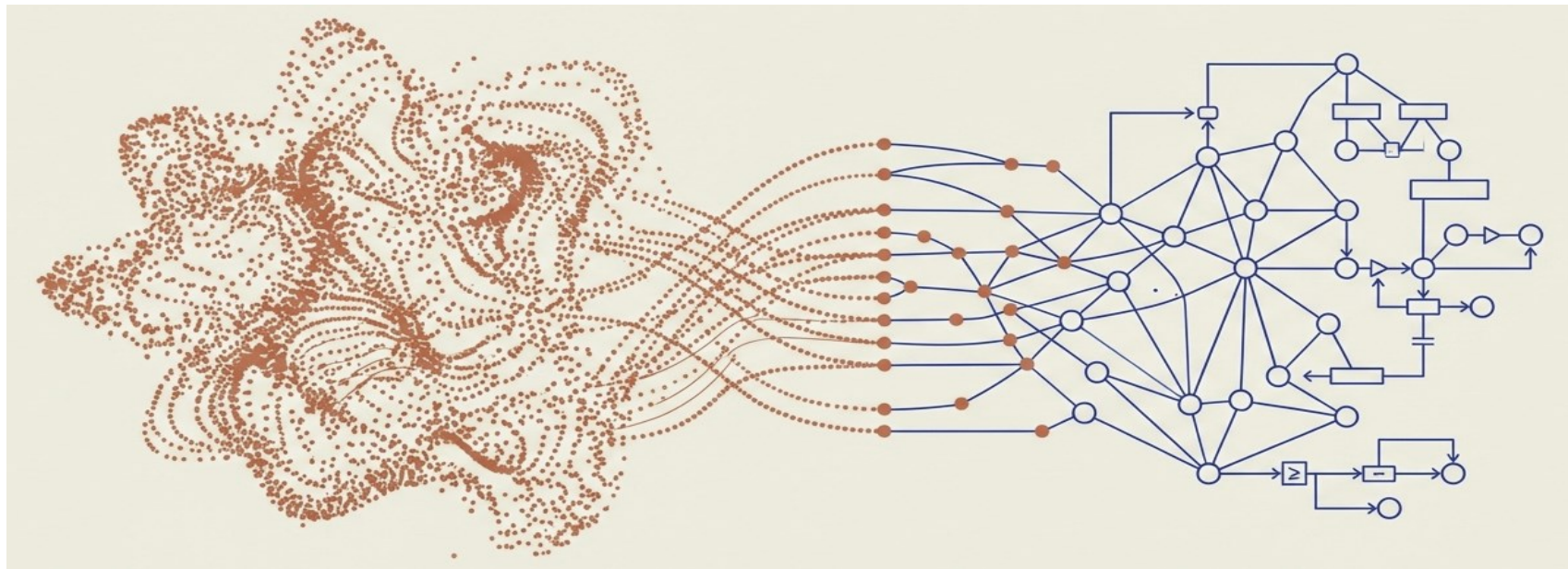
- In passo ulteriore è fornire non solo una classificazione ma anche una spiegazione della classificazione delle singole frasi.
- Per questo si utilizzano tecniche di XAI (global e/o local)



Structure of the course up to now

- Overview
- What is Symbolic, Classification
- Hybrid solutions
- Physically Informed, PINN
- Deep Reinforced Learning and Symbolic at Scale
- From RAG LLM to Agentic LLM ←
- Knowledge \leftrightarrow NN
- MLOps at Scale

Da NLP a LLM



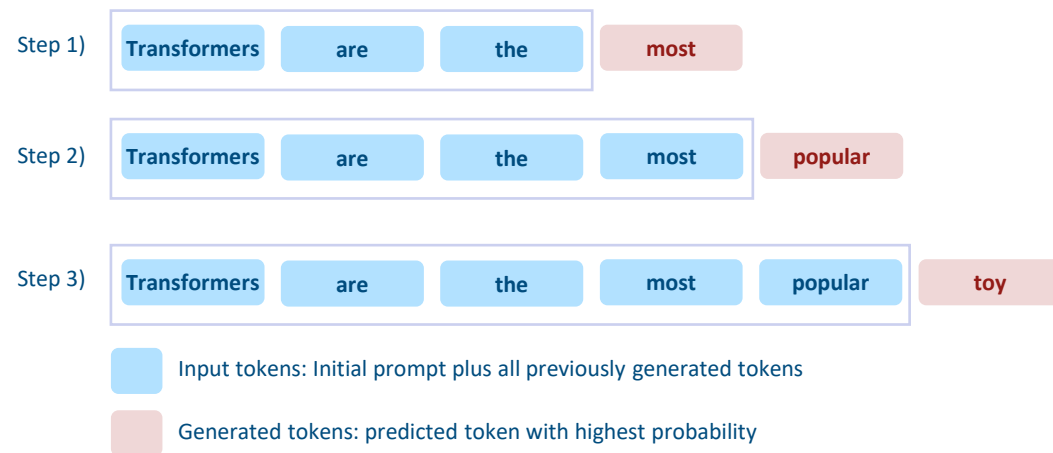
La rivoluzione dei Transformers (2017)

- Hanno sostituito le RNN andando a introdurre il meccanismo di «Attention» per pesare la rilevanza delle parole nelle frasi
- **BERT (Bidirectional Encoder Representations from Transformers) (2018)**,
 - BERT: uno dei primi modelli basati su Transformers a ottenere risultati straordinari per i classici compiti di NLP:
 - riconoscimento, classificazione, disambiguazione, etc.
- **GPT (Generative Pretrained Transformer):** introdotto da OpenAI
 - tra i primi a passare a modelli di auto-regressione per generare testo.
 - **Generazione di risposte in linguaggio naturale**
 - La generazione di risposte, *una sorta di predizione a partire da una sequenza data di parole, il prompt, la richiesta.*

Da BERT ai Large Language Models (LLM)

- I LLM, Large Language Model sono l'evoluzione dei **GPT** (e.g., chat **GPT-4**, **GPT-5**), e si distinguono per la loro scala in termini di:
 - **Numero di parametri:** i LLM hanno miliardi di parametri, per catturare informazioni molto più complesse e dettagliate sul linguaggio.
 - **Dimensione del dataset:** i LLM sono addestrati su dataset enormi, spesso contenenti centinaia di Gbyte-TByte di testo provenienti da varie fonti (libri, articoli, pagine web).
 - **Capacità zero-shot e few-shot:** i LLM, grazie alla loro scala, possono eseguire compiti complessi:
 - Zero-shot: produzione di risposte sulla base della loro conoscenza di base
 - Few-shot: produzione di risposte sulla base di conoscenza aggiuntiva
 - **Alcune architetture di LLM continuano a imparare** ma magari non è sempre cosa buona, anche perché si rischia di virare il modello, e dimenticare...

Modelli di Linguaggio Generativi (GPT)



- L'obiettivo del modello consiste nello stimare il token successivo più probabile in base al prompt e a tutti i token generati in precedenza.
- Dopo aver generato il token successivo, questo viene aggiunto al prompt, estendendo così la sequenza di input per prevedere il token successivo.

- Questo processo iterativo continua fino a quando non viene generato uno speciale token di fine sequenza (`<|endoftext|>`) o quando viene raggiunta una lunghezza massima predefinita.
- Questo processo è chiamato **generazione di testo condizionale**. Ampia gamma di applicazioni: dal completamento di frasi alla generazione di contenuti creativi basati su input specifici dell'utente.

Large Language Models: LLM

➤ Intuizione a partire dal modello **GPT-2**:

- I modelli supervisionati tradizionali si basano su grandi set di dati etichettati, limitandone l'adattabilità
- Relazione tra approcci supervisionati e non supervisionati: i task non supervisionati possono allinearsi quelli supervisionati imparando a prevedere le sequenze di testo.
- I modelli linguistici generativi autoregressivi di grandi dimensioni non supervisionati come GPT-2 (*self-supervised* con 1,5 miliardi di parametri) possono acquisire caratteristiche linguistiche complesse e generalizzare tra le attività senza la necessità di dati specifici etichettati.
- I modelli linguistici ben addestrati possono essere adattati a compiti specifici con un minimo numero di dati etichettati.

➤ Training su vasti corpora di testi:

- Addestrato sul set di dati WebText (milioni di pagine Web curate).
- Impara dalle sequenze del linguaggio naturale per acquisire diversi modelli linguistici.
- Consente al modello di comprendere task complessi come la traduzione, produrre risposte alle domande da dati di testo non elaborati.

➤ Zero-Shot Learning:

- Il modello apprende le istruzioni dell'attività in modo implicito dai dati di testo naturale.
- Apprendimento Zero-Shot: esegue un'attività senza esempi o addestramento su quell'attività specifica (*fine-tuning*), basandosi esclusivamente sul contesto o sulle istruzioni fornite durante l'inferenza.
- GPT-2 può eseguire task senza fine-tuning su set di dati specifici.

Large Language Models: LLM

- Il modello **GPT-3** ha ulteriormente sviluppato e migliorato il concetto GPT-2:
- 175 miliardi di parametri.
 - Addestrato su 300 miliardi di token da vari set di dati: Common Crawl (una vasta raccolta di pagine web), WebText2, Wikipedia...
 - Miglioramenti alle attività di apprendimento contestuale, zero-shot, one-shot e few-shot.
 - Genera testi coerenti e contestualmente rilevanti, che riflettono la comprensione di vari domini e stili.
 - Capace di eseguire ragionamenti aritmetici e di utilizzare correttamente il nuovo vocabolario nel contesto, mostrando abilità cognitive avanzate.

Large Language Models: LLM

➤ Novità introdotte con i successivi modelli **GPT-4** e **GPT-5**:

- **Multimodalità**: il modello accetta **testo e immagini** (sempre generando un output testuale)
- Capacità di ragionamento
- Chain-of-Thoughts (CoT) avanzato: invece di fornire una risposta immediata e diretta, il modello risolve un problema complesso scomponendolo in una sequenza di passaggi logici intermedi. Questo processo graduale permette di arrivare a una risposta finale più precisa e spiegabile.
- Significativi progressi su generazione di codice in vari linguaggi di programmazione, Instruction-Following e gestione di contesti molto lunghi (centinaia di migliaia / milioni di token)
- Primi sviluppi di capacità «Agentic» (GPT-5): abilità del modello di pianificare e svolgere sequenze di azioni (eseguire applicazioni e codice, interrogare basi di dati, fare chiamate web, interazione con pagine web ecc.)

➤ **Sviluppi Futuri:**

- Migliorare capacità agentic
- Multimodalità profonda (testo-immagini-audio-video) in tempo reale

The LLM Landscape — Key Models

A snapshot of major LLMs as of 2024–2025, organised by provider and access model

OpenAI

GPT-4o, GPT-4 Turbo, o1, o3

Access: API / ChatGPT

Closed weights, RLHF-aligned

Anthropic

Claude 3 Opus / Sonnet / Haiku
Claude 4 Opus / Sonnet

Access: API / Claude.ai

Constitutional AI, safety-focused

Google

Gemini Ultra / Pro / Flash
Gemma 2 (open)

Access: API / Gemini app

Multimodal, long context (1M)

Meta

LLaMA 3.1 8B / 70B / 405B

Access: Open weights

Most popular open-source family

Mistral

Mistral 7B, Mixtral 8x7B
Mistral Large

Access: Open weights / API

Mixture-of-Experts efficiency

Cohere

Command R+, Embed, Rerank

Access: Enterprise API

RAG & retrieval-optimised

LLM usage



Strengths

- Minimum infrastructure
- Excellent for general explanations, drafting, brainstorming, and many coding tasks
- Easy to test in one afternoon

Limits

- No guaranteed access to your latest files
- No built-in source selection from large corpora
- You ask the model to remember facts it may never have seen

Critical points of the General Accessible LLM

- Generazione di informazione errata (*Hallucination*), anche a fraintendimenti
- Informazioni non sempre aggiornate e profonde su tutto
- Mancanza di conoscenza in domini specifici forniti dall'utente
- Le risposte spesso non hanno riferimenti precisi ed è difficile verificare
- Hanno una dimensione del prompt limitata, e costosa ... 10 Gbyte di prompt ??
- Rischioso fornire documenti personali / privati ad un LLM pubblico che potrebbe poi incorporarli nella sua futura conoscenza, senza pensare al fatto che stai fornendo files privati che non sai dove vanno a finire e che uso ne viene fatto. Potresti non essere autorizzato a fornire tali file fuori dal tuo PC, dalla tua azienda.

- **ALTERNATIVA:**

- faccio retraining di un LLM in locale
- Costoso e complesso addestrarli (*Fine-Tuning*) su basi di conoscenza specifiche (documentazione tecnica, manualistica, documenti legali, contratti ecc.)



