

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Informatica indirizzo AI & Data Analytics

A.a. 2025/2026

Sessione di laurea Ottobre 2026

Progettazione e Validazione di un Framework di Valutazione Pedagogica per Sistemi Retrieval-Augmented Generation in Ambienti di E-Learning

Relatore:

Alessio Sacco

Candidato:

Carmine Francesco Benvenuto

Indice

Elenco delle figure VI

1 Introduzione 1
- 1.1 Contesto e motivazione . . . 1
  - 1.1.1 Il gap valutativo . . . 2
  - 1.1.2 La specificità della valutazione pedagogica . . . 3
- 1.2 Definizione del problema . . . 3
- 1.3 Contributi della tesi . . . 4
  - 1.3.1 Contributo 1 — Il framework `rag_eval` . . . 4
  - 1.3.2 Contributo 2 — Il Pedagogical Evaluation Score (PES) . . . 5
  - 1.3.3 Contributo 3 — Validazione empirica multi-benchmark . . . 6
- 1.4 Domande di ricerca . . . 7
- 1.5 Inquadramento metodologico e contesto applicativo . . . 8
- 1.6 Struttura della tesi . . . 9

2 Related Works e Stato dell'Arte 10
- 2.1 Retrieval-Augmented Generation (RAG) . . . 10
  - 2.1.1 Fondamenti e motivazione . . . 10
  - 2.1.2 Architettura di riferimento . . . 11
  - 2.1.3 Tassonomia dei sistemi RAG . . . 12
  - 2.1.4 RAG in ambito educativo e tutoring conversazionale . . . 13
- 2.2 Valutazione dei sistemi RAG . . . 15
  - 2.2.1 Metriche lessicali tradizionali . . . 15
  - 2.2.2 Metriche basate su rappresentazioni dense . . . 17
  - 2.2.3 Il paradigma LLM-as-a-Judge . . . 19
  - 2.2.4 Framework di valutazione RAG esistenti . . . 20
- 2.3 Valutazione pedagogica dei sistemi conversazionali . . . 22
  - 2.3.1 Teorie dell'apprendimento di riferimento . . . 22
  - 2.3.2 Conversational Uptake . . . 23
  - 2.3.3 Readability e adattamento linguistico . . . 23
  - 2.3.4 No Immediate Disclosure (NID) . . . 24
  - 2.3.5 Benchmark e dataset di riferimento . . . 25
- 2.4 Tecnologie e strumenti utilizzati . . . 26
  - 2.4.1 Large Language Models . . . 26
  - 2.4.2 Modelli di embedding e Natural Language Inference . . . 27

2.4.3 LangChain . . . 28
2.4.4 Inferenza locale: Ollama e vLLM . . . 29

3 Architettura del Framework `rag_eval` 31
3.1 Principi architetturali e design pattern . . . 31
3.1.1 Pattern architetturali adottati . . . 31
3.2 Modelli di dominio e contratti dati . . . 32
3.2.1 `TestSample` — L'unità atomica di valutazione . . . 32
3.2.2 `EvalResult` — L'esito atomico della metrica . . . 33
3.2.3 `EvalReport` — Il container aggregato di sessione . . . 33
3.3 Astrazione multi-provider per modelli linguistici . . . 33
3.3.1 Provider supportati nativamente . . . 34
3.3.2 Caching deterministico basato su SHA-256 . . . 34
3.3.3 Telemetria dei consumi ed economica . . . 34
3.4 Client di embedding densi . . . 35
3.5 Pipeline di valutazione e orchestrazione asincrona . . . 35
3.6 La Suite di Metriche RAG Implementate e Profilazione Computazionale 36
3.6.1 Tassonomia e implementazione delle metriche . . . 37
3.6.2 Profilazione computazionale, latenze e costi . . . 38
3.6.3 Criteri di selezione e albero di instradamento operativo . . . 39
3.7 Generazione sintetica di dataset . . . 40
3.7.1 Partizionamento semantico dinamico . . . 40
3.7.2 Generazione multi-tipo e iniezione di contesti distrattori . . . 40
3.8 Adattatori per benchmark standard . . . 41
3.9 Integrazione LangChain . . . 44
3.10 Sistema di reporting multi-formato . . . 44
3.11 Command Line Interface (CLI) . . . 45
3.11.1 Iniezione intelligente delle dipendenze . . . 45

4 Il Pedagogical Evaluation Score (PES) 47
4.1 Motivazione teorica e fondamenti didattici . . . 47
4.1.1 I limiti delle metriche RAG tradizionali nei contesti educativi 47
4.1.2 Principi pedagogici per la valutazione . . . 48
4.1.3 Metriche RAG statiche vs. Metriche Didattiche Dinamico-Adattive . . . 49
4.2 Architettura gerarchica a tre livelli del PES . . . 49
4.2.1 Pipeline computazionale e flusso informativo . . . 49
4.2.2 Algoritmo di valutazione asincrona parallela . . . 52
4.3 Livello 0 — Faithfulness Safety Gate . . . 53
4.3.1 Motivazione del vincolo non compensativo sulla fedeltà . . . 53
4.3.2 Decomposizione in asserzioni atomiche e verifica logica . . . 53
4.3.3 Soglia di arresto e configurazione del gate . . . 53
4.4 Livello 1 — Classificazione dello Stato Pedagogico e Ponderazione Dinamico-Adattiva . . . 54
4.4.1 Tassonomia degli stati pedagogici e il paradigma dinamico-adattivo . . . 54

4.4.2 Motivazione dell'azzeramento di w_1 negli Stati B e C 56
4.5 Livello 2 — Tassonomia delle sotto-metriche 57
4.5.1 M_1 — Uptake 57
4.5.2 M_2 — Linguistic Adaptation 58
4.5.3 M_3 — Scaffolding, NID & Helpfulness 60
4.6 Formulazione matematica unificata e proprietà analitiche 62
4.6.1 Equazione maestra con lisciamento ϵ-smoothing 62
4.6.2 Proprietà matematiche: non-compensatorietà e confronto tra medie 62
4.6.3 Sintesi del paradigma dinamico-adattivo e raccordo sperimentale 63
4.7 Esempi applicativi e analisi qualitativa 63
4.8 Considerazioni architetturali e trade-off ingegneristici 64

5 Validazione Sperimentale e Discussione 66
5.1 Setup sperimentale 66
5.2 Risultati su MRBench v3 67
5.2.1 Distribuzione del PES per classe qualitativa umana 67
5.2.2 Test di significatività e correlazione con i giudizi umani 69
5.2.3 Graduatoria comparativa dei modelli e baseline umane 71
5.3 Risultati su MathDial 72
5.3.1 Capacità discriminativa: Scaffolding vs. Telling 72
5.3.2 Comportamento del Faithfulness Gate nei dialoghi interattivi . 74
5.4 Studio di ablazione e ottimizzazione dei pesi 74
5.5 Analisi discriminativa e gerarchia delle componenti 77
5.5.1 Analisi per decili e natura ordinale del PES 77
5.5.2 Scomposizione dell'effetto del Gate e coerenza tra modelli . . . 79
5.6 Caso di studio aziendale: Piattaforma E-Learning (Clait S.r.l.) 80
5.6.1 Valutazione RAG Triad e Refusal Accuracy (50 Query Operative e Stress Test) 80
5.6.2 Valutazione Didattica e Tutoriale PES (30 Scenari Formativi) 81
5.6.3 Profilazione operativa, telemetria e monitoraggio in produzione 82
5.7 Discussione critica e lezioni apprese 83
5.7.1 Modelli aperti vs. proprietari e sovranità del dato didattico . . 84
5.7.2 Il divario tra RAG documentale e tutoring didattico: lezioni dal caso aziendale 84
5.7.3 Il Faithfulness Gate tra integrità scientifica e paradosso maieutico 85
5.7.4 La decostruzione del paradosso dell'esperto umano 86
5.8 Linee guida per la progettazione di tutor RAG didattici 87
5.9 Minacce alla validità e limitazioni 88
5.9.1 Validità di costrutto 88
5.9.2 Validità interna 88
5.9.3 Validità esterna 88
5.9.4 Validità delle conclusioni e affidabilità statistica 89
5.10 Risposte alle domande di ricerca 89
5.11 Sintesi del capitolo 90

6 Lavori Futuri 91
6.1 Evoluzione metodologica ed epistemica della metrica PES 92
6.1.1 Adaptive ed Epistemic Faithfulness Gate: superamento del paradosso maieutico 92
6.1.2 Modelli di Conversational Uptake End-to-End e filtraggio delle misconcezioni 93
6.1.3 Ponderazione continua e differenziabile ancorata all'Ablation Study 94
6.1.4 Sotto-metriche M_4 (Supporto Emotivo) e M_5 (Carico Cognitivo e ZPD) 94
6.2 Allineamento didattico: dalla valutazione alla generazione controllata 95
6.2.1 Socratic Constrained Decoding e Test-Time Steering 95
6.2.2 Curatela dei dataset e superamento della *Curse of Knowledge* 96
6.2.3 Reinforcement Learning from AI Feedback (RLAIF) e GRPO 96
6.2.4 Direct Preference Optimization (DPO) e Self-Play Multi-Agente 97
6.3 Dal singolo turno al RAG agentico e longitudinale 97
6.3.1 Valutazione longitudinale di traiettoria (Session-Level Evaluation) 97
6.3.2 Supporto RAG multimodale nativo 98
6.3.3 Valutazione del Pedagogical Tool-Use nei RAG agentici multi-step 98
6.4 Potenziamento architetturale e monitoraggio continuo in produzione . 99
6.4.1 Distillazione di giudici compatti (Small Language Models su Edge) 99
6.4.2 Continuous Monitoring e rilevamento del degrado didattico (Pedagogical Drift) 99
6.4.3 Standardizzazione open-source ed ecosistema di benchmark . . 100
6.5 Validazione clinica, cross-dominio e impatto didattico 100
6.5.1 Studi longitudinali di Learning Gain su studenti reali 100
6.5.2 Validazione con docenti di ruolo e accordo inter-valutatore . . 101
6.5.3 Generalizzazione a domini umanistico-giuridici e metriche di leggibilità native 101
6.6 Roadmap evolutiva e sintesi delle priorità 101

7 Conclusione 102
7.1 Sintesi del lavoro svolto e contributi architetturali 102
7.2 Riscontro analitico alle domande di ricerca 104
7.3 Risultati chiave e impatto scientifico ed industriale 105
7.3.1 La decostruzione del paradosso dell'esperto umano e la *Curse of Knowledge* 106
7.3.2 Competitività dei modelli Open-Weight e tutela della sovranità didattica 107
7.3.3 Validazione operativa sul campo e sostenibilità industriale (Clait S.r.l.) 107
7.4 Linee guida architetturali per la progettazione di Tutor RAG 108
7.5 Raccordo con l'agenda dei lavori futuri 109
7.6 Riflessione finale ed etica: l'IA come levatrice del pensiero 110

Bibliografia 111

Elenco delle figure

2.1 Confronto architetturale tra i paradigmi *Naive RAG* e *Advanced RAG* [22]. Rispetto al flusso lineare elementare della variante Naive, l'Advanced RAG introduce moduli specializzati di *pre-retrieval* (elaborazione della query e indicizzazione avanzata) e *post-retrieval* (re-ranking e filtraggio contestuale) a monte e a valle del recupero, ottimizzando la pertinenza del contesto prima della generazione con LLM. 13
2.2 La Triade RAG estesa e le relazioni diagnostiche tra query, contesti recuperati, risposta generata e ground truth. 20

3.1 Tassonomia gerarchica delle metriche RAG integrate nel framework `rag_eval`. 37
3.2 Albero di instradamento per la selezione del profilo di valutazione ottimale. 39

4.1 Diagramma architetturale completo della metrica composita Pedagogical Evaluation Score (PES), con evidenziazione dei tre livelli operativi, del gate di sicurezza documentale e dell'aggregazione non lineare. . . 51

5.1 Struttura complessiva della campagna di validazione sperimentale del framework `rag_eval`. 67
5.2 Distribuzione statistica e profilo di densità del PES rispetto alle annotazioni umane (*Providing Guidance*) su MRBench v3 ($N = 2.475$). 68
5.3 Matrice delle correlazioni tra sotto-metriche PES e dimensioni di valutazione umana su MRBench v3. 70
5.4 Classifica comparativa dei modelli linguistici e dei tutor umani per punteggio PES medio su MRBench v3. 71
5.5 Distribuzione comparativa dei punteggi PES rispetto alle quattro strategie didattiche (*Teacher Moves*) su MathDial ($N = 2.040$). . . . 73
5.6 Mappa termica della correlazione di Spearman ρ al variare continuo dei pesi (w_1, w_2, w_3) nello studio di ablazione su MRBench v3. 76
5.7 Architettura gerarchica e ruoli diagnostici delle componenti del PES: dal filtro di ammissibilità documentale (Livello 0) all'integrazione asimmetrica delle dimensioni pedagogiche (Livello 2). 78
5.8 Composizione percentuale delle etichette umane nei decili di punteggio PES su MRBench v3. 79

5.9 Dashboard Grafana per il monitoraggio in produzione della pipeline RAG: ripartizione dei token, finestre di costo e latenze disaggregate per microservizio. 83

6.1 Direttrici di espansione metodologica ed epistemica per la metrica PES. 92
6.2 Integrazione del PES per il controllo a tempo di inferenza e per l'allineamento tramite RLAIF e DPO. 95
6.3 Direttrici di estensione architetturale per la valutazione di traiettorie e sistemi agentici. 98

7.1 I tre contributi metodologici ed ingegneristici cardine sviluppati nella tesi magistrale. 103
7.2 Quadro sinottico dei risultati chiave, delle evidenze scientifiche e dell'impatto applicativo. 106

Capitolo 1

Introduzione

1.1 Contesto e motivazione

L'avvento dell'intelligenza artificiale generativa sta profondamente ridefinendo i paradigmi dell'istruzione, della didattica e della formazione avanzata. Nel corso di pochi anni, i modelli linguistici di grandi dimensioni (*Large Language Models*, LLM) sono passati dall'essere oggetti di sperimentazione confinati nei laboratori di ricerca a strumenti pervasivi e trasversali. Essi si dimostrano capaci di sostenere dialoghi articolati, sintetizzare e spiegare costrutti concettuali complessi e rimodulare il proprio registro linguistico e stilistico in base al profilo dell'interlocutore.

In questo scenario, l'impiego di tali architetture nel ruolo di *tutor conversazionali* intelligenti, ovvero agenti capaci di guidare lo studente nell'apprendimento attraverso il dialogo, rappresenta una delle direzioni applicative di maggior impatto scientifico e sociale, ma allo stesso tempo tra le più delicate e rischiose dell'intero ecosistema dell'IA.