A Neuro-Symbolic Classification

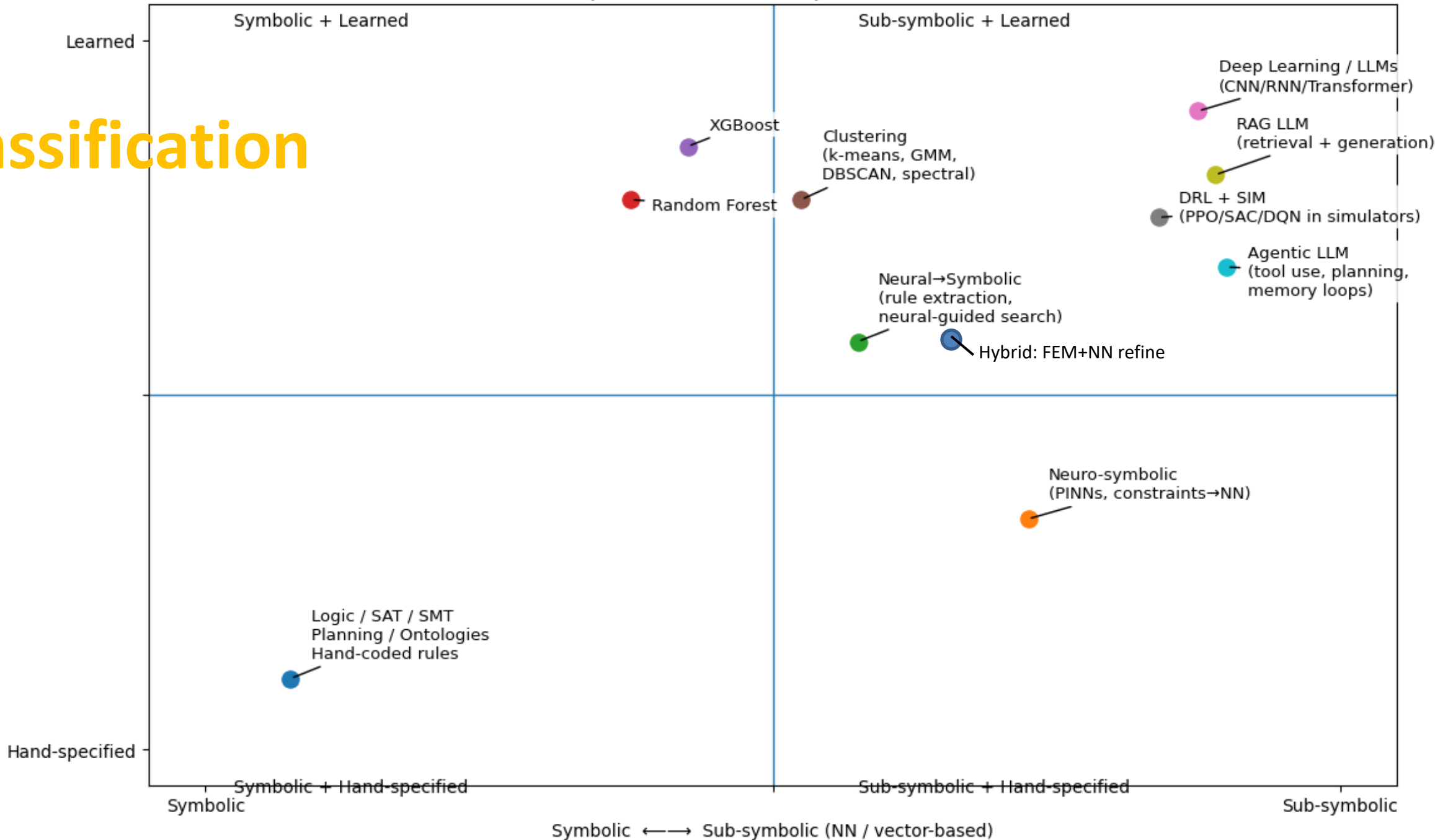
- **Symbolic → Neural (knowledge as bias/constraints)** inject structure into learning:
 - Logic constraints as loss terms (“semantic loss” style)
 - Physics constraints (**PINNs**) and other mechanistic constraints
 - Type systems, ontologies, invariances, monotonicity constraints
 - → improved generalization, robustness, safety/spec compliance.
- **Neural → Symbolic (learning symbols/programs)** extract discrete structure
 - Rule learning / program induction from data
 - Symbol grounding: learn mappings from perception to symbols
 - Model distillation into rules/trees for interpretability
 - → interpretability, compositional generalization, verifiable reasoning.
- **Joint / co-trained**



2D Map of ML/DL Techniques (clean labels)

Classification

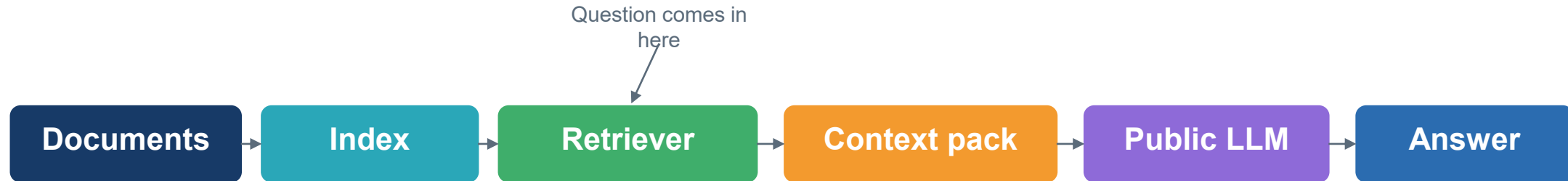
Hand-specified \longleftrightarrow Learned from data



Retrieval Augmented Generation: RAG

- Ulteriori requisiti che applicazioni di AI devono soddisfare in ambiti specifici:
 - Produrre risposte accurate e complete su basi di conoscenza specifiche
 - anche di grandi dimensioni
 - Private e proprietarie
 - Con aggiornamenti frequenti
 - Spiegabilità (Explainability) e tracciabilità delle risposte, avere traccia da dove ha preso queste informazioni nella propria KB
 - Non avere costi di Fine-Tuning, non fare fine tuning, cambiare il LLM se necessario
 - Multimodalità se necessario

Classic Simple RAG



Why this changes the game

Instead of hoping the model internally knows the answer, the system actively fetches evidence that matches the current question.

Important nuance

RAG does not mean a weak model. In practice, many strong systems use a top-tier LLM as the generator and let retrieval supply missing or current facts.

Typical components

Vector store, keyword search, reranker, chunking logic, prompt template, citation formatter, and telemetry.

Comparison: LLM vs RAG, 1

Example question

“Can students submit the final project one week late if they are on the exchange programme?”

A public LLM can reason about the language of the question, but it cannot know your course rule unless the rule is in context.

LLM alone

May answer from generic academic customs, similar institutions, or patterns in training data.

RAG

Retrieves the actual course policy and gives an answer linked to that document.

For “what is true here?” questions, retrieval usually beats generic model memory.

Comparison: LLM vs RAG, 2

LLM-alone answer

Answer:

“The deadline extension is normally granted for mobility programmes.”

Problem:

The statement may sound plausible, but where did it come from? Can a student verify it?

RAG answer

Answer:

“The policy allows a one-week extension for exchange-programme students with prior approval.”

Evidence:

Course handbook §4.2;
assessment policy update, 12
Sept.

Why this matters

- Easier user trust
- Better debugging
- Simpler human review
- More suitable for regulated or high-stakes settings
- Better feedback loops for system improvement

Selected RAG Approach



Why this beats “paste everything into the prompt”

Retrieval can limit context to documents the current user is actually allowed to see.

Deployment flexibility

You can combine enterprise search, private storage, and even a public generator model while keeping the corpus under tighter control.

Nuance

RAG is not automatically private. Privacy comes from secure indexing, access control, logging policy, and model routing decisions.

Comparison: LLM vs RAG, 3

Public LLM alone

Best when:

- task is generic
- no private corpus is needed
- users can manually provide the necessary context

Weak point:

large “everything prompt”
strategies become brittle and
expensive.

RAG

Best when:

- facts are large, private, or dynamic
- only a few documents matter per query
- answers should reference evidence

Strength:

fetch only the relevant context at
runtime.

Fine-tuning

Best when:

- you need new style, format, or policy-following behaviour
- repeated task behaviour should change

Weak point for fresh facts:

updating knowledge can be
slower and more expensive than
re-indexing documents.

Comparison: LLM vs RAG, 4

Course assistant

Questions:

- “What counts toward the final grade?”
- “Which readings are compulsory this week?”

Why RAG wins:

It can fetch the syllabus, reading list, and latest announcements.

Research-lab knowledge bot

Questions:

- “Which dataset version was used in experiment B?”
- “Where is the annotation guideline?”

Why RAG wins:

Answers come from lab notes, repositories, and meeting documents.

Student-services helper

Questions:

- “Who can apply for accommodation support?”
- “What is the current visa document checklist?”

Why RAG wins:

Policies and checklists change; retrieval keeps answers current.

Il Lato Oscuro: Allucinazioni e Rischi Professionali

Precedente: Mata v. Avianca, 2023.
La corte stabilisce che...

La corte stabilisce che di nei
continuzione su ressenire la
precedente, tanisa mpremada,
courtra voido e response fonnati di
contrattualioso.



- **Il Problema:** L'AI genera citazioni, precedenti o fatti plausibili ma inesistenti. Tasso di allucinazione del 16.7% nelle ricerche generaliste.
- **Il Rischio:** Sanzioni disciplinari severe per il deposito di memorie inventate.
- **La Regola d'Oro:** L'AI è un assistente, non un'autorità. La verifica manuale sulle fonti primarie ufficiali è un obbligo deontologico assoluto.

L'Infrastruttura Sicura: LLM vs. RAG

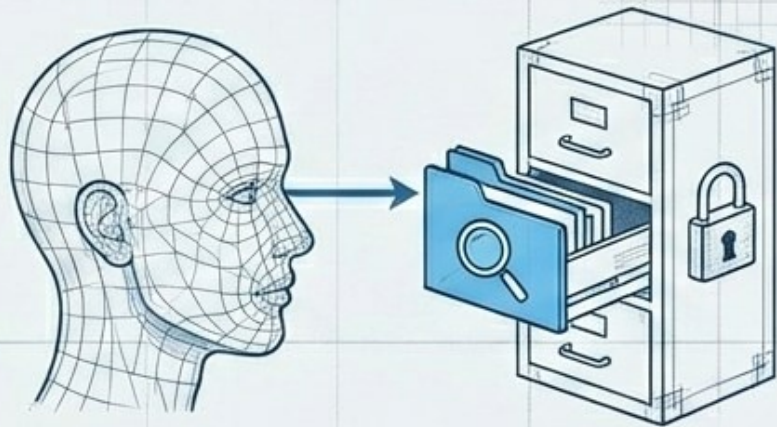
Solo LLM (Modelli Generici)



Ottimo per ideazione e riassunti generici.

- **Criticità:** Rischio elevato di allucinazioni, dati non tracciabili, mancanza di fonti primarie.

RAG (Retrieval-Augmented Generation)



Le risposte sono vincolate a documenti esterni e privati caricati dall'utente.

- **Vantaggio:** Tracciabilità assoluta delle fonti (Evidence Traceability).
- Azzeramento delle allucinazioni 'inventate' e aggiornamento normativo in tempo reale.

Comparison LLM vs RAG-LLM

Usa un LLM Generico Quando...

L'obiettivo è una spiegazione generica o l'ideazione creativa.

I fatti necessari sono già forniti nel prompt dell'utente.

Non sono richieste citazioni o tracciabilità delle fonti.

La velocità di setup è prioritaria rispetto alla precisione assoluta.

Usa l'Architettura RAG Quando...

Le risposte dipendono da documenti esterni e specifici.

Il corpus di dati è privato, voluminoso o in evoluzione.

La tracciabilità delle prove (citazioni) è fondamentale.

Lo stesso compito di conoscenza si ripete su larga scala con utenti diversi.

Comparison Conclusions

Use LLM alone when

- The task is mostly generic explanation or ideation
- The needed facts are already in the user prompt
- Citations are not required
- The domain corpus is tiny or irrelevant
- Speed of setup matters more than deep system quality

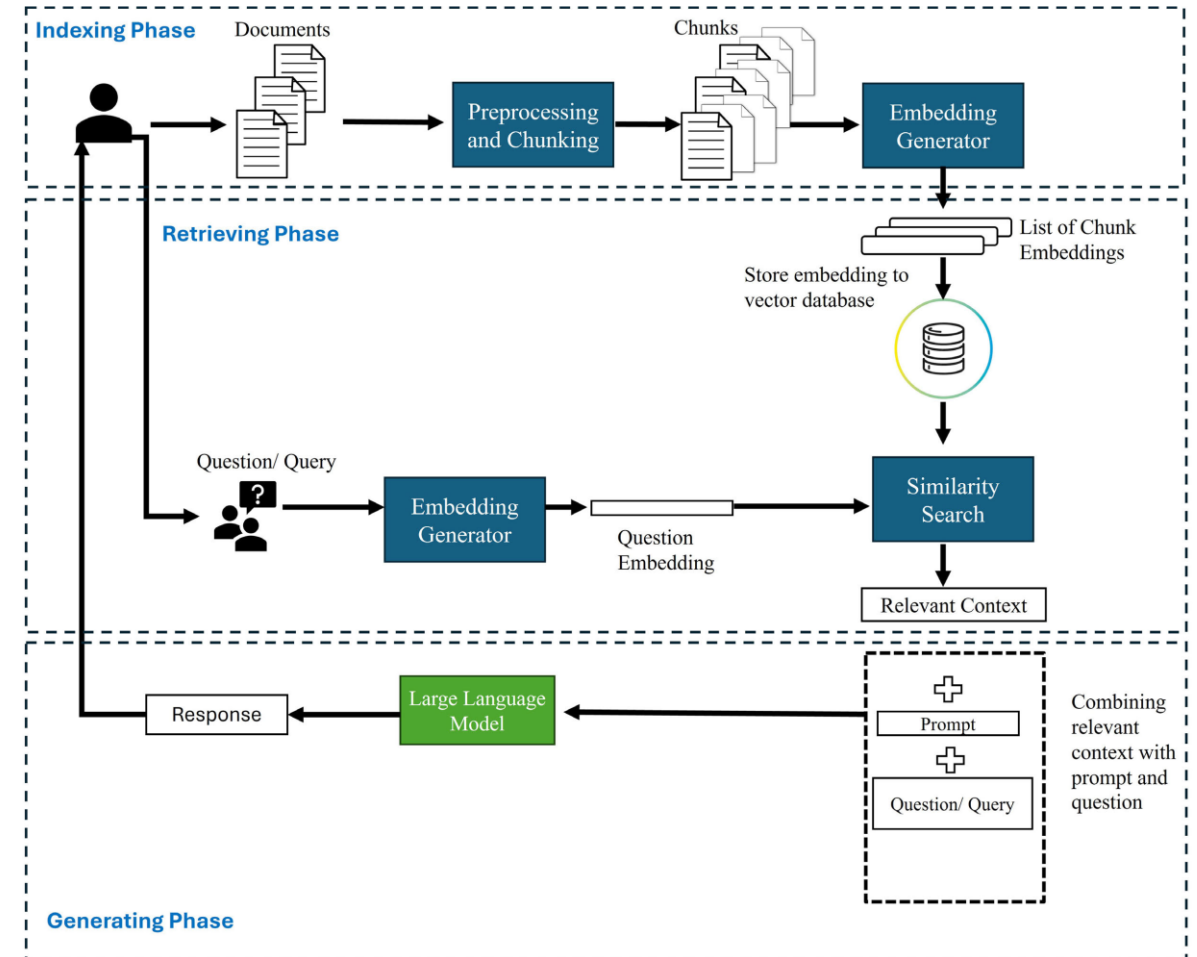
Use RAG when

- Answers depend on external documents
- The corpus is private, large, or changing
- Evidence traceability matters
- Different users should see different material
- The same knowledge task repeats at scale