L'affermazione del paradigma della *Retrieval-Augmented Generation* (RAG), formalizzato originariamente da Lewis et al. [1], ha impresso un'ulteriore accelerazione a questo settore. L'architettura RAG accoppia un modulo di recupero delle informazioni (*retriever*) — incaricato di estrarre passaggi rilevanti o evidenze fattuali da una base di conoscenza esterna e verificabile — a un modulo generativo (*generator*), costituito da un LLM che sintetizza la risposta vincolandola congiuntamente al prompt dell'utente e ai contesti documentali recuperati.

Tale impostazione mitiga in modo sostanziale una delle vulnerabilità strutturali più critiche dei modelli puramente generativi: la tendenza alla generazione di allucinazioni (*hallucinations*), ovvero la produzione di asserzioni sintatticamente fluide e persuasive ma prive di fondamento [2, 3]. Imponendo che la generazione si basi deterministicamente su fonti autorevoli, i sistemi basati su RAG garantiscono un grado superiore di veridicità, verificabilità e tracciabilità dell'informazione. Si tratta di requisiti imprescindibili e non negoziabili nei contesti formativi, dove la propagazione di nozioni errate o fuorvianti comprometterebbe irrimediabilmente il processo educativo.

Di conseguenza, l'integrazione di sistemi RAG in ambienti di tutoraggio conversazionale ha registrato un incremento esponenziale. Grandi piattaforme dell'EdTech

— quali ad esempio Khan Academy tramite l'assistente *Khanmigo*, Duolingo Max, unitamente a molteplici iniziative promosse da istituzioni universitarie ed enti sovranazionali — stanno progressivamente incorporando agenti conversazionali per la didattica, spaziando dall'istruzione primaria alla formazione professionale e specialistica. Proiezioni di settore stimano che il mercato globale dell'IA applicata all'istruzione supererà i 30 miliardi di dollari entro il 2032 [4]. Tuttavia, a questa rapida espansione applicativa non è corrisposta una parallela maturazione delle metodologie e dei framework dedicati alla valutazione della qualità dell'interazione.

1.1.1 Il gap valutativo

L'introduzione di un agente di tutoraggio basato su intelligenza artificiale pone un interrogativo metodologico fondamentale: *in che modo è possibile quantificare e validare la qualità pedagogica di un agente conversazionale?*

Finora, la ricerca in *Natural Language Processing* (NLP) e *Information Retrieval* (IR) si è concentrata soprattutto su metriche tradizionali, strumenti utili e necessari ma da soli non sufficienti per valutare un contesto didattico.

Metriche quali **BLEU** [5], **ROUGE** [6], **Token-F1** ed **Exact Match** quantificano il grado di sovrapposizione lessicale tra la stringa generata e una risposta di riferimento (*ground truth*). Concepiti originariamente per la traduzione automatica e la sintesi di testi, tali indicatori mostrano forti limiti nel dialogo pedagogico: soffrono di insensibilità semantica (penalizzando enunciati formalmente dissimili ma equivalenti), risultano inadeguati alla natura aperta del dialogo tutoriale (in cui un tutor efficace può articolare spiegazioni valide in molteplici forme didattiche) e mancano di una dimensione pragmatica, non riuscendo a discernere se la risposta guidi lo studente nello sviluppo del ragionamento o si limiti a un trasferimento nozionistico passivo.

D'altra parte, le **metriche basate su rappresentazioni dense (embedding)**, come la *Semantic Similarity* e **BERTScore** [7], misurano la vicinanza semantica tra i testi nello spazio vettoriale. Pur superando i limiti della semplice corrispondenza lessicale, esse non distinguono tra una spiegazione didattica strutturata e una risposta corretta ma puramente nozionistica, risultando incapaci di valutare un'impostazione socratica del dialogo.

Infine, il **paradigma LLM-as-a-Judge e i framework RAG standard** — resi possibili dall'impiego di modelli avanzati in veste di valutatori automatici [8] e formalizzati da strumenti quali RAGAS [9], ARES [10], TruLens e DeepEval — consentono di quantificare dimensioni quali la fedeltà ai contesti (*faithfulness*), la pertinenza della risposta (*answer relevance*) e la precisione/richiamo del contesto (*context precision* e *context recall*).

Pur costituendo un progresso fondamentale nell'analisi qualitativa dei sistemi RAG generali, le metriche correnti rispondono esclusivamente al quesito: *«La risposta è corretta, rilevante e ben fondata sui documenti?»*. Esse tralasciano del tutto la questione pedagogica: *«La risposta promuove e sostiene efficacemente l'apprendimento e il ragionamento critico del discente?»*.

1.1.2 La specificità della valutazione pedagogica

La qualità didattica di un intervento formativo è un concetto per sua natura multidimensionale. I principali modelli teorici dell'apprendimento — dalla *Zona di Sviluppo Prossimale* (ZPD) di Vygotsky [11] alla teoria dello *scaffolding* di Wood et al. [12], fino all'interrogazione socratica e al paradigma del *productive failure* [13] — concordano sul fatto che un tutor efficace non debba limitarsi a fornire la soluzione, ma debba sostenere, stimolare e guidare il processo di ragionamento dello studente.

Dal punto di vista operativo e computazionale, un agente conversazionale didatticamente efficace deve soddisfare i seguenti requisiti:

1. **Preservazione della fedeltà fattuale:** garantire che ogni affermazione sia corretta e verificabile; una risposta che contiene allucinazioni o inesattezze compromette sul nascere qualsiasi valore educativo;
2. **Ripresa conversazionale (*Conversational Uptake*):** recepire e valorizzare i contributi dello studente nei turni precedenti, ricollegandosi al suo filo logico ed elaborando i concetti introdotti [14];
3. **Sensibilità al contesto pedagogico:** adattare la strategia dialogica all'obiettivo del momento, distinguendo tra la spiegazione di un nuovo concetto, la correzione di un errore o la verifica delle competenze;
4. **Mancata rivelazione immediata della soluzione (*No Immediate Disclosure* – NID):** evitare di svelare subito la risposta finale, guidando lo studente a ricavarla autonomamente attraverso suggerimenti progressivi (*hints*) e domande mirate;
5. **Adattamento linguistico e stilistico:** calibrare il registro e la complessità del lessico in base alle esigenze dello studente, impiegando un tono incoraggiante e formulando feedback costruttivi.

Attualmente non esistono framework o metriche capaci di combinare organicamente queste dimensioni in una valutazione automatica e unificata. Questa carenza metodologica e tecnologica costituisce la motivazione principale del presente lavoro di tesi.

1.2 Definizione del problema

Il problema affrontato in questa tesi può essere riassunto nel seguente interrogativo di ricerca:

> *Come è possibile valutare in modo automatico e rigoroso la qualità pedagogica di un tutor conversazionale basato su RAG, tenendo conto contemporaneamente della correttezza fattuale rispetto alle fonti, dell'efficacia didattica dell'intervento e della capacità di adattarsi al contesto della conversazione?*

Per rispondere a questo quesito, il lavoro è stato articolato in tre sotto-problemi principali e interconnessi:

- **Sotto-problema 1 — Progettazione dell'infrastruttura di valutazione:** La valutazione completa di un sistema RAG richiede una piattaforma software in grado di orchestrare metriche eterogenee (lessicali, basate su embedding e guidate da LLM), supportare molteplici provider di modelli linguistici (sia commerciali su cloud sia open-weight in locale) e gestire dataset e reportistica in modo asincrono e affidabile. Poiché i framework esistenti non consentono di integrare facilmente dimensioni pedagogiche dedicate, si rende necessario progettare una nuova architettura modulare, aperta ed estendibile.
- **Sotto-problema 2 — Formalizzazione della metrica pedagogica:** Occorre tradurre i principi psicopedagogici dell'apprendimento in un indice quantitativo composito, capace di combinare dimensioni quali lo *scaffolding*, la ripresa conversazionale e l'adattamento linguistico. Questa formulazione deve saper modulare la pesatura delle singole dimensioni in base all'obiettivo del turno didattico (insegnamento, correzione o verifica), azzerare il punteggio in presenza di allucinazioni fattuali (*safety gating*) e risultare calcolabile in maniera del tutto automatizzata e scalabile.
- **Sotto-problema 3 — Validazione empirica e calibrazione:** La metrica teorizzata deve essere validata su benchmark realistici a confronto con valutazioni umane esperte. L'obiettivo è verificarne la solidità metodologica e statistica, accertando la correlazione con il giudizio degli esperti, la capacità di discriminare tra strategie didattiche maieutiche e rivelazione diretta della soluzione, e la coerenza delle stime tra modelli linguistici di diversa taglia e architettura.

1.3 Contributi della tesi

La tesi risponde ai sotto-problemi enunciati introducendo tre contributi cardine sul piano ingegneristico, metodologico e sperimentale:

1.3.1 Contributo 1 — Il framework `rag_eval`

Sul piano ingegneristico, la tesi introduce `rag_eval`, un framework software modulare ed estendibile progettato per la valutazione sistematica di pipeline RAG didattiche e general-purpose. L'architettura del sistema adotta una struttura a livelli disaccoppiata (*Interface* → *Pipeline* → *Metrics* → *Datasets* → *Reports* → *Core/Utils*) mirata a garantire manutenibilità e facilità di estensione.

Al suo interno, la piattaforma integra una suite di 15 metriche, combinando 10 indicatori standard dello stato dell'arte (modelli LLM-as-a-Judge, metriche lessicali e allineamenti su vettori densi) con 5 metriche specializzate per la valutazione pedagogica. L'interazione con i modelli linguistici è mediata da un'astrazione multi-provider (`LLMClient`) che uniforma l'accesso sia ad API commerciali (OpenAI, Anthropic, Google GenAI) sia a motori di inferenza open-weight su hardware locale (Ollama, vLLM), integrando un meccanismo di caching deterministico basato su impronte crittografiche SHA-256, politiche di retry con backoff esponenziale e monitoraggio dei consumi di token.

Il motore di orchestrazione asincrono, sviluppato su Python `asyncio`, assicura scalabilità e resilienza mediante checkpointing su disco *crash-safe*, isolamento degli errori a livello di singolo campione e controllo del rate limiting tramite semafori. L'ecosistema si completa con un generatore automatico di dataset sintetici con iniezione di passaggi distrattori, adattatori pronti all'uso per i principali benchmark accademici (HotpotQA, SQuAD 2.0, Natural Questions, TriviaQA, RGB, RECALL, MRBench v3 e MathDial), callback non invasivi per il tracciamento di catene LangChain LCEL e un sistema di reporting multi-target in grado di produrre esportazioni tabulari (JSON, CSV) e dashboard HTML interattive, completamente autonome e fruibili offline.

1.3.2 Contributo 2 — Il Pedagogical Evaluation Score (PES)

Sul piano metodologico, viene teorizzata e formalizzata la metrica composita **Pedagogical Evaluation Score (PES)**, strutturata secondo una pipeline gerarchica a tre livelli funzionali:

1. **Livello 0 — Faithfulness Safety Gate (Filtro di Fedeltà Fattuale):** La risposta prodotta dal tutor y viene scomposta in enunciati atomici (*claims*) la cui consistenza fattuale è verificata a fronte dei contesti documentali recuperati dal RAG. Qualora l'indice di fedeltà scenda al di sotto della soglia critica τ (fissata di default a $\tau = 0.50$ e calibrabile a $\tau = 0.25$ in contesti tutoriali complessi), l'intervento è considerato allucinatorio e il punteggio globale viene azzerato istantaneamente, interrompendo la pipeline:

$$\text{Faithfulness}(y) < \tau \implies \text{PES}(y) = 0 \tag{1.1}$$

2. **Livello 1 — Pedagogical State Intent Classifier (Classificazione dell'Intento Didattico):** Superato il filtro di sicurezza, un classificatore basato su LLM analizza la dinamica della conversazione e assegna il turno a uno di tre stati didattici discreti:

   - $\mathcal{S}_A$ (*Concept Teaching*): spiegazione ed elaborazione di un concetto teorico;
   - $\mathcal{S}_B$ (*Error Remediation*): correzione puntuale e bonifica di un fraintendimento dello studente;
   - $\mathcal{S}_C$ (*Socratic Assessment*): stimolazione maieutica della comprensione tramite domande-guida aperte.

   Lo stato identificato $\mathcal{S}$ determina dinamicamente il vettore di ponderazione delle sotto-metriche $\mathbf{w}(\mathcal{S}) = (w_1(\mathcal{S}), w_2(\mathcal{S}), w_3(\mathcal{S}))$.

3. **Livello 2 — Calcolo Parallelo delle Sotto-metriche Specializzate:**

   - M_1 (*Conversational Uptake & Context Elaboration*): stima la coesione dialogica e la capacità del tutor di recepire e sviluppare i contributi dello studente, combinando modelli neurali di *Natural Language Inference* multilingue (`mDeBERTa-v3-base-mnli-xnli`) con indici di sovrapposizione lessicale;

- M_2 (*Linguistic Adaptation*): misura l'adattamento stilistico fondendo formule di leggibilità normalizzate (adattamento del *Flesch Reading Ease*), coefficienti di tentatività socratica (densità di costrutti esplorativi) e analisi del sentiment orientata al rinforzo costruttivo;
- M_3 (*No Immediate Disclosure & Helpfulness*): quantifica, tramite giudici LLM vincolati da prompt dedicati, la capacità del tutor di non svelare prematuramente la soluzione finale (*NID Score*) e l'efficacia complessiva dell'aiuto fornito (*Helpfulness Score*).

L'indice finale è calcolato mediante una **media geometrica pesata**:

$$\mathrm{PES}(y) = \begin{cases} 0 & \text{se Faithfulness}(y) < \tau \\ \prod_{i=1}^{3} M_i^{\overline{w}_i(\mathcal{S})} & \text{altrimenti} \end{cases} \tag{1.2}$$

dove $\overline{w}_i(\mathcal{S}) = \frac{w_i(\mathcal{S})}{\sum_{j=1}^{3} w_j(\mathcal{S})}$ rappresenta il peso normalizzato. L'adozione della media geometrica rispetto a quella aritmetica è fondamentale: essa assicura che una grave carenza anche su una singola dimensione (come la rivelazione diretta della risposta, $M_3 \approx 0$) penalizzi severamente il giudizio sintetico, imponendo un profilo pedagogico equilibrato.

1.3.3 Contributo 3 — Validazione empirica multi-benchmark

Sul piano sperimentale, la validazione empirica è stata condotta su due benchmark complementari:

- **MRBench v3** [15]: corpus di 2,475 record con annotazioni umane multidimensionali su dialoghi di tutoraggio matematico generati da 9 modelli (compresi GPT-4, Claude 3.5 Sonnet, Gemini 1.5 Pro, Llama 3.1 8B/405B, Phi-3, Mistral, affiancati da baseline umane di tipo *Expert* e *Novice*);
- **MathDial** [16]: selezione di 2,040 turni estratti da interazioni reali di tutoraggio matematico, annotati per mosse didattiche (*Teacher Moves*: `probing`, `focus`, `generic`, `telling`).