Retrieval Augmented Generation: RAG

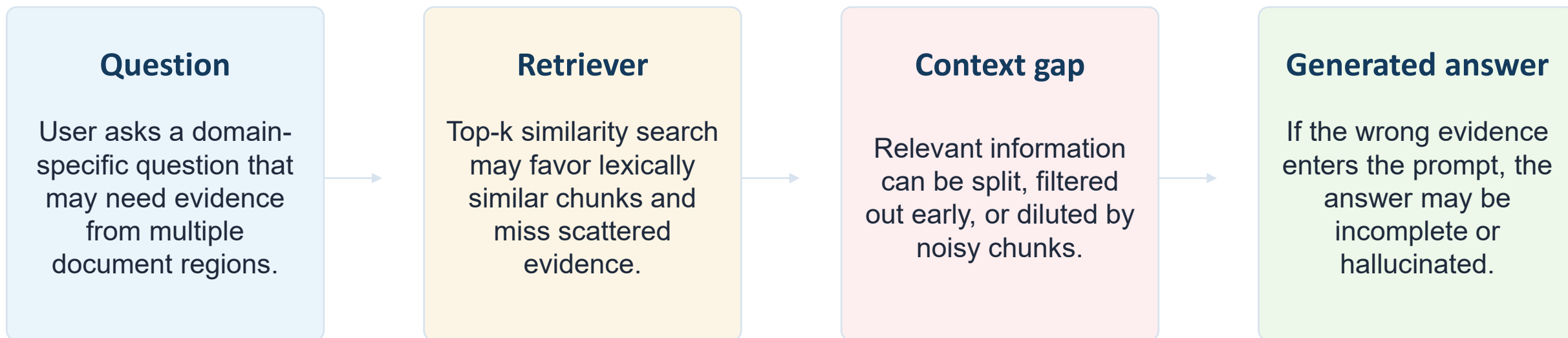
➤ Approccio RAG:

- Una base di conoscenza specifica viene suddivisa in frammenti (*chunks*), in genere consistenti dal punto di vista semantico (*semantic chunking*);
- Per ogni chunk vengono generati i corrispondenti embeddings e indicizzati in un *Vector Database*;
- **Retrieval**: viene generato l'embedding della domanda dell'utente (*query*) e viene fatta una ricerca di similarità nel Vector DB per recuperare i chunks più rilevanti rispetto alla domanda iniziale;
- **Augmented Generation**: I chunks più rilevanti vengono inclusi come contesto insieme alla query iniziale (*augmented prompt*) ed inviati al LLM, che in questo modo è in grado di produrre una risposta più accurata e centrata sulle parti rilevanti della base di conoscenza specifica.



Where traditional RAG can fail

The bottlenecks the authors target



Paper diagnosis

Traditional RAG often retrieves chunks that are lexically similar but semantically misaligned with user expectations. That is why the authors treat the retriever as a critical bottleneck rather than a solved component.

What prior RAG research was trying to fix

A snapshot of the related-work landscape

TABLE 1. Summary of previous research addressing challenges in Retrieval-Augmented Generation (RAG) and related tasks.

Paper	Datasets	Metrics	Proposed Method	Result																												
[25]	Retrieval-Augmented Generation Benchmark (RGB)	Exact String Match (ESM), Normalized Discounted Cumulative Gain(NDCG)	Dartboard	<table border="1"> <thead> <tr> <th>Metric</th> <th>RGB</th> </tr> </thead> <tbody> <tr> <td>ESM</td> <td>85.6</td> </tr> <tr> <td>NDCG</td> <td>0.975</td> </tr> </tbody> </table>	Metric	RGB	ESM	85.6	NDCG	0.975																						
Metric	RGB																															
ESM	85.6																															
NDCG	0.975																															
[26]	PopQA, Biography, PubHealth, Arc-Challenge	Accuracy	CRAG, Self-CRAG	<table border="1"> <thead> <tr> <th>Metric</th> <th>Method</th> <th>PopQA</th> <th>Bio</th> <th>Pub</th> <th>Arc</th> </tr> </thead> <tbody> <tr> <td rowspan="2">Accuracy</td> <td>CRAG</td> <td>39.8</td> <td>47.7</td> <td>9.1</td> <td>25.8</td> </tr> <tr> <td>Self-CRAG</td> <td>49.0</td> <td>69.1</td> <td>0.6</td> <td>27.9</td> </tr> </tbody> </table>	Metric	Method	PopQA	Bio	Pub	Arc	Accuracy	CRAG	39.8	47.7	9.1	25.8	Self-CRAG	49.0	69.1	0.6	27.9											
Metric	Method	PopQA	Bio	Pub	Arc																											
Accuracy	CRAG	39.8	47.7	9.1	25.8																											
	Self-CRAG	49.0	69.1	0.6	27.9																											
[37]	Arxiv, BBCNews, NarrativeQA	ROUGE score	LeanContext	<table border="1"> <thead> <tr> <th>Metric</th> <th>Arxiv</th> <th>BBC</th> <th>Narrative</th> </tr> </thead> <tbody> <tr> <td>ROUGE</td> <td>0.23</td> <td>0.50</td> <td>0.52</td> </tr> </tbody> </table>	Metric	Arxiv	BBC	Narrative	ROUGE	0.23	0.50	0.52																				
Metric	Arxiv	BBC	Narrative																													
ROUGE	0.23	0.50	0.52																													
[17]	PopQA, AmbigNQ, HotpotQA, MMNU	Exact Match (EM), F1	Rewrite-Retrieve-Read	<table border="1"> <thead> <tr> <th>Metric</th> <th>HotpotQA</th> <th>AmbigNQ</th> <th>PopQA</th> <th>MMNU</th> </tr> </thead> <tbody> <tr> <td>EM</td> <td>34.38</td> <td>47.8</td> <td>45.72</td> <td>43.2</td> </tr> <tr> <td>F1</td> <td>45.97</td> <td>60.71</td> <td>49.51</td> <td>-</td> </tr> </tbody> </table>	Metric	HotpotQA	AmbigNQ	PopQA	MMNU	EM	34.38	47.8	45.72	43.2	F1	45.97	60.71	49.51	-													
Metric	HotpotQA	AmbigNQ	PopQA	MMNU																												
EM	34.38	47.8	45.72	43.2																												
F1	45.97	60.71	49.51	-																												
[27]	HotpotQA (HP), 2WikiMultiHopQA (MH), MuSiQue (M), Bamboogle (B), Feverous (F), StrategyQA (S)	Exact Match (EM), F1, Acc+	ITER-RETGEN	<table border="1"> <thead> <tr> <th>Metric</th> <th>HP</th> <th>MH</th> <th>M</th> <th>B</th> <th>F</th> <th>S</th> </tr> </thead> <tbody> <tr> <td>EM</td> <td>45.1</td> <td>33.5</td> <td>26.1</td> <td>40</td> <td>-</td> <td>-</td> </tr> <tr> <td>F1</td> <td>60.4</td> <td>47.4</td> <td>42.0</td> <td>50.7</td> <td>-</td> <td>-</td> </tr> <tr> <td>Acc+</td> <td>72.9</td> <td>58.4</td> <td>41.0</td> <td>60.8</td> <td>70.5</td> <td>74.1</td> </tr> </tbody> </table>	Metric	HP	MH	M	B	F	S	EM	45.1	33.5	26.1	40	-	-	F1	60.4	47.4	42.0	50.7	-	-	Acc+	72.9	58.4	41.0	60.8	70.5	74.1
Metric	HP	MH	M	B	F	S																										
EM	45.1	33.5	26.1	40	-	-																										
F1	60.4	47.4	42.0	50.7	-	-																										
Acc+	72.9	58.4	41.0	60.8	70.5	74.1																										
[45]	Natural Question, Trivia, HotpotQA	Exact Match (EM), F1	Recite Read	<table border="1"> <thead> <tr> <th>Metric</th> <th>HotpotQA</th> <th>Trivia</th> <th>NQ</th> </tr> </thead> <tbody> <tr> <td>EM</td> <td>37.11</td> <td>83.5</td> <td>35.84</td> </tr> <tr> <td>F1</td> <td>48.37</td> <td>88.03</td> <td>49.12</td> </tr> </tbody> </table>	Metric	HotpotQA	Trivia	NQ	EM	37.11	83.5	35.84	F1	48.37	88.03	49.12																
Metric	HotpotQA	Trivia	NQ																													
EM	37.11	83.5	35.84																													
F1	48.37	88.03	49.12																													
[48]	TriviaQA	Exact Match (EM), F1	Credibility-aware Attention Modification (CrAM)	<table border="1"> <thead> <tr> <th>Metric</th> <th>Trivia</th> </tr> </thead> <tbody> <tr> <td>EM</td> <td>36.9</td> </tr> <tr> <td>F1</td> <td>48.45</td> </tr> </tbody> </table>	Metric	Trivia	EM	36.9	F1	48.45																						
Metric	Trivia																															
EM	36.9																															
F1	48.45																															
[21]	Natural Questions, TriviaQA, WebQuestions	Jacc, Hit, Exact Match (EM), F1	REAR	<table border="1"> <thead> <tr> <th>Metric</th> <th>NQ</th> <th>Trivia</th> <th>WebQ</th> <th>SQuAD</th> </tr> </thead> <tbody> <tr> <td>JAcc</td> <td>74.85</td> <td>83.78</td> <td>74.17</td> <td>65.76</td> </tr> </tbody> </table>	Metric	NQ	Trivia	WebQ	SQuAD	JAcc	74.85	83.78	74.17	65.76																		
Metric	NQ	Trivia	WebQ	SQuAD																												
JAcc	74.85	83.78	74.17	65.76																												

Common pattern across prior work

- Rewrite queries to retrieve better evidence.
- Rerank or correct the retrieved documents.
- Compress or summarize context before generation.
- Inject reliability or credibility into the read stage.
- But most methods still depend on retrieval quality upstream.

CA-RAG's novelty is to remove the initial retrieval filter rather than optimize it.

RAG is not Magic for all

Failure 1 Bad chunking

Relevant facts are split badly, so the retriever never sees the right evidence unit.

Failure 2 Poor retrieval

The search stage returns near-matches, not the document that actually answers the question.

Failure 3 Noisy context

Too many weak passages dilute the prompt and distract the generator.

Failure 4 Access mistakes

Permission checks, freshness rules, or metadata filters are implemented incorrectly.

Context-Aware Retrieval Augmented Generation

Core paper: Enrico Collini, Felix Indra Kurniadi, Paolo Nesi, and Gianni Pantaleo
IEEE Access (2025)

Lesson emphasis:

- why traditional RAG can miss the right evidence
- how CA-RAG moves relevance checking after generation
- what the experiments show about accuracy, abstention, and cost

Received 8 September 2025, accepted 23 September 2025,
date of publication 25 September 2025, date of current version 2 October 2025.
Digital Object Identifier 10.1109/ACCESS.2025.3614553



Context-Aware Retrieval Augmented Generation Using Similarity Validation to Handle Context Inconsistencies in Large Language Models

ENRICO COLLINI[✉], **FELIX INDRA KURNIADI**, **PAOLO NESI**[✉], (Life Member, IEEE),
AND GIANNI PANTALEO[✉]

Distributed Systems and Internet Technologies Laboratory (DISIT Lab), Department of Information Engineering, University of Florence, 50139 Florence, Italy
Corresponding author: Paolo Nesi (paolo.nesi@unifi.it)

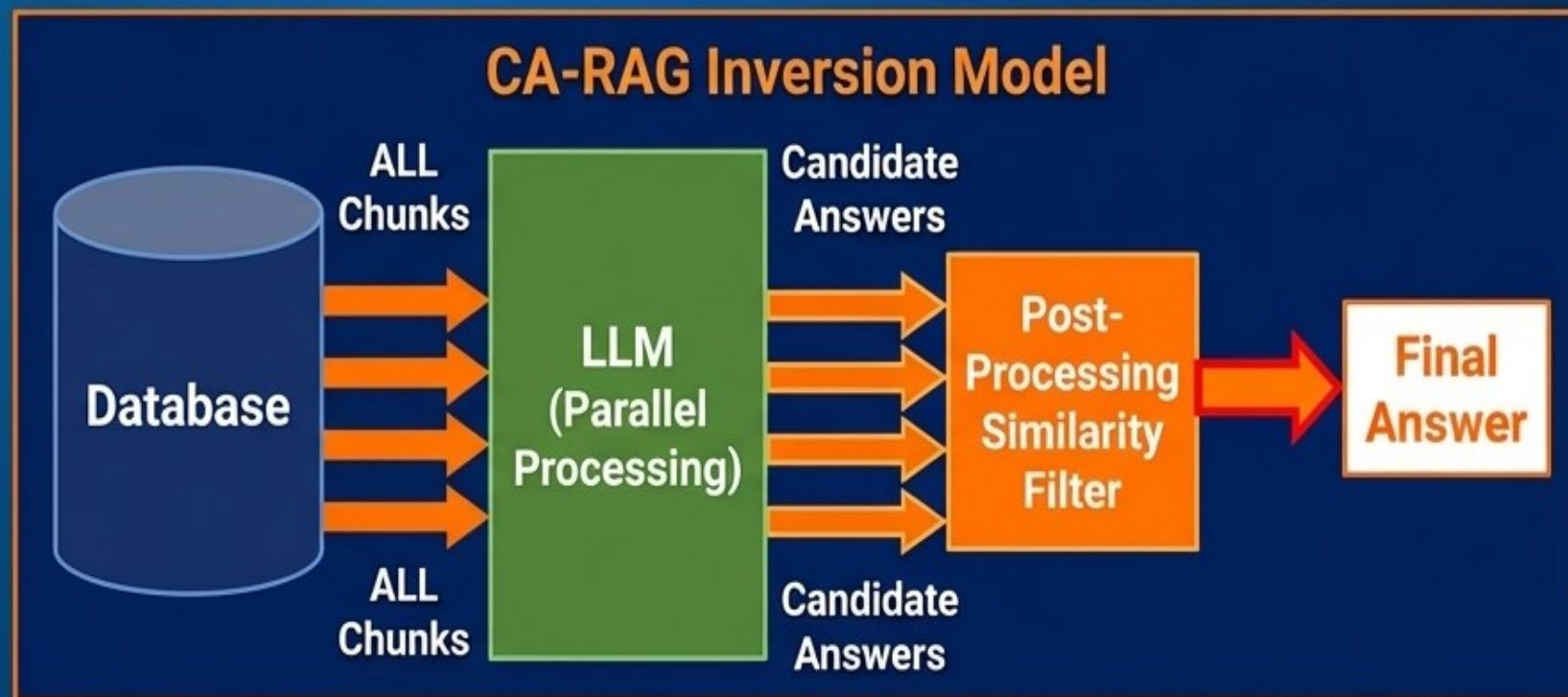
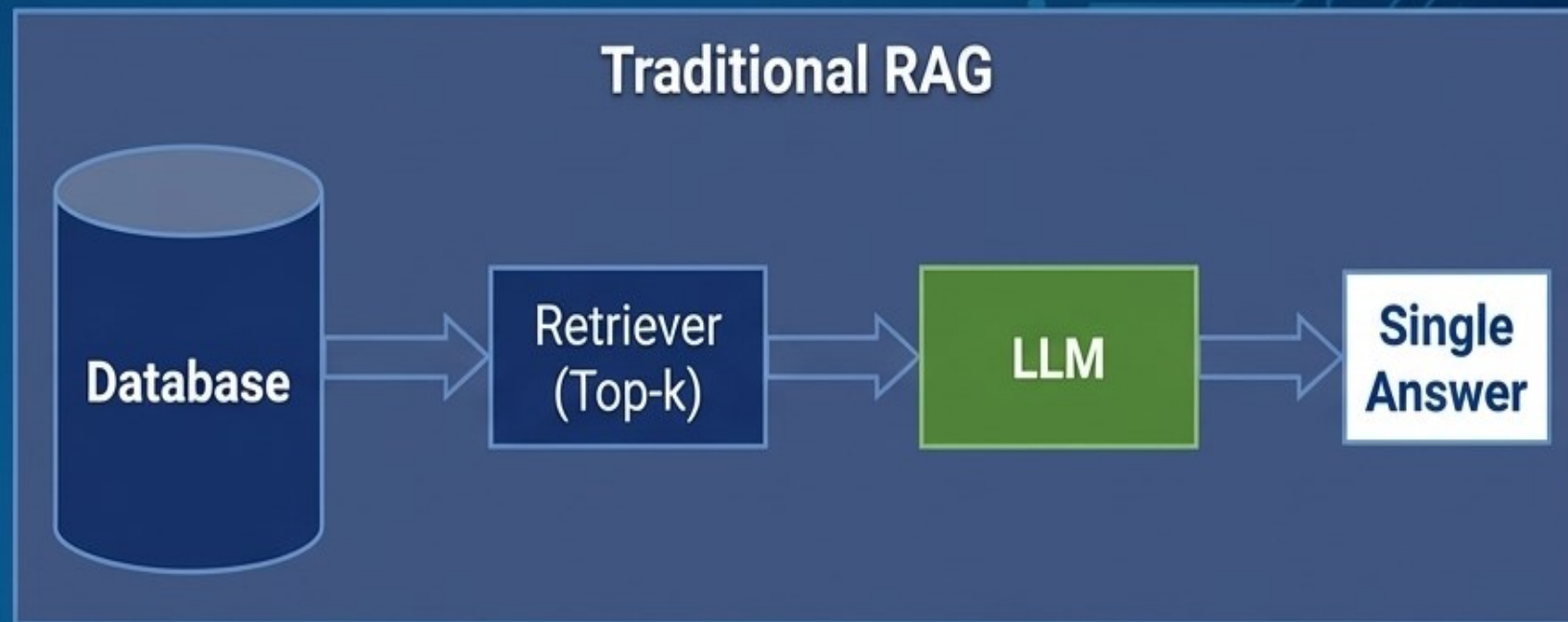
This work was supported by the Ministry University of Research (MUR), project CAI4DSA of FAIR PE1.

ABSTRACT Large Language Models (LLMs) have transformed natural language processing by offering human-like responses. However, issues such as incorrect information (hallucinations) and errors in specific subject areas remain, especially in Retrieval Augmented Generation (RAG) systems. This study introduces a Context-Aware Retrieval Augmented Generation (CA-RAG), which simplifies the process by removing the need to separately find relevant chunks of a document. Instead, after dividing the document into chunks, both the question and chunks are directly given to the LLMs to produce answers. The method then focuses on improving the answers through additional post-processing, aiming to reduce errors and make the answers more relevant to the question. To evaluate the effectiveness of CA-RAG, two scenarios have been designed. Scenario 1 involved experiments using widely adopted and recognized benchmark datasets, such as TriviaQA, Natural Questions, AmbigQA and Stanford Question Answering Dataset (SQuAD). In this context, the proposed CA-RAG method, combined with similarity measure (either cosine similarity or dot

We will treat the paper as both a research study and a design pattern for grounded QA.

The Paradigm Shift: Inverting the Filter

- **The Hypothesis:** Eliminate the **initial vector-similarity** retrieval filter **entirely**.
- **The Mechanism:** Provide **ALL** extracted chunks directly to the LLM. Generate **candidate answers for every chunk**.
- **The Shift:** Move the **relevance judgment** from a dumb vector search to the LLM's own **interpretative ability**. **Filter answers, not documents**.



Traditional RAG vs. CA-RAG

Same goal, different place for relevance judgment

Traditional RAG

Input to generator

Question + retrieved top-k chunks

Dependency

Strong dependency on retriever quality

Main risk

Missing evidence before generation even begins

Strength

Usually faster and easier to scale

Best fit

Latency-sensitive systems and larger corpora

CA-RAG

Input to generator

Question + each candidate chunk in turn

Dependency

Weakens dependence on an upstream retriever

Main risk

Higher compute cost and many LLM calls

Strength

Can abstain when answer-context similarity is low

Best fit

Accuracy-critical QA over moderate-size corpora

CA-RAG at a glance

Three stages: indexing, generation, post-processing

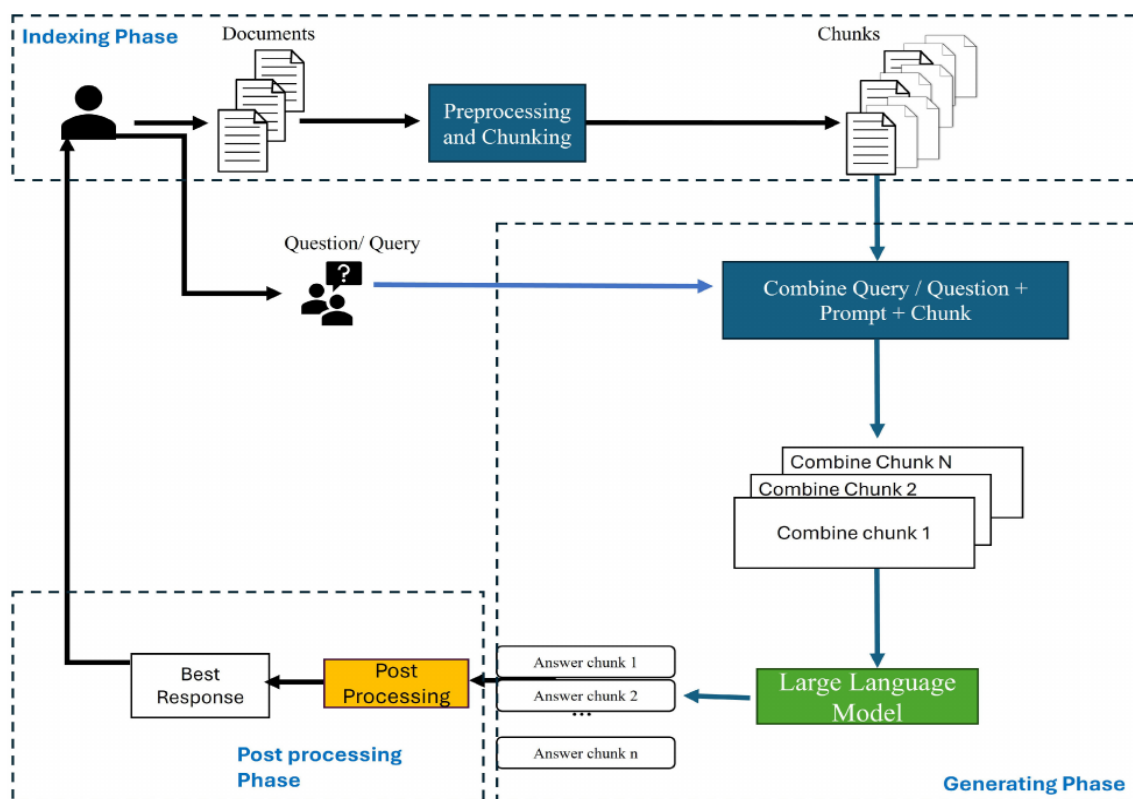


FIGURE 2. Context aware retrieval augmented generation.

What changes relative to RAG?

Indexing

Pre-process and chunk documents, including a semantic chunking option.

Generation

The LLM answers separately for each chunk instead of first picking top-k chunks.

Post-processing

Similarity validation selects or rejects candidate answers based on answer-context alignment.

Key design choice
Move the final relevance check after generation.

Step 1: semantic chunking

Plain-language meaning of Algorithm 1

Algorithm 1 Semantic Chunking

```
0: Input: Text  $T$ , Threshold  $\theta$ 
0: Output: List of Chunks  $C$ 
0:  $S \leftarrow \text{Split\_Sentences}(T)$  {Split text into sentences}
0:  $C \leftarrow []$  {Initialize list of clusters}
0:  $\text{CurrentCluster} \leftarrow []$  {Initialize current cluster}
0:  $E_{\text{prev}} \leftarrow \text{NULL}$  {Initialize previous embedding}
0: for each sentence  $s \in S$  do
0:    $E \leftarrow \text{Embed}(s)$  {Convert sentence into embedding}
0:    $E_{\text{normalized}} \leftarrow \frac{E}{\|E\|}$  {Normalize embedding}
0:   if  $E_{\text{prev}} \neq \text{NULL}$  then
0:      $\text{Similarity} \leftarrow E_{\text{normalized}} \cdot E_{\text{prior}}$  {Compute cosine
0: similarity}
0:     if  $\text{Similarity} < \theta$  then
0:        $C \leftarrow C \cup \{\text{CurrentCluster}\}$  {Store current
0: cluster}
0:        $\text{CurrentCluster} \leftarrow [s]$  {Start new cluster}
0:     else
0:        $\text{CurrentCluster} \leftarrow \text{CurrentCluster} \cup \{s\}$ 
0: {Add to current cluster}
0:     end if
0:   else
0:      $\text{CurrentCluster} \leftarrow \text{CurrentCluster} \cup \{s\}$  {Add
0: first sentence to cluster}
0:   end if
0:    $E_{\text{prev}} \leftarrow E_{\text{normalized}}$  {Update previous embedding}
0: end for
0:  $C \leftarrow C \cup \{\text{CurrentCluster}\}$  {Store final cluster}
0: return  $C = 0$ 
```

Each sentence is then transformed into an embedding (E) using the spaCy library [53]. The embeddings are normalized by dividing each E by its Euclidean norm ($\|E\|$), which can be seen in the following equation:

$$E_{\text{normalized}} = \frac{E}{\|E\|} \quad (1)$$

The similarity between the current normalized embedding ($E_{\text{normalized}}$) and the previous one (E_{prior}) is calculated to find their similarity:

$$\text{Similarity} = E_{\text{normalized}} \cdot E_{\text{prior}} \quad (2)$$

Here, $E_{\text{normalized}} \cdot E_{\text{prior}}$ shows the dot product of the two normalized vectors. If the similarity value is below a predefined threshold (θ), the sentence is stored in a new cluster (C). Otherwise, it is added to the current cluster. This iterative process continues until all sentences are processed, creating semantically coherent chunks.

The generated chunks become the input for the next stages. Unlike traditional RAG workflows (that use algorithms such as sparse or dense retrievers in the retrieving stage), the proposed method skips the retrieval step and goes directly to the generation stage. By skipping retrievers, the approach reduces complexity of this phase, while maintaining better

How the chunker works

- Split the document into sentences.
- Embed each sentence using spaCy.
- Compare each sentence embedding with the previous one using cosine similarity.
- If similarity falls below a threshold, start a new chunk; otherwise keep building the current chunk.

Why this matters

Semantic chunking tries to preserve topical coherence better than naive text splitters, which the authors argue can help keep context meaningful.

Steps 2 and 3: generate, validate, abstain

Why post-processing is the paper's real differentiator

2) GENERATION STAGE

In the generation stage, each context chunk (c) is combined with a prompt (p) and the question (q), producing n chunk-based answers. These answers are generated using Zephyr [54], which was chosen for its ability to generate context-aware responses efficiently. The generated chunk-based answers are then passed into the post-processing stage.

3) POST-PROCESSING STAGE

In the post-processing stage, two different techniques are used. Both techniques use a similarity function (either cosine similarity or dot product) to evaluate how relevant each answer is with respect to the context. The first technique (named as Technique 1 in the following) identifies the best answer, a^* , based on the similarity between the generated answer and the question (i.e., by taking the highest similarity score):

$$a^* = \arg \max_{a_i \in A} (\text{sim}(a_i, Q)) \quad (3)$$

where a^* is the best answer, $A = \{a_1, a_2, \dots, a_n\}$ is the set of given answer, Q is the question and $\text{sim}(x, y)$ denotes a similarity function instantiated as either cosine similarity or dot product.

The second technique (named as Technique 2 in the following) removes answers which are not relevant: if the similarity between a generated answer and its context is below a threshold (θ), the system either marks the answer as "not relevant" and remove it completely. In this way, the set of relevant answers is obtained:

$$A_{\text{filtered}} = \{a_i \in A \mid \text{sim}(a_i, C) \geq \theta\} \quad (4)$$

$$\forall a_i \in A, \text{ if } \text{sim}(a_i, C) < \theta, a_i \notin A_{\text{filtered}} \quad (5)$$

Subsequently, the generated answer with the highest similarity score is chosen as the best answer, a^* :

$$A_{\text{sorted}} = \text{Sort}(A_{\text{filtered}}, \text{sim}(a_i, C), \text{descending}) \quad (6)$$

$$a^* = \arg \max_{a_i \in A_{\text{sorted}}} (\text{sim}(a_i, C)) \quad (7)$$

where $A = \{a_1, a_2, \dots, a_n\}$ is the set of all generated answers, C is the context, θ is the similarity threshold, and A_{filtered} is the set of filtered relevant answers. Both these post-processing techniques have been considered for evaluation (as part of the CA-RAG setup described in point 2) of Section IV-D). However, only Technique 2 (i.e., CA-RAG with similarity calculated between generated answer and context/chunk) has been considered in the presentation of the experimental results. The rationale behind the adoption of the only Technique 2 relies on the fact that, since the LLM generates each candidate answer conditioned on both the question and the context chunk, we may assume that the user intent is already included and taken

Technique 1

Pick the answer whose similarity to the question is highest.