I riscontri sperimentali confermano la solidità metodologica e la capacità diagnostica del PES su entrambi i benchmark.

Su MRBench v3, il test non parametrico di Kruskal-Wallis rivela una separazione statisticamente significativa tra i tre livelli qualitativi annotati dagli esperti (*Yes / To some extent / No*, con $H = 6.06$, $p = 0.048$), confermando che i punteggi medi seguono l'andamento monotonamente decrescente atteso. La correlazione per ranghi di Spearman con il giudizio umano risulta debole in valore assoluto ($\rho = +0.059$, $p = 0.003$), un esito riconducibile alla compressione della varianza nei punteggi del benchmark, ma statisticamente significativa e coerente con la direzione teorica prevista. La calibrazione dinamica dei pesi — che azzera l'uptake ($w_1 = 0$) negli

stati di correzione ($\mathcal{S}_B$) e verifica ($\mathcal{S}_C$), evitando che il tutor amplifichi gli errori dello studente — incrementa il differenziale diagnostico tra risposte ottime e insufficienti del 28.2%.

La validazione più incisiva proviene dal corpus MathDial, dove il PES evidenzia una spiccata capacità di penalizzare la rivelazione prematura della soluzione: impiegando il valutatore locale open-weight Llama 3.1 8B, le risposte di tipo *telling* (spoilering diretto) subiscono una decurtazione del -27.4% rispetto alle strategie di scaffolding maieutico ($+37.7\%$ di premio per lo scaffolding), con significatività statistica estrema (ANOVA $F = 71.98$, $p = 3.04 \times 10^{-44}$) e una dimensione dell'effetto pari al 9.59% della varianza spiegata ($\eta^2 = 0.0959$). A ulteriore riprova della scalabilità e della generalizzabilità del formalismo, una valutazione condotta impiegando un modello di frontiera proprietario (*GPT-5 Nano*, $N = 1{,}197$) con gate di fedeltà disattivato sui turni puramente maieutici registra un ulteriore balzo prestazionale: la penalizzazione dello spoilering si accentua al -30.5% ($+43.8\%$ di vantaggio per lo scaffolding), la varianza spiegata sale a $\eta^2 = 0.1244$ (12.44%, un incremento relativo del $+29.7\%$) e la correlazione di Spearman rispetto all'ordinamento didattico delle mosse più che raddoppia ($\rho = +0.1733, p = 1.59 \times 10^{-9}$ sul PES e $\rho = +0.2037, p = 1.12 \times 10^{-12}$ sulla componente M_3). Tale riscontro comprova come la metrica tragga pieno beneficio dall'impiego di motori valutativi avanzati senza mutare l'impianto matematico, pur mantenendo nei modelli locali open-source la configurazione cardine a tutela della riservatezza e della sovranità dei dati didattici.

L'analisi disaggregata delle componenti convalida l'efficacia selettiva del *Faithfulness Gate*, che intercetta il 14.0% delle risposte giudicate insufficienti dagli esperti a fronte dell'11.3% di quelle valide. Tra le singole dimensioni, la sotto-metrica M_3 (*No Immediate Disclosure & Helpfulness*) emerge come l'unico driver individualmente correlato in modo significativo al ground truth ($\rho = +0.043$, $p = 0.047$), confermando la centralità dello scaffolding nella qualità pedagogica percepita.

Infine, la graduatoria tra i modelli generativi vede primeggiare Gemini 1.5 Pro (0.5412) e Meta Llama 3.1 405B (0.5345), con la versione compatta Llama 3.1 8B (0.5191) pienamente competitiva rispetto a modelli commerciali proprietari come GPT-4 e Claude 3.5 Sonnet. Di particolare rilievo è la collocazione inferiore dei tutor umani esperti (*Expert Tutor*): a causa della *Curse of Knowledge* [17], gli esperti tendono a fornire spiegazioni dirette e sintetiche, venendo penalizzati dal PES che privilegia la progressività maieutica dello scaffolding.

1.4 Domande di ricerca

L'indagine scientifica condotta nel presente documento si articola attorno a quattro domande di ricerca fondamentali (*Research Questions*):

RQ1 (Infrastruttura di Valutazione). *È possibile superare i limiti delle metriche lessicali, dense e LLM-as-a-Judge esistenti mediante un'architettura software modulare, asincrona e multi-provider, capace di integrare organicamente dimensioni pedagogiche specializzate?*

RQ2 (Validità di Costrutto del PES). *Il Pedagogical Evaluation Score (PES) è in grado di discriminare tra risposte tutoriali efficaci e inefficaci, manifestando una correlazione statisticamente verificabile con il giudizio umano esperto?*

RQ3 (Interazione tra Livelli e Gating). *In quale misura il Faithfulness Gate, la classificazione dello stato pedagogico e la calibrazione adattiva dei pesi interagiscono nel determinare la capacità diagnostica complessiva della metrica?*

RQ4 (Scaffolding vs. Spoiling). *Con quale efficacia il PES discrimina tra strategie didattiche di scaffolding maieutico e comportamenti di rivelazione diretta della soluzione (spoilering)?*

1.5 Inquadramento metodologico e contesto applicativo

L'approccio metodologico adottato in questo lavoro integra in modo organico tre direttrici complementari: l'ingegneria del software, la formalizzazione matematica e l'analisi statistica inferenziale, declinate sulle esigenze operative della formazione digitale.

Sul piano ingegneristico, lo sviluppo del framework `rag_eval` fa leva su pattern architetturali consolidati quali *Registry*, *Strategy*, *Adapter* e *Dependency Injection*, garantendo contratti sui dati formalmente verificati a runtime mediante `Pydantic v2`. La robustezza e l'affidabilità del sistema sono presidiate da una suite completa di test unitari e di integrazione (`pytest`), analisi statica dei tipi (`mypy`) e standard rigorosi di qualità del codice.

Sul versante teorico-computazionale, la formulazione del PES traduce i principi della letteratura pedagogica in operatori analitici non lineari. Essa combina la scomposizione atomica delle asserzioni, filtri di sicurezza condizionali (*faithfulness gating*), macchine a stati discreti per la profilazione dell'intento dialogico e schemi di aggregazione basati su medie geometriche pesate. L'apparato di validazione sperimentale adotta a sua volta rigorose procedure di statistica inferenziale: coefficienti di correlazione per ranghi di Spearman, test parametrici (ANOVA) e non parametrici (Kruskal-Wallis e Mann-Whitney), procedure di correzione per confronti multipli (Bonferroni) e la quantificazione puntuale della dimensione dell'effetto (*effect size*) tramite le metriche d di Cohen ed η^2.

La genesi applicativa della ricerca si colloca nell'ambito della progettazione e sperimentazione di soluzioni avanzate per l'e-learning e l'alta formazione continua, contesti in cui l'adozione di agenti conversazionali esige standard computazionali oggettivi e riproducibili. Sebbene il framework e la metrica siano concepiti con un livello di astrazione generale per il tutoring dialogico, i requisiti pratici dei contesti formativi reali hanno guidato due scelte progettuali fondamentali:

- L'implementazione del supporto pienamente bilingue (italiano e inglese) per tutti i moduli di analisi stilistica e lessicale;
- La progettazione dell'infrastruttura multi-provider orientata a consentire l'esecuzione locale e privata di modelli open-weight (tramite istanze Ollama e

vLLM), a tutela della riservatezza dei dati didattici degli studenti e in piena conformità con i requisiti normativi del GDPR.