Technique 2

Filter out answers whose similarity to their source chunk is below a threshold; then choose the best remaining answer.

The paper reports results for Technique 2 because it acts as a hallucination filter and enables conservative abstention ("no context provided") for out-of-context questions.

Teaching interpretation

- CA-RAG does not say every answer is better; it says badly grounded answers should be rejected more often.
- That makes the method especially relevant in high-stakes domains where "I don't know" can be preferable to a confident fabrication.

Step 1: Intelligent Indexing & The Chunking Dilemma



Fixed-Size Chunking

...the quick brown fox jumps over the lazy d...
deals. And the thar wite lures are inind as
wish as a iaane or e quick brown fox jumps
laught had the em f white ders, and in-
thon using the book tional interneetions,
and preiaeding to th erity is mucking....



- **Pros:** Fast, predictable computational load.
- **Cons:** Randomly slices semantic boundaries, mixes unrelated topics into single chunks.

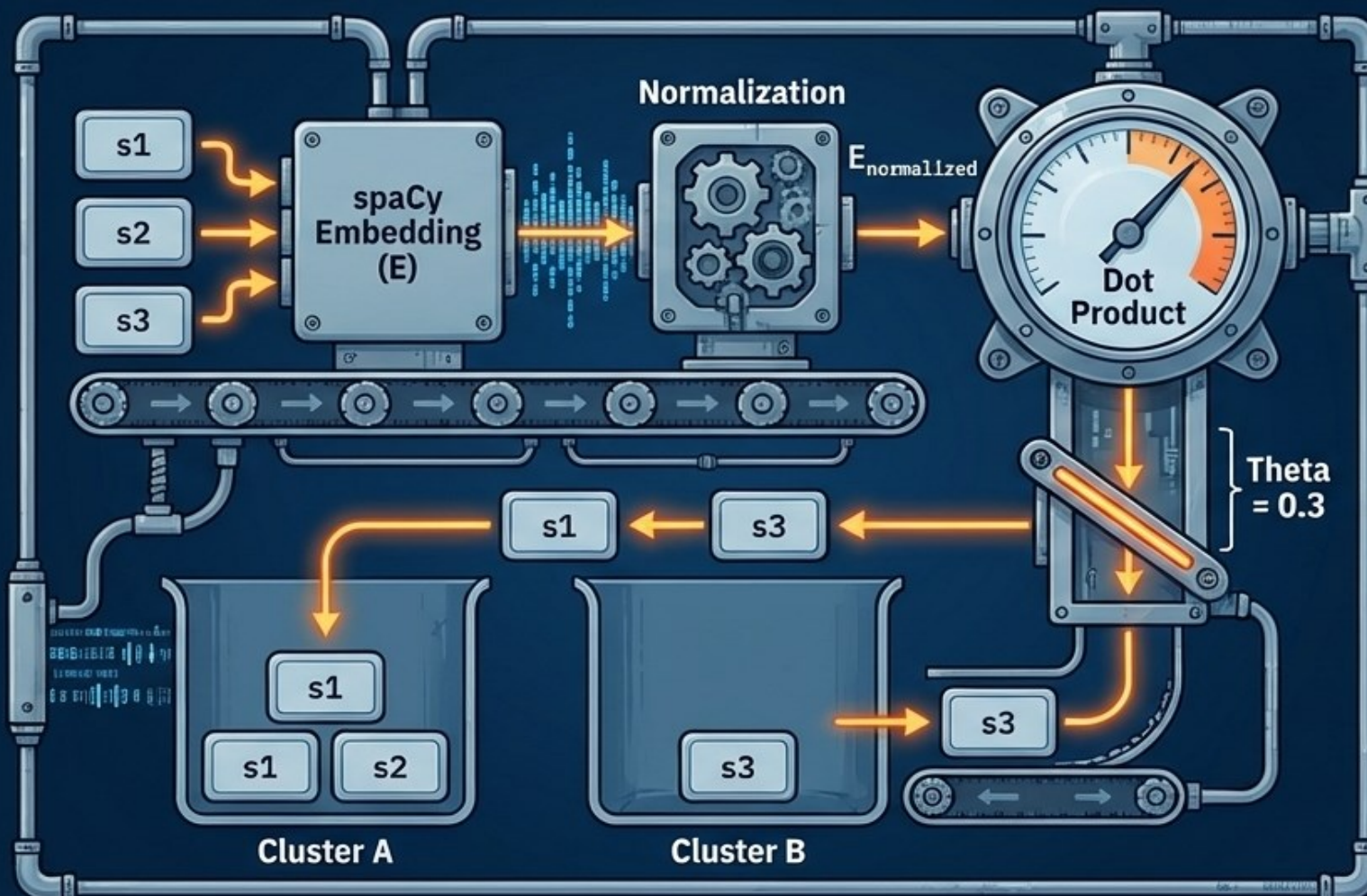
Semantic Chunking



- **Pros:** Preserves contextual meaning, completely reduces noise.
- **Cons:** Variable chunk sizes, computationally heavier during indexing.

Traditional fixed text splitters fundamentally degrade the context before the LLM ever sees it.

Algorithm 1: Semantic Chunking Process



Deep Dive: The Semantic Chunking Algorithm

Normalization: $E_{\text{normalized}} = E / ||E||$

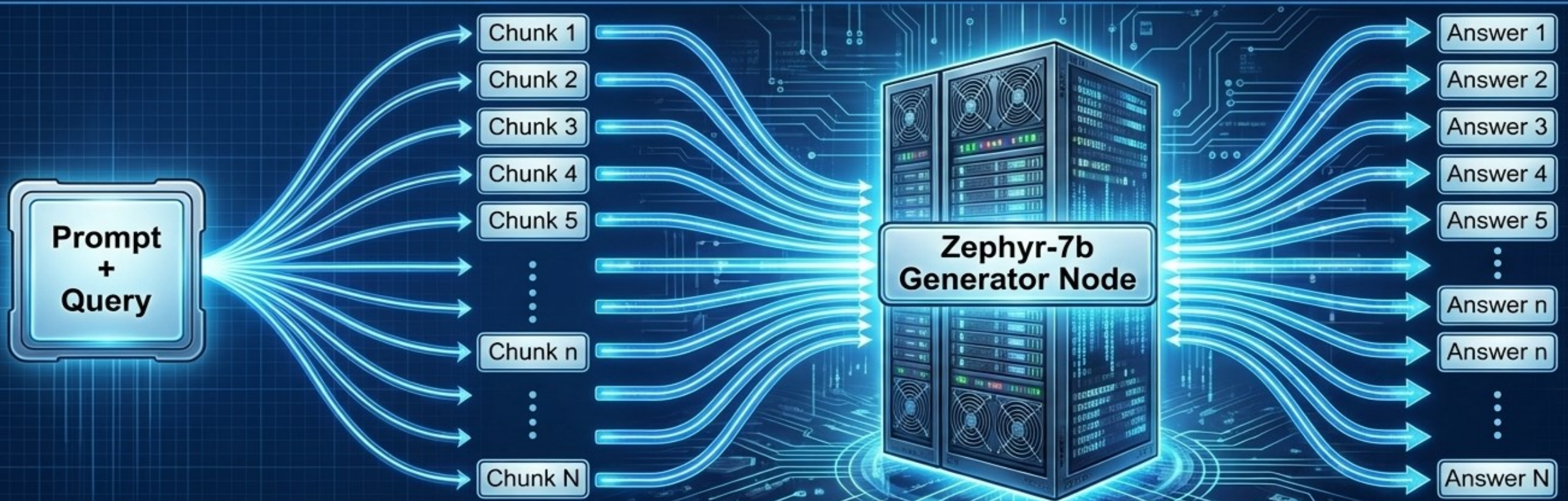
Similarity: $E_{\text{normalized}} * E_{\text{prior}}$

Logic:

- If similarity < Theta: initiate a new chunk.
- Else: append to current chunk.

Empirical Threshold: Theta = 0.3 prevents weak semantic connections from forcing bad clusters, as true paraphrases typically score 0.7-0.9.

Step 2: Parallel Context-Aware Generation



Step 2: Parallel Context-Aware Generation

Bypassing Retrieval

No Top-k selection. Every semantically distinct chunk is given a chance to answer the query independently.

The Engine

Zephyr-7B-Beta chosen for context-aware efficiency with a 4096 token window.

Constraints

Max chunk length restricted to 512 tokens. Generated output strictly capped at 100 tokens to force concise, filterable candidate answers.

Step 3: Post-Processing Similarity Validation

Technique 1 (Baseline - comparing to Query):

$$a^* = \operatorname{argmax}(sim(a_i, Q))$$

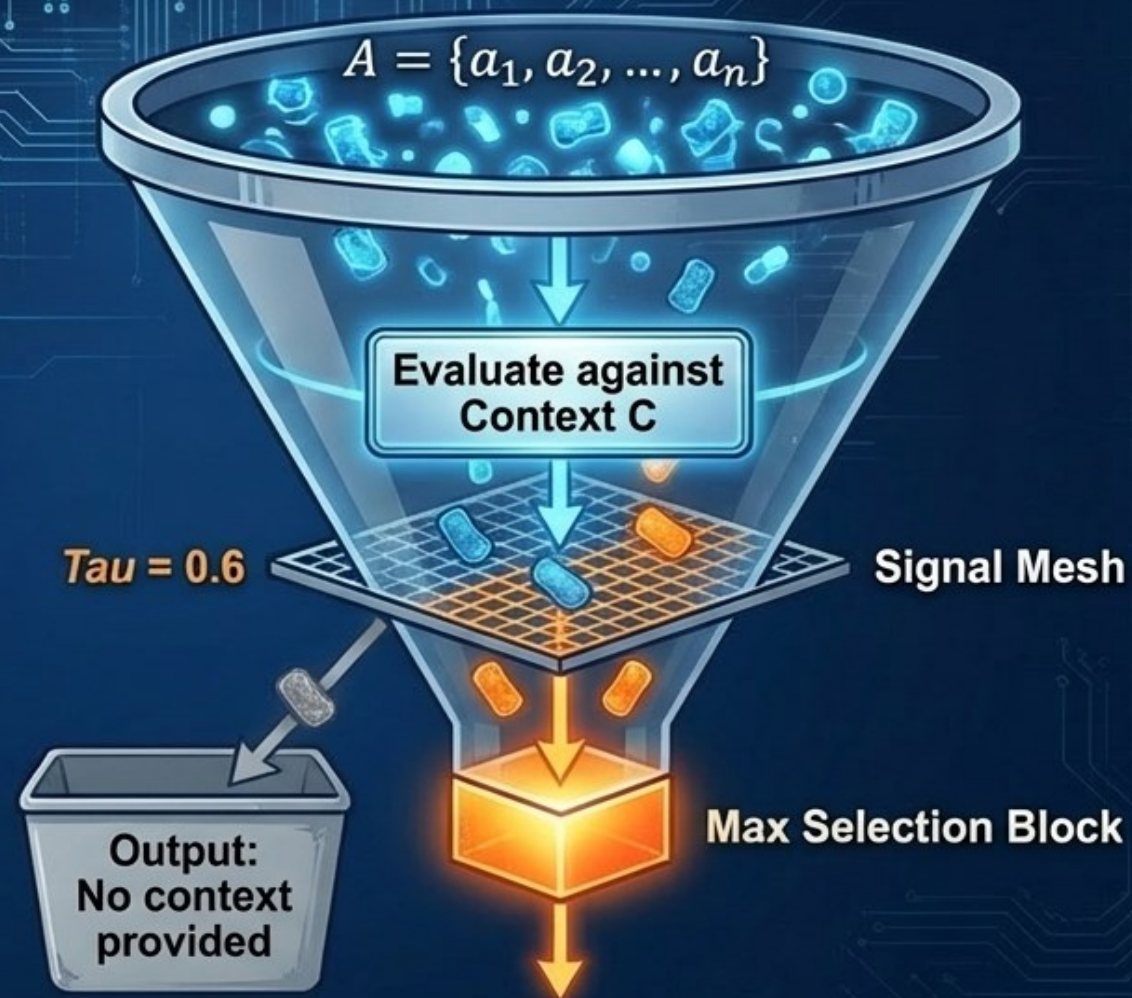
Technique 2 (Chosen Method - comparing to Context):

$$A_{filtered} = \{a_i \in A \mid sim(a_i, C) \geq \tau\}$$

Final Selection: $a^* = \operatorname{argmax}(sim(a_i, C))$

Why Technique 2?

The LLM's generation already incorporates user intent. Validating against the exact context chunk acts as a strict, mathematical hallucination filter.



Experimental Design: The Evaluation Matrix

Scenario 1: Standard Benchmarks

Datasets:

~10k pairs across TriviaQA (long context), NaturalQA, SQuAD (short/explicit), AmbigQA.

Evaluation Metric:

BERTScore (Semantic F1, Precision, Recall).

Purpose:

Establish baseline architectural viability against current literature.

Scenario 2: Real-World Domains

Datasets:

Custom full scientific papers (Bike sharing, Landslides, etc.).

Evaluation Metric:

Human domain-expert evaluation (1-3 evaluators).

Query Types:

Specific, General, and strictly Out-of-Context questions to explicitly test hallucination resistance.

What the datasets are actually testing

Each dataset stresses a different kind of QA behavior

Benchmark datasets (Scenario 1)

SQuAD

Explicit answer spans in relatively short contexts.

TriviaQA

Longer context and more cross-sentence reasoning.

Natural Questions

Deeper comprehension beyond simple span matching.

AmbigQA

Ambiguous questions with multiple valid answers.

Custom dataset (Scenario 2)

Five scientific-paper topics: bike sharing predictions, landslide prediction, reputation assessment, scenario editor, and thermal camera for people counting.