1.6 Struttura della tesi

Il resto della trattazione si articola nei successivi sei capitoli, organizzati come segue:

- **Capitolo 2 — Related Works e Stato dell'Arte:** Fornisce le basi teoriche e posiziona il lavoro rispetto alla letteratura, introducendo l'architettura RAG e la sua tassonomia, passando in rassegna le metriche di valutazione (lessicali, dense, LLM-as-a-Judge) e i relativi framework, analizzando i principi pedagogici di riferimento e i benchmark esistenti, e descrivendo le tecnologie adottate.
- **Capitolo 3 — Architettura del Framework `rag_eval` e Suite Diagnostica:** Illustra in dettaglio le scelte architetturali del framework realizzato, analizzando i modelli di dominio, le astrazioni multi-provider, l'orchestratore asincrono, la suite delle dieci metriche integrate con relativa profilazione computazionale e albero decisionale d'uso, il generatore di dataset sintetici, i moduli di integrazione e l'interfaccia a riga di comando.
- **Capitolo 4 — Il Pedagogical Evaluation Score (PES):** Espone la formalizzazione analitica del PES, dettagliando la motivazione didattica rispetto alla staticità delle metriche standard, la formulazione matematica del Faithfulness Gate, il classificatore dello stato pedagogico, le singole sotto-metriche e lo schema di calibrazione adattiva dei pesi.
- **Capitolo 5 — Validazione Sperimentale e Discussione:** Dettaglia l'infrastruttura di test, i corpora MRBench v3 e MathDial e il caso aziendale Clait S.r.l., analizzando i riscontri quantitativi (correlazioni, test di ipotesi, ranking tra LLM, analisi delle *Teacher Moves*, studi di ablazione e telemetria in produzione). Il capitolo integra la discussione critica dei risultati, le linee guida architetturali per tutor RAG, l'analisi delle minacce alla validità e la risposta alle domande di ricerca.
- **Capitolo 6 — Lavori Futuri:** Traccia le prospettive evolutive della ricerca, focalizzandosi sui modelli di uptake end-to-end, sull'integrazione di nuove dimensioni cognitive (supporto emotivo e ZPD), sull'estensione multimodale e sull'impiego del PES come funzione di ricompensa per l'allineamento tramite RLAIF e DPO.
- **Capitolo 7 — Conclusione:** Ripercorre sinteticamente i contributi e i risultati chiave conseguiti, delineando le considerazioni conclusive sull'impatto applicativo ed educativo del lavoro svolto.

Capitolo 2

Related Works e Stato dell'Arte

Il presente capitolo fornisce le basi teoriche e tecnologiche sulle quali si fonda l'intero lavoro di tesi. La trattazione si articola in quattro sezioni: la prima introduce il paradigma della Retrieval-Augmented Generation e il suo impiego in ambito educativo; la seconda passa in rassegna le metodologie di valutazione dei sistemi RAG, dalle metriche lessicali tradizionali fino al paradigma LLM-as-a-Judge; la terza analizza lo stato dell'arte nella valutazione pedagogica dei sistemi conversazionali, collegando le teorie dell'apprendimento alla formalizzazione computazionale; la quarta, infine, descrive le principali tecnologie e gli strumenti adottati nel presente lavoro.

2.1 Retrieval-Augmented Generation (RAG)

2.1.1 Fondamenti e motivazione

I modelli linguistici di grandi dimensioni (*Large Language Models*, LLM), basati sull'architettura Transformer [18], hanno dimostrato capacità straordinarie nella generazione di testo fluente e nella risoluzione di compiti linguistici complessi [19]. Tuttavia, essi presentano limitazioni strutturali ben documentate: i modelli linguistici non apprendono in tempo reale, ciò che sanno si ferma al giorno del loro addestramento e aggiornarli tramite fine-tuning è un'operazione costosa. Inoltre, tendono a "allucinare" [2, 3], ossia a inventare dettagli inesistenti pur presentandoli con un tono convincente e grammaticalmente impeccabile.

La **Retrieval-Augmented Generation** (RAG), introdotta formalmente da Lewis et al. [1], nasce per mitigare tali limitazioni integrando una componente di recupero delle informazioni (*retrieval*) a monte del processo generativo. L'intuizione fondamentale è semplice ma potente: anziché affidarsi esclusivamente alla conoscenza parametrica del modello, il sistema interroga una base documentale esterna per recuperare passaggi pertinenti alla query dell'utente, li concatena al prompt e li sottopone al generatore come contesto condizionante.

2.1.2 Architettura di riferimento

Per comprendere appieno il funzionamento di un sistema RAG, è utile immaginarlo non come un singolo blocco monolitico, bensì come una catena modulare che trasforma radicalmente il modo in cui l'intelligenza artificiale risponde alle domande. Se un classico modello linguistico opera come uno studente a cui viene chiesto di sostenere un esame "a libro chiuso" — affidandosi esclusivamente a ciò che ricorda dai pesi della propria rete neurale —, un'architettura RAG trasforma la prova in un esame "a libro aperto": prima si consultano con cura i testi e gli archivi di riferimento, poi si elabora la risposta.

Questo flusso operativo si articola in una pipeline sequenziale guidata da tre componenti funzionali strettamente coordinate:

1. **Retriever (Modulo di Recupero):** agisce come il "ricercatore" del sistema. Il suo compito consiste nello scandagliare una base di conoscenza esterna — spesso formata da decine di migliaia o persino milioni di frammenti di testo (*chunk*) preventivamente indicizzati — per individuare i passaggi più rilevanti rispetto al quesito posto dall'utente. Poiché inviare l'intero archivio all'LLM sarebbe impossibile per vincoli di memoria e costi computazionali, il retriever seleziona solo i contenuti più promettenti attraverso due filosofie complementari:

   - *Retrieval sparso*: affonda le radici nell'Information Retrieval classica ed è rappresentato da algoritmi probabilistici come BM25 [20]. Questo metodo si basa sulla coincidenza lessicale esatta delle parole chiave: è fulmineo ed eccelle nell'intercettare codici, acronimi o termini tecnici rari, ma manifesta una forte rigidità concettuale.
   - *Retrieval denso*: reso celebre da modelli come DPR (*Dense Passage Retrieval*) di Karpukhin et al. [21], supera i limiti delle sole parole chiave proiettando sia la richiesta dell'utente sia i documenti all'interno di uno spazio vettoriale continuo comune. Il recupero non si basa sulla sintassi, bensì sulla vicinanza geometrica (ad esempio calcolata tramite la similitudine del coseno tra embedding): in questo modo il sistema cattura il significato semantico e l'intento dell'utente, riconoscendo come pertinenti anche perifrasi e sinonimi.

   Nelle implementazioni all'avanguardia, questi due paradigmi vengono spesso fusi in un *recupero ibrido* (*hybrid search*), capace di combinare la precisione puntuale della ricerca per parole chiave con la flessibilità concettuale del recupero semantico.

2. **Context Augmentation (Arricchimento del Contesto):** rappresenta il vero ponte di collegamento tra la fase di ricerca e quella di scrittura. Recuperare informazioni corrette non basta se queste vengono riversate alla rinfusa nel prompt: frammenti duplicati, passaggi poco chiari o un ordine disorganico rischiano di distrarre l'LLM o di saturare la finestra di contesto. In questo stadio intermedio, i documenti recuperati possono essere filtrati, compressi e riordinati mediante modelli dedicati di *re-ranking* (come i cross-encoder), che esaminano

a fondo l'effettiva utilità di ciascun estratto. Le evidenze selezionate vengono quindi impaginate insieme alla domanda originale in un prompt strutturato (*augmented prompt*), spesso corredato da istruzioni di sistema (*system prompt*) che vincolano il modello a rispondere fondando le proprie argomentazioni unicamente sui dati forniti.

3. **Generator (Modulo Generativo):** è il "comunicatore", ovvero l'interfaccia intellettuale che dialoga con l'utente. Ricevuto il prompt arricchito con le fonti necessarie, un LLM analizza i testi, sintetizza i concetti e genera la risposta finale in un linguaggio naturale e scorrevole. Il vantaggio cruciale di questa separazione dei ruoli è duplice: da un lato, il modello generativo non deve più sforzarsi di memorizzare ogni singolo fatto del mondo — riducendo drasticamente le allucinazioni —, dall'altro può concentrare tutta la propria capacità computazionale sull'elaborazione logica, sulla sintesi e sull'adattamento del tono espositivo. Sul piano implementativo, il generatore può essere un servizio commerciale avanzato fruibile via API (come GPT-4 o Claude) oppure un modello open-weight eseguibile su server locali (come Llama o Mistral), una soluzione quest'ultima particolarmente preziosa in contesti educativi per tutelare la riservatezza dei dati e garantire la sovranità tecnologica dell'infrastruttura.

In sintesi, la forza del paradigma RAG risiede nel separare in modo netto la memoria a lungo termine del sistema (la conoscenza documentale, aggiornabile istantaneamente e a basso costo senza modificare la rete) dalla sua capacità di ragionamento e articolazione linguistica (demandata al modello generativo).

2.1.3 Tassonomia dei sistemi RAG

La letteratura recente [22] identifica tre generazioni architetturali di sistemi RAG, distinte per complessità e sofisticazione:

Naive RAG rappresenta l'implementazione di base del paradigma: un singolo passo di retrieval seguito dalla generazione. La query dell'utente viene utilizzata direttamente come chiave di ricerca, i primi k passaggi recuperati vengono concatenati al prompt, e il modello genera la risposta. Pur nella sua semplicità, questo approccio soffre di diverse limitazioni: sensibilità alla formulazione della query, assenza di meccanismi di verifica della pertinenza e rischio di "perdersi nel mezzo" (*lost in the middle*) quando il contesto è lungo [23].

Advanced RAG introduce moduli di ottimizzazione sia a monte (*pre-retrieval*) sia a valle (*post-retrieval*) della fase di recupero. Le tecniche pre-retrieval includono la riscrittura della query (*query rewriting*), la scomposizione in sotto-query (*query decomposition*) e l'espansione con ipotesi generate dal modello stesso (*HyDE*, Gao et al. [24]). Le tecniche post-retrieval comprendono il re-ranking dei passaggi tramite modelli cross-encoder, il filtraggio per rilevanza e la compressione contestuale. Il sistema Self-RAG [25] estende ulteriormente questo paradigma introducendo token di riflessione (*critique tokens*) che consentono al modello di valutare autonomamente la necessità del retrieval e la qualità

delle evidenze recuperate. Come illustrato nello schema comparativo della Figura 2.1, rispetto alla sequenzialità elementare del Naive RAG, l'Advanced RAG interpone stadi mirati di controllo sia prima dell'interrogazione dell'indice vettoriale sia a valle del recupero dei documenti.

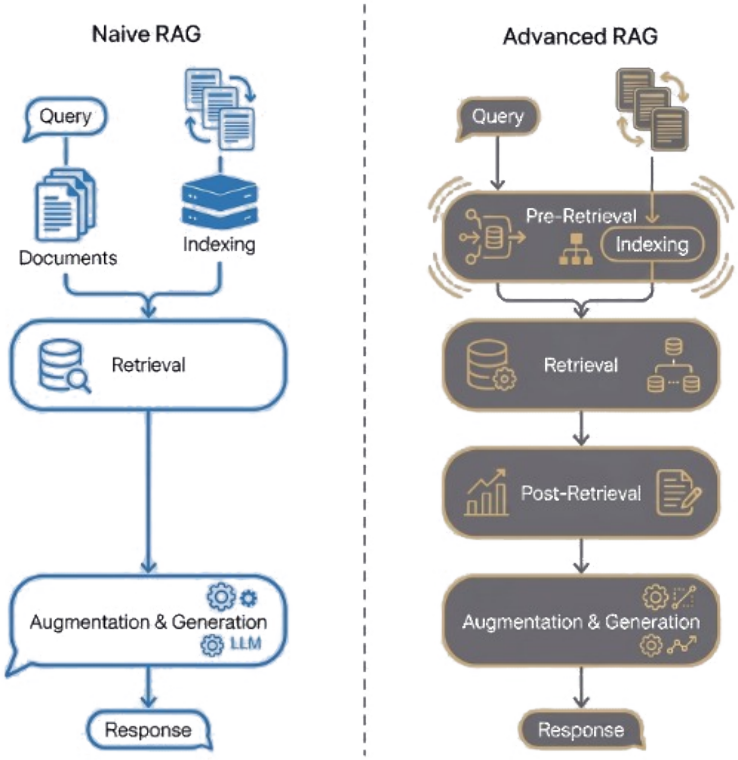


Figura 2.1: Confronto architetturale tra i paradigmi *Naive RAG* e *Advanced RAG* [22]. Rispetto al flusso lineare elementare della variante Naive, l'Advanced RAG introduce moduli specializzati di *pre-retrieval* (elaborazione della query e indicizzazione avanzata) e *post-retrieval* (re-ranking e filtraggio contestuale) a monte e a valle del recupero, ottimizzando la pertinenza del contesto prima della generazione con LLM.

Modular RAG propone una decomposizione flessibile dell'intera pipeline in moduli specializzati e ricombinabili. Framework come FLARE [26] e DSPy [27] consentono di progettare flussi di recupero e generazione adattivi, in cui il sistema decide dinamicamente quando e cosa recuperare, iterando il processo fino a raggiungere un criterio di sufficienza.

2.1.4 RAG in ambito educativo e tutoring conversazionale

L'applicazione del paradigma Retrieval-Augmented Generation al tutoraggio educativo conversazionale costituisce una delle frontiere applicative più promettenti, ma al contempo più impegnative, dell'intelligenza artificiale moderna. A differenza del classico *question answering* (QA) documentale — dove il criterio di successo

risiede nella generazione rapida di una risposta corretta e diretta —, un assistente per l'apprendimento deve perseguire finalità pedagogiche complesse: non risolvere il compito al posto dello studente, bensì sostenerne lo sviluppo cognitivo, stimolarne il ragionamento critico e guidarlo con pazienza maieutica verso la comprensione autonoma.

La ricerca educativa ha ampiamente documentato il valore trasformativo dell'istruzione personalizzata: già nel celebre studio sul *"2 Sigma Problem"*, Bloom [28] dimostrò come il tutoring individuale produca prestazioni superiori di due deviazioni standard rispetto alla didattica frontale in aula. I sistemi pionieristici di tutoraggio intelligente (ITS), quali AutoTutor [29] e i Cognitive Tutors [30], hanno confermato l'efficacia di un dialogo guidato passo-passo [31], scontrandosi tuttavia con elevati costi di sviluppo e una marcata rigidità di dominio. L'avvento dei Large Language Models e la loro integrazione con architetture RAG hanno riaperto questa sfida su scala globale, abilitando interazioni fluide e aperte (esemplificate da piattaforme come *Khanmigo* di Khan Academy e Duolingo Max). Recenti sperimentazioni controllate sul campo [32] confermano come un tutor basato su intelligenza artificiale, se opportunamente allineato ai principi dell'apprendimento attivo, possa eguagliare o persino superare i risultati didattici delle tradizionali lezioni d'aula.