For each topic, the authors created 3 specific questions + 3 general questions + 3 out-of-context questions.

Pedagogical point
Scenario 2 is closer to how QA works over real technical documents.

How the experiments are organized

Two scenarios, different goals

Scenario 1 - benchmark QA

- Datasets: TriviaQA, Natural Questions, SQuAD, AmbigQA.
- Goal: compare CA-RAG with multiple RAG baselines under controlled conditions.
- Metric: BERTScore precision, recall, and F1.
- Question: does CA-RAG still help on standard QA benchmarks?

Scenario 2 - custom scientific-paper QA

- Domain-specific documents from five scientific papers.
- Question types: specific, general, and out-of-context.
- Metric: manual evaluation using TP, FP, TN, FN, then accuracy / precision / recall / F1.
- Question: can CA-RAG reject unsupported answers in realistic long-document settings?

Shared configuration: Zephyr-7B-Beta, up to 512-token chunks, up to 512 new tokens, MiniLM + Contriever embeddings, RAG-Rerank baselines, and 2 x RTX 4090 GPUs.

Scenario 1: The Context-Length Divide

Scenario 1: The Context-Length Divide

Key Findings:

- **Long-Context Mastery:** CA-RAG vastly dominates TriviaQA, achieving a peak F1 score of 0.9089 using fixed-size/Contriever.
- **The Semantic vs. Fixed Divide:** RAG with rerankers requires fixed-size chunks so the reranker can rescue relevant sentences from cluttered noise. CA-RAG's exhaustive parallel generation requires no such rescue.
- **Synthesis:** Traditional RAG is sufficient for short, explicit lookups (SQuAD). For complex, cross-sentence reasoning over long documents, CA-RAG is definitively superior.

TriviaQA
(Long Context)

0.9089

SQuAD
(Short Context)

Traditional RAG +
Reranker Advantage

CA-RAG
Advantage

Scenario 1 results: benchmark datasets

CA-RAG is strongest where longer context matters more

Chunking Method	Method	Embedding Model	Trivia QA			Natural QA			SQuAD			Ambig QA		
			Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Semantic	RAG	miniLM	0.8536	0.8158	0.8334	0.8737	0.8448	0.8582	0.8852	0.8465	0.8649	0.8311	0.8069	0.8179
	RAG	Contriever-Msmarco	0.8511	0.8338	0.8413	0.8909	0.8970	0.8930	0.8964	0.9130	0.9039	0.8413	0.8453	0.8420
	RAG-Rerank	miniLM	0.8545	0.8382	0.8452	0.8914	0.8974	0.8934	0.8972	0.9141	0.9049	0.8399	0.8464	0.8419
	RAG-Rerank	Contriever-Msmarco	0.8545	0.8383	0.8453	0.8914	0.8974	0.8934	0.8971	0.9140	0.9048	0.8400	0.8465	0.8420
	CA-RAG without post processing	miniLM	0.8443	0.8185	0.8303	0.8127	0.8037	0.8076	0.8169	0.8179	0.8170	0.8023	0.7911	0.7960
	CA-RAG with post processing Cosine Similarity	miniLM	0.8547	0.8235	0.8380	0.8827	0.8561	0.8684	0.8958	0.8578	0.8759	0.8342	0.8129	0.8226
	CA-RAG with post processing Cosine Similarity	Contriever-Msmarco	0.8449	0.8829	0.8628	0.8571	0.8840	0.8695	0.8580	0.8969	0.8764	0.8141	0.8364	0.8242
	CA-RAG with post processing Dot product	miniLM	0.8454	0.8826	0.8629	0.8571	0.8840	0.8695	0.8580	0.8969	0.8764	0.8141	0.8364	0.8243
CA-RAG with post processing Dot product	Contriever-Msmarco	0.8583	0.8693	0.8628	0.8567	0.8566	0.8558	0.8599	0.8717	0.8651	0.8280	0.8359	0.8309	
Fixed-Size	RAG	miniLM	0.8404	0.8206	0.8304	0.8822	0.8848	0.8835	0.8792	0.8893	0.8842	0.8417	0.8356	0.8386
	RAG	Contriever-Msmarco	0.8404	0.8206	0.8304	0.8822	0.8848	0.8835	0.8792	0.8893	0.8842	0.8417	0.8356	0.8386
	RAG-Rerank	miniLM	0.8594	0.8372	0.8471	0.9071	0.9227	0.9141	0.8851	0.9009	0.8929	0.8553	0.8538	0.8533
	RAG-Rerank	Contriever-Msmarco	0.8594	0.8371	0.8470	0.9071	0.9227	0.9141	0.8972	0.9009	0.8981	0.8553	0.8538	0.8533
	CA-RAG without post processing	miniLM	0.8594	0.8738	0.8654	0.8721	0.8847	0.8774	0.8698	0.8957	0.8819	0.8229	0.8415	0.8312
	CA-RAG with post processing Cosine Similarity	miniLM	0.8951	0.9084	0.9007	0.8842	0.8986	0.8904	0.8817	0.9107	0.8953	0.8508	0.8723	0.8605
	CA-RAG with post processing Cosine Similarity	Contriever-Msmarco	0.8950	0.9087	0.9008	0.8841	0.8986	0.8903	0.8815	0.9108	0.8952	0.8496	0.8721	0.8597
	CA-RAG with post processing Dot product	miniLM	0.8870	0.8943	0.8896	0.8506	0.8465	0.8477	0.8517	0.8599	0.8550	0.8473	0.8592	0.8522
CA-RAG with post processing Dot product	Contriever-Msmarco	0.9075	0.9120	0.9089	0.8700	0.8726	0.8704	0.8756	0.8891	0.8816	0.8609	0.8714	0.8652	

What to notice

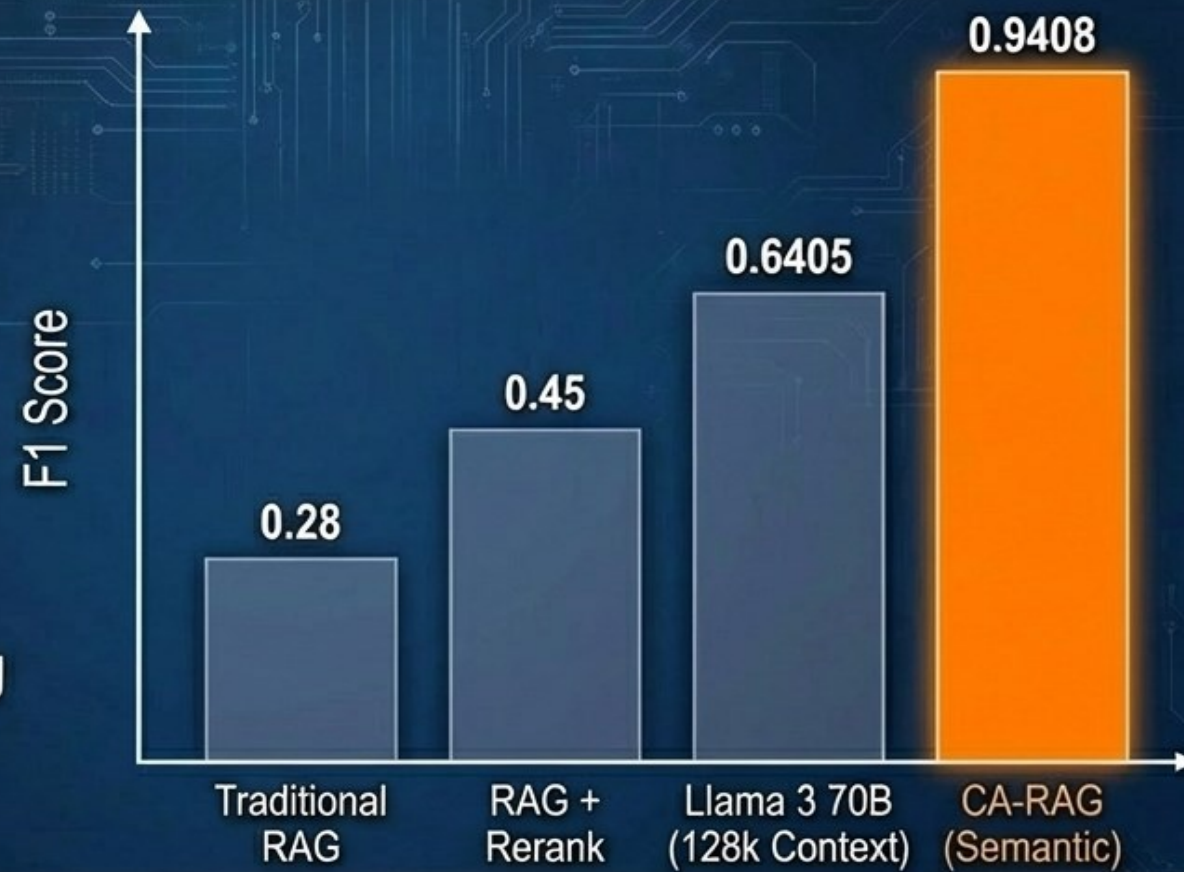
- TriviaQA: best F1 = 0.9089 with CA-RAG + dot product + Contriever under fixed-size chunking.
- AmbigQA: best F1 = 0.8652 with the same CA-RAG configuration.
- NaturalQA and SQuAD: RAG-Rerank remains stronger, suggesting CA-RAG is not universally dominant.
- Interpretation from the paper: CA-RAG helps more when context is longer or evidence is harder to surface with top-k retrieval.

Big idea
This is not “CA-RAG beats everything”;
it is “CA-RAG changes where
performance gains appear.”

Scenario 2: Domain-Specific Superiority

Data Insights:

- **The Retrieval Collapse:** On full scientific articles, traditional retrieval architectures utterly collapse, failing to pull the correct chunks into the top-k.
- **Beating Massive Context:** CA-RAG significantly out-performs a massive LLM (Llama 3 70B) reading the entire 10,000+ word document at once in a 128k context window.
- **The Insight:** Strict algorithmic similarity filtering on discrete, semantic chunks is far more reliable for factual grounding than a massive model's internal attention mechanisms.



Scenario 2 results: custom scientific-document QA

This is where the paper makes its strongest case

Method	Embedding Model	Semantic Chunking				Fixed-Size Chunking			
		Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
RAG	miniLM	0.4900	0.5500	0.5900	0.5700	0.4286	0.1481	0.8000	0.2500
RAG	Contriever-Msmarco	0.4762	0.4615	0.6	0.5218	0.4286	0.1481	0.8000	0.2500
RAG-Rerank	miniLM	0.6191	0.6667	0.72	0.6923	0.7619	0.6923	0.900	0.7826
RAG-Rerank	Contriever-Msmarco	0.5476	0.5926	0.6667	0.6275	0.7619	0.6923	0.900	0.7826
CA-RAG without post processing	miniLM	0.4600	0.3400	0.6900	0.4500	0.6092	0.4435	0.9373	0.6021
CA-RAG with post processing (Cosine Similarity)	miniLM	0.9200	0.8700	1.0000	0.9300	0.8539	0.7821	0.9985	0.8771
CA-RAG with post processing (Cosine Similarity)	Contriever-Msmarco	0.9144	0.8935	0.9934	0.9408	0.9297	0.8871	1.0000	0.9400
CA-RAG with post processing (Dot product)	miniLM	0.9704	0.9557	1.0000	0.9773	0.7126	0.5925	0.9617	0.7732
CA-RAG with post processing (Dot product)	Contriever-Msmarco	0.9832	0.9748	1.0000	0.9872	0.7782	0.6827	0.9780	0.8041
Long Context QA with Llama3.3	All Context (no chunking)								
		Acc.		Prec.		Rec.		F1	
		0.7619		0.6667		0.9474		0.7824	

Headline results

0.987

Best F1 (semantic chunking, dot product, Contriever)

0.940

Best F1 (fixed-size chunking, cosine similarity, Contriever)

Why it matters

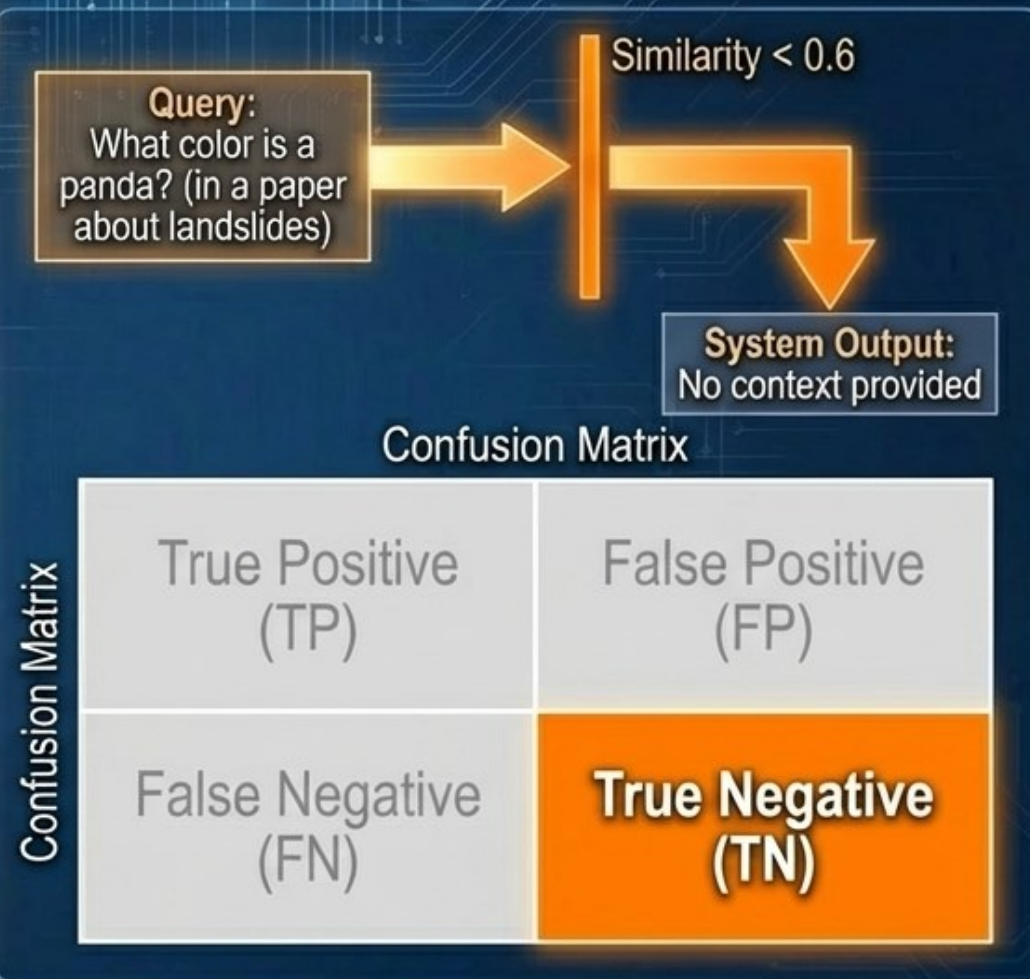
- CA-RAG post-processing sharply reduces false positives on out-of-context questions.
- The long-context Llama 3.3 baseline scores F1 = 0.7824, far below the best CA-RAG setup in this scenario.
- The paper argues that answer validation is especially valuable when whole scientific papers create long, messy context.

Scenario 3: The True Negative Advantage

The True Negative Advantage: Enforcing Abstention

Key Findings:

- **The Hallucination Trap:** Traditional systems attempt to answer irrelevant, out-of-context questions using internal model memory, creating confident falsehoods.
- **The CA-RAG Filter in Action:** If the mathematical similarity between the generated answer and the specific source chunk is below 0.6, the system strictly forces a 'No context provided' abstention.
- **Impact:** This results in a massive increase in True Negatives (TN) and the near-elimination of False Positives (FP). This conservative abstention is a non-negotiable requirement for medical, legal, and scientific applications.

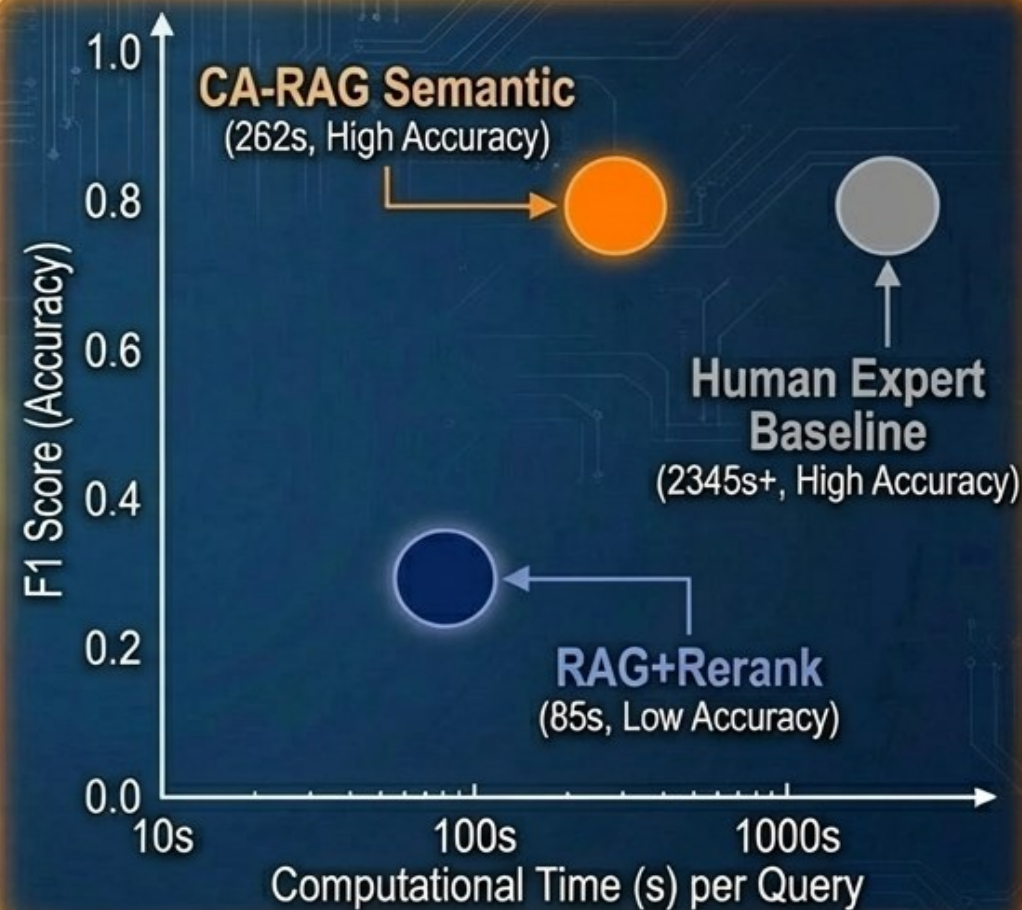


Synthesis: The Compute-Accuracy Trade-Off

Synthesis: The Compute-Accuracy Trade-Off

Analytical Breakdown:

- **The Cost of Grounding:** CA-RAG averages 262 seconds per query using Semantic Chunking, compared to RAG's 85 seconds.
- **Fixed vs. Semantic Cost:** Fixed-size chunking explodes processing time to 2289s due to a higher volume of fragmented chunks triggering excess LLM generation calls.
- **The Human Benchmark:** An expert requires roughly 39 to 65 minutes to read a 10,000-word paper and extract a grounded answer. CA-RAG completes this in ~4 minutes.
- **Verdict:** For zero-tolerance domains, trading ~3 minutes of compute time for near-perfect factual grounding is a highly advantageous architectural choice.



Conclusions & Future Research Directions

Zero-Hallucination Specialized AI

Architecture (Inversion)

Eliminates the
retrieval bottleneck.
Generating first and
filtering second is
fundamentally more
robust for complex
contexts.

Performance (Accuracy over Speed)

Highly consistent
across embedding
models (MiniLM vs
Contriever).
Achieves
state-of-the-art
hallucination
resistance.

Future Work (CoT Integration)

Integrating
"Chain-of-Thought"
prompting during
generation to
enhance multi-hop
abstract reasoning
prior to similarity
validation.

STABLE FOUNDATION

Speed is the main trade-off

CA-RAG buys faithfulness by spending more computation

262.6 s

Average time per question for CA-RAG with semantic chunking

85 s

Approx. average time per question for RAG with reranking

2289.5 s

Average time per question for CA-RAG with fixed-size chunking

Interpretation

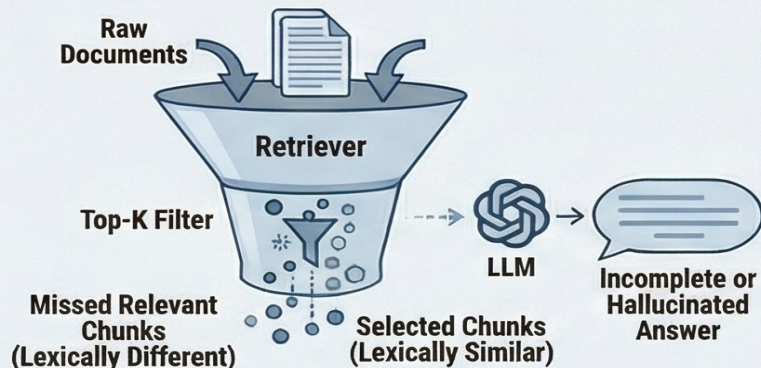
- Semantic chunking dramatically lowers CA-RAG cost relative to fixed-size chunking because it yields fewer chunks and thus fewer LLM calls.
- The paper argues this cost can be justified in domains where answer accuracy and completeness are more important than latency.
- The authors also compare against a rough human-reading upper bound and claim the best CA-RAG configuration is still faster than a human expert reading the entire document from scratch.

TABLE 5. Computational time of the proposed method and traditional RAG (related to Scenario 2).

Chunking Method	Method	Embedding model	Average time per question (s)
Semantic	RAG	miniLM	33.32
	RAG	Contriever-msmarco	33.46
	RAG-Rerank	miniLM	86.36
	RAG-Rerank	Contriever-msmarco	85.81
	CA-RAG	miniLM	262.63
	CA-RAG	Contriever-msmarco	262.65
Fixed-Size	RAG	miniLM	24.02
	RAG	Contriever-msmarco	24.31
	RAG-Rerank	miniLM	240.62
	RAG-Rerank	Contriever-msmarco	233.32
	CA-RAG	miniLM	2289.47
	CA-RAG	Contriever-msmarco	2289.49

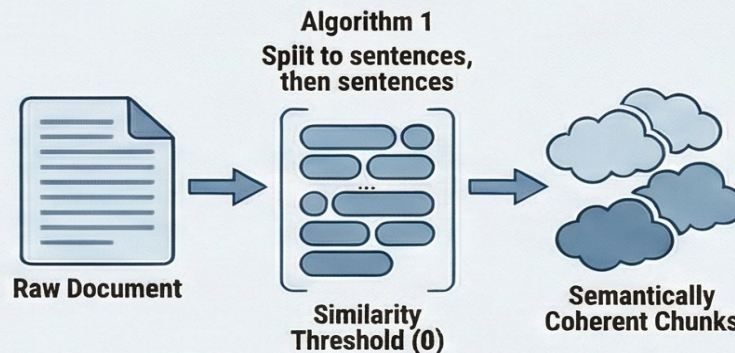
CA-RAG: Eliminating the Retrieval Bottleneck for Higher LLM Accuracy

1. The Problem: The Traditional Retrieval Bottleneck



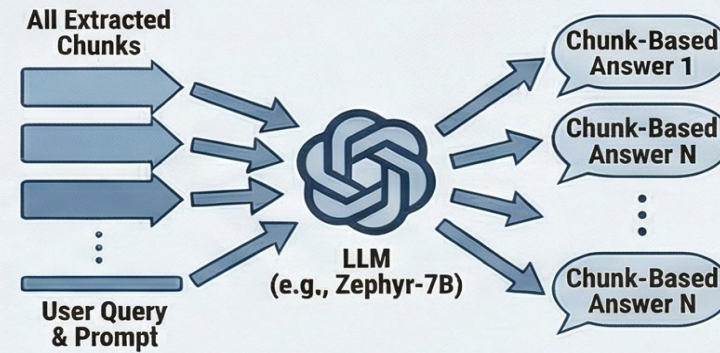
Traditional RAG systems often miss key information because the initial "Top-K" retrieval phase filters out relevant chunks that may be lexically different but semantically vital.

2. Step 1: Semantic Indexing & Chunking



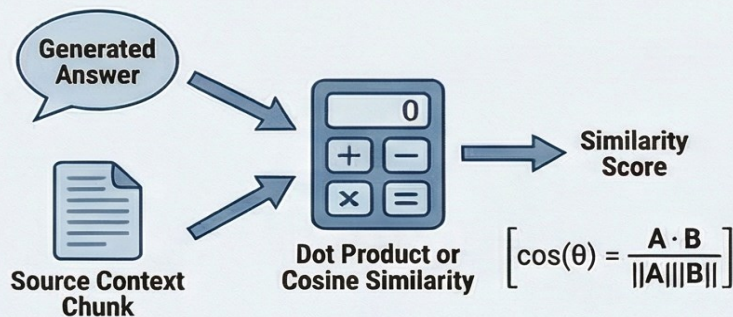
Unlike fixed-size splitting, CA-RAG uses Algorithm 1 to split text into sentences and cluster them into "Semantically Coherent Chunks" based on a similarity threshold (θ).

3. Step 2: Unfiltered Generation Phase



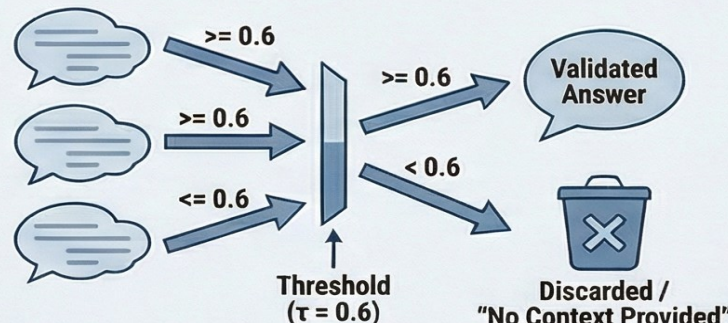
Every extracted chunk is combined with the user query and a prompt, then passed directly to the LLM to generate a series of "chunk-based answers."

4. Step 3: Similarity Validation (Post-Processing)



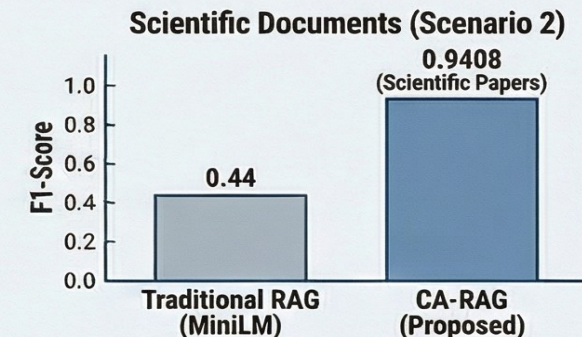
The system calculates a similarity score (Dot Product or Cosine Similarity) between the generated answer and its source context chunk to verify relevance.

5. Step 4: Filtering & Best Response Selection



Answers falling below the similarity threshold ($\tau = 0.6$) are discarded or marked "No Context Provided" to prevent hallucinations and handle out-of-context questions.

6. The Result: Superior Precision in Specialized Domains



In Scenario 2 (Scientific Documents), CA-RAG achieved an F1-score of 0.94, significantly outperforming traditional RAG and even long-context models like Llama 3.3-70B.

Strengths, limitations, and validity threats

How to read the paper critically

What is genuinely strong here?

- The paper frames abstention as a desirable behavior, not a failure mode.
- It evaluates both standard benchmarks and a more realistic custom scenario.
- It reports when CA-RAG loses, which makes the contribution more believable.
- It explicitly discusses accuracy/latency trade-offs instead of hiding them.

What should we question?


- Similarity between answer and context is only a proxy for groundedness; multihop or abstract reasoning may break it.
- Processing every chunk scales poorly as corpora get larger.
- The custom dataset is useful but still relatively small and handcrafted.
- Results depend on threshold choices and chunking design, which may require tuning in new domains.



CA-RAG problems


- No multimodality
- Multihope

Logged in as:
edoardo_tester

 Your Profile

Logout

SnapAssistant model:

Llama3.3 

New Chat

EdoChat1

EdoChat2

EdoChat3

EdoChat4

prova

chatNuova

chatTest1LLAMA

chatTest2LLAMA

testGianniLLAMA

testTastoInvioPreChanges

ChatTestPostPSWINVIO


<https://www.snap4city.org/1116>


SnapAdvisor

Current chat: debug0

Visualize references

Personal
Advisor LLM

 Come creo una dashboard?

 Per creare una dashboard, puoi utilizzare la piattaforma Snap4City. La dashboard si compone di un insieme di widget grafici che possono essere impostati separatamente assegnando una serie di parametri, come origine dati, dimensioni, colori, forma, font, allarmi, relazioni con altri, ecc. (SOURCE: Snap4 City per la creazione di Smart City Control Room, SCCR). Inoltre, la piattaforma offre la possibilità di creare vere e proprie applicazioni interattive, multipagina/multidashboard, e non semplici storie monopagina da scrollare. Puoi anche condividere la dashboard con altri utenti anche fuori dalla Smart City Control Room tramite canali autenticati e protetti in HTTPS, in accordo al GDPR (SOURCE: Snap4 City per la creazione di Smart City Control Room, SCCR). Per vedere alcuni esempi di dashboard, puoi visitare il sito <https://www.snap4city.org/dashboardSmartCity/view/index.php?iddashboard=MjU0Mg==> (SOURCE: Snap4 City 1st Overview v4 8 annot). Tuttavia, non ho trovato informazioni dettagliate sulla procedura di creazione di una dashboard. Se hai bisogno di ulteriore assistenza, ti consiglio di contattare direttamente il team di supporto di Snap4City. Non so dirti altro in merito.

Write a message...





Powered by FIWARE

FREE TRIAL

PEN Test Passed

EU GDPR COMPLIANT

SNAP4 Appliances and Dockers Installations

EUROPEAN OPEN SCIENCE CLOUD

Node-RED

JS Foundation

E015 digital ecosystem

NVIDIA

OPERATION AND PLAN - CONTROL ROOMS - DECISION SUPPORT SYSTEMS - WHAT-IF ANALYSIS - OPTIMIZATION - APPLICATIONS

HORIZONTAL AI PLATFORM

MOBILITY AND TRANSPORT

SMART ENERGY AND SMART BUILDING

ENVIRONMENT AND WASTE MANAGEMENT

CITY USER'S SERVICES AND TOURISM MANAGEMENT

SNAPADVISOR

BUSINESS INTELLIGENCE - SIMULATIONS - VISUAL ANALYTICS - SYNOPTICS - GRAPHICAL WIDGETS - ANALYTICS

DASHBOARDS, WIDGETS TEMPLATES

**PREDICTION - ANOMALY DETECTION - CLUSTERING - ROUTING - SENTIMENT NLP - TRAFFIC FLOW - PEOPLE FLOWS - SDG
15 MIN CITY INDEX - KPI - HEATMAPS - ORIGIN DESTINATION - MAPS - VECTOR FIELD - ETC...**

**API - MICROSERVICES - GIS - BPM
VIDEO - REPORTS - MAPS - 3D ...**

**EXPERT SYSTEM, KNOWLEDGE BASE SEMANTIC REASONING
SMART DATA MODEL
IOT DEVICE MODELS, DATA SPACES**

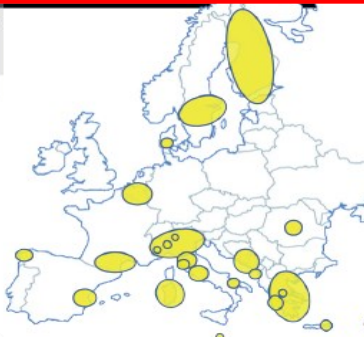
**BIG DATA ANALYTICS, ARTIFICIAL INTELLIGENCE
EXPLAINABLE AI, MACHINE LEARNING, GENERATIVE AI
OPERATIVE RESEARCH, STATISTICS**

**VISUAL PROGRAMMING, ADAPTERS
DATA FLOWS, WORKFLOWS
PARALLEL DISTRIBUTED PROCESSING
DATA DRIVEN**

FULL INTEROPERABILITY, ANY: DATA, BROKERS, NETWORKS AND VERTICALS



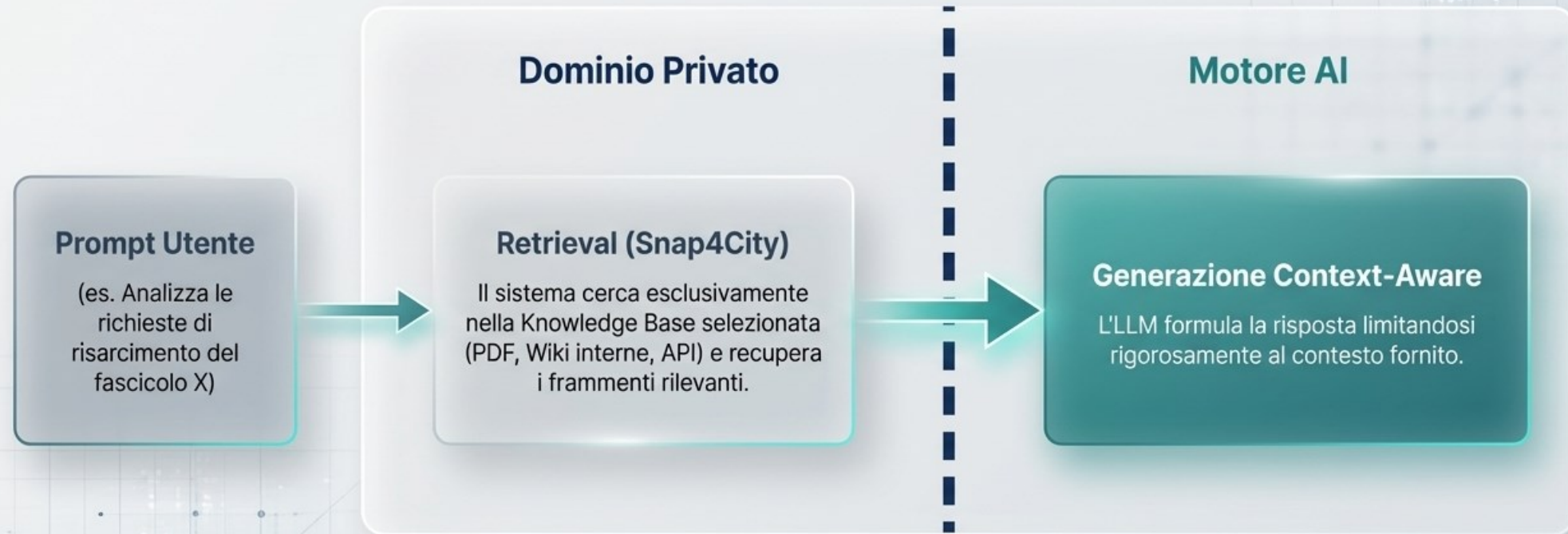
- NATIVE AND EXTERNAL APPLICATIONS**
- Smart Parking
 - Smart Light
 - Smart Waste
 - Smart Energy
 - Smart Building
 - Smart Tourism
 - ...



**DEVELOPMENT ENVIRONMENT AND METHODOLOGY
VISUAL PROGRAMMING, ML, AI, HPC
TRAINING COURSES**

Architettura RAG di SnapAdvisor: Flusso di Conoscenza Controllato

Separazione sicura tra dati privati e motore AI per risposte precise e verificabili.



Controllo del Dominio:
L'utente decide quali documenti interrogare.

Spiegabilità:
Ogni risposta include riferimenti diretti (SOURCE: Doc X, Pag Y).

Nessuna Interferenza:
Modulare e multi-utente; i dati restano isolati.

SnapAdvisor: AI based Assistants, DSS

- **Snap4CityAdvisor:**
 - for developers and promoters knowing all Snap4City and DISIT Lab literature, articles, manuals, slides, codes, etc. The SnapAdvisor with this skill is accessible only for selected Snap4City users. It allows users of saving a lot of time by providing answers to facilitate the development of smart solutions, to the exploitation of Snap4City tools, and recently on data and services accessible on the platform, producing offers, etc.
- **Legal Advisor**
 - expert on specific disputes at service of the Legal department of Careggi Hospital of Florence. It allows to save time in recovering precise information from complex legal documents, ordering of events, understanding causes and effects, producing reports, etc.
- **Expert of industrial machines, user manuals, technical manuals, rules,**
 - Answering on technical manuals
- **Complains and Questionnaire analysis**
 - Answering on trends, via questionnaires collected via QR, blobs, emails, etc.
- **Commercial Advisor:**
 - processing orders understanding them and preparing the offer, thus reducing the time to process them.
- **Generative Designs:**
 - Multimodal Generative AI supporting designers in producing innovations
- etc.

See: "Context-Aware Retrieval Augmented Generation using Similarity Validation to handle Context Inconsistencies in Large Language Models", IEEE Access, 2025. <https://doi.org/10.1109/ACCESS.2025.3614553>

AI based Assistants, DSS

- **Tourism domain**
 - PAVAL assistant for providing Touristic information
- **Legal domain**
 - Assessment of mediation proneness, prediction of mediation results, NLP, BERT (Florence Forum)
 - Assessment/prediction of risk on legal disputes wrt hospitals
- **Training and development domains**
 - **SnapAdvisor**: LLM advanced RAG, covering both applications and much more including the Advisor / HelpDesk for Snap4City platform
 - Multi language on private documents, etc.

SnapAdvisor

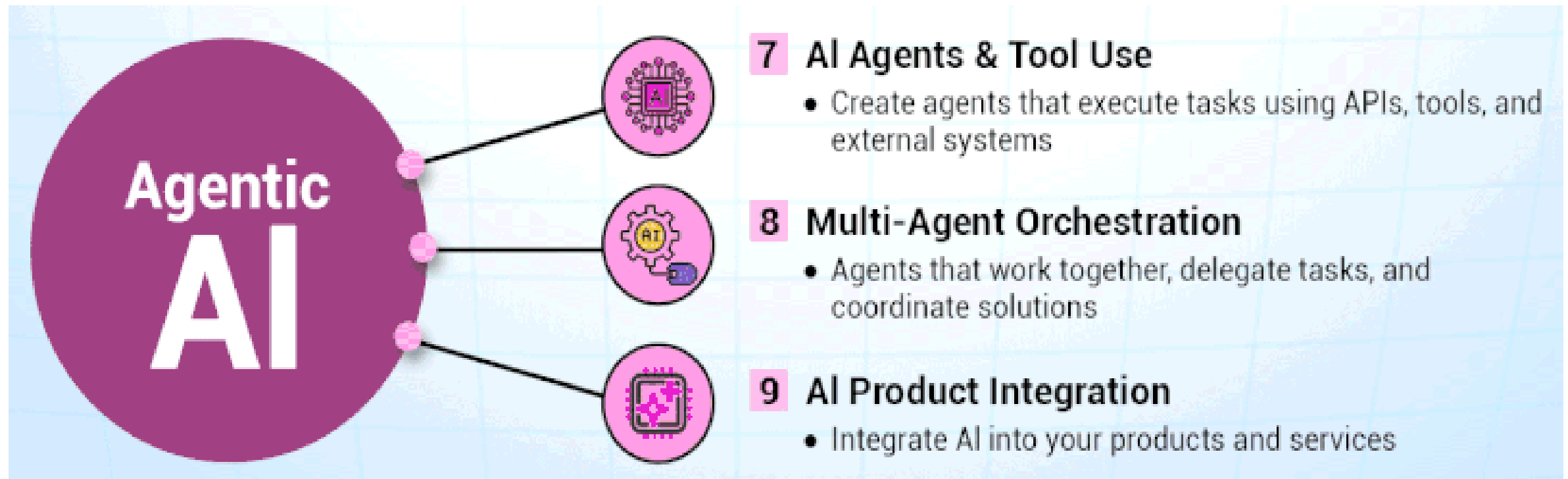
- **working on your private content** and thus producing answers.
 - Select a subset of documents
- **domain control**: you can decide the knowledge base (internal wikis, PDFs, APIs),
- **explainability** capability, provide references to your documents
- **multilingual via content**
- **Multimodality: images, slides, text**
- **modularity**: it is possible to pass from one collection of documents to another, and multiple users can work on the advisor asking for different topic on different collections/domains at the same time, independently as needs change, without any interferences among them.
- **lower hallucinations** <https://www.snap4city.org/1116>

SnapAdvisor

- **Each User can**

- have a number of Collections. Each collection may include an arbitrary number of documents: PDF, Web pages, TXT, Slides, images
- perform a number of Chats on each Collection.
 - The Chat is a discussion multisection/questions for a purpose.
 - The SnapAdvisor takes into account the evolution of the chat (multi-shot).
 - At each Question in the Chat the user may decide to select a subset of documents of the selected Collection.
- delegate the use of his/her Collections to other users.

The evolution is now: Agent/Agentic AI





UNIVERSITÀ
DEGLI STUDI
FIRENZE

DINFO
DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

DISIT
DISTRIBUTED SYSTEMS AND
INTERNET TECHNOLOGIES LAB
DISTRIBUTED DATA INTELLIGENCE
AND TECHNOLOGIES LAB





UNIVERSITÀ
DEGLI STUDI
FIRENZE

DINFO
DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

DISIT
DISTRIBUTED SYSTEMS AND
INTERNET TECHNOLOGIES LAB
DISTRIBUTED DATA INTELLIGENCE
AND TECHNOLOGIES LAB





UNIVERSITÀ
DEGLI STUDI
FIRENZE

DINFO
DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

DISIT
DISTRIBUTED SYSTEMS AND
INTERNET TECHNOLOGIES LAB
DISTRIBUTED DATA INTELLIGENCE
AND TECHNOLOGIES LAB





UNIVERSITÀ
DEGLI STUDI
FIRENZE

DINFO
DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

DISIT
DISTRIBUTED SYSTEMS AND
INTERNET TECHNOLOGIES LAB
DISTRIBUTED DATA INTELLIGENCE
AND TECHNOLOGIES LAB





UNIVERSITÀ
DEGLI STUDI
FIRENZE

DINFO
DIPARTIMENTO DI
INGEGNERIA
DELL'INFORMAZIONE

DISIT
DISTRIBUTED SYSTEMS AND
INTERNET TECHNOLOGIES LAB
DISTRIBUTED DATA INTELLIGENCE
AND TECHNOLOGIES LAB