Tuttavia, l'impiego diretto di un LLM per l'insegnamento pone forti rischi metodologici. Come evidenziato da Gupta et al. [33], i modelli di ultima generazione sono eccellenti motori di *question answering*, ma si rivelano spesso tutor inadeguati. In primo luogo, la loro natura probabilistica comporta una carenza di vero ragionamento simbolico, traducendosi in errori di calcolo e allucinazioni concettuali in discipline formali come la matematica [34]. In secondo luogo, le procedure di allineamento commerciale (RLHF) tendono a premiare la prontezza e la fornitura immediata del risultato: calato nel ruolo di docente, l'LLM manifesta la tendenza a rivelare prematuramente la soluzione (*immediate disclosure* o *spoilering*). Su benchmark come MathDial [16], ChatGPT finisce per svelare direttamente la risposta nel 66% dei casi e fornisce riscontri correttivi inesatti o fuorvianti nel 59% delle interazioni, vanificando lo sforzo cognitivo dello studente.

In questo contesto, il paradigma RAG assume un ruolo chiave: funge da garanzia epistemica, vincolando il modello a fonti verificate (manuali, sillabi ed eserciziari) e riducendo le allucinazioni fattuali. Tuttavia, la sola correttezza documentale non garantisce la qualità pedagogica del dialogo. Indagini recenti [34, 35] dimostrano infatti che i modelli attuali presentano un marcato ritardo rispetto ai docenti umani: faticano a calibrare la complessità del linguaggio all'età dello studente rischiando di generare sovraccarico cognitivo, tendono a un'eccessiva acquiescenza (*sycophancy*) che elude gli errori con elogi generici, e denotano difficoltà nel fornire un aiuto didattico realmente efficace.

Poiché i framework di valutazione convenzionali per sistemi RAG (quali RAGAS o TruLens) misurano unicamente la fedeltà del recupero e la coerenza fattuale, essi risultano del tutto ciechi verso la dimensione educativa. Si rende perciò necessario definire nuovi paradigmi di valutazione pedagogica automatizzata: dopo aver passato in rassegna le tecniche di valutazione RAG standard nella Sezione 2.2, la Sezione 2.3 analizzerà sistematicamente le teorie dell'apprendimento, i costrutti dialogici e le metriche computazionali su cui si fonda la proposta metodologica di questa tesi.

2.2 Valutazione dei sistemi RAG

La valutazione automatica della qualità delle risposte generate da un sistema RAG è un problema aperto e multidimensionale. Le metodologie sviluppate dalla comunità NLP e IR possono essere organizzate in tre famiglie principali, distinte per il tipo di rappresentazione linguistica su cui operano.

2.2.1 Metriche lessicali tradizionali

Le metriche di prima generazione costituiscono il fondamento storico della valutazione quantitativa nell'Elaborazione del Linguaggio Naturale (NLP) e nell'Information Retrieval (IR). Il principio operativo alla base di queste misure si basa sull'ipotesi che la qualità e la correttezza di un testo generato $\hat{y}$ possano essere stimate quantificando il grado di sovrapposizione lessicale superficiale — calcolata a livello di caratteri, token o sequenze di n-grammi — rispetto a una o più sequenze umane di riferimento (*ground truth*) $y \in \mathcal{Y}$.

Il principale pregio di questa famiglia di metriche risiede nella totale trasparenza deterministica e nell'estrema efficienza computazionale ($\mathcal{O}(|\hat{y}| + |y|)$), caratteristiche che ne consentono l'esecuzione in tempo reale su grandi volumi di dati a costo zero, senza dipendere da modelli neurali o chiamate ad API esterne. Nel contesto delle pipeline RAG tradizionali, esse sono state storicamente impiegate per stimare la fedeltà della risposta generata rispetto alle risposte corrette di riferimento fornite dai benchmark di Question Answering.

L'indice pionieristico e più celebre tra le misure basate sulla precisione è rappresentato dal **BLEU** (*Bilingual Evaluation Understudy*), introdotto da Papineni et al. [5] nell'ambito della traduzione automatica. La metrica si propone di quantificare la percentuale di n-grammi generati dal modello che compaiono anche nella risposta di riferimento. Se venisse calcolata una semplice precisione, un modello potrebbe ripetere più volte una singola parola corretta (ad esempio scrivendo dieci volte «il») e ottenere un punteggio ingannevole del 100%. Per prevenire questa distorsione, BLEU adotta una *precisione modificata* (p_n), in cui ogni n-gramma viene conteggiato solo fino al numero massimo di volte in cui compare effettivamente nel riferimento (*clipping*):

$$p_n = \frac{\sum_{\text{n-gram} \in \hat{y}} \text{Count}_{\text{clip}}(\text{n-gram})}{\sum_{\text{n-gram} \in \hat{y}} \text{Count}(\text{n-gram})} \tag{2.1}$$

D'altra parte, affidarsi unicamente alla precisione tenderebbe a premiare risposte brevissime composte da poche parole sicure. Per compensare questo squilibrio, BLEU introduce una penalità per brevità (*Brevity Penalty*, BP) che riduce il punteggio se la lunghezza della risposta generata $c = |\hat{y}|$ è inferiore a quella del riferimento $r = |y|$:

$$\text{BP} = \begin{cases} 1 & \text{se } c > r \\ \exp\left(1 - \frac{r}{c}\right) & \text{se } c \leq r \end{cases} \tag{2.2}$$

Il punteggio BLEU finale combina geometricamente le precisioni modificate ponderate fino a un ordine massimo N (convenzionalmente $N = 4$ con pesi uniformi $w_n = 1/N$),

moltiplicandole per la penalità di brevità:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \ln p_n\right) \tag{2.3}$$

Un problema operativo cruciale nell'applicazione di BLEU a risposte brevi o a singoli turni di dialogo è la scomparsa degli n-grammi di ordine superiore: se un singolo p_n risulta nullo (ad esempio nessun 4-gramma condiviso), la media geometrica azzera l'intero punteggio. Per ovviare a tale degenerazione nelle applicazioni conversazionali, si ricorre a strategie di lisciamento empirico, tra cui il consolidato *Method 1* di Chen e Goodman [36], che assegna un valore positivo infinitesimo ai conteggi nulli.

Se il BLEU privilegia la precisione, il framework **ROUGE** (*Recall-Oriented Understudy for Gisting Evaluation*), proposto da Lin [6] per la valutazione della sintesi automatica di testi, adotta una prospettiva complementare incentrata primariamente sulla *recall*. Nel task di generazione di risposte, infatti, la priorità informativa risiede spesso nell'accertare che i fatti salienti contenuti nella fonte siano stati effettivamente inclusi dall'agente, tollerando l'eventuale presenza di espressioni di raccordo accessorie.

All'interno della famiglia ROUGE si distinguono principalmente due varianti di riferimento. La prima è **ROUGE-N**, che misura la recall degli n-grammi del riferimento catturati nella risposta candidata:

$$\text{ROUGE-}N = \frac{\sum_{\text{n-gram}\in y} \text{Count}_{\text{match}}(\text{n-gram})}{\sum_{\text{n-gram}\in y} \text{Count}(\text{n-gram})} \tag{2.4}$$

In questo ambito, `ROUGE-1` valuta la copertura dei singoli unigrammi (indice della densità concettuale complessiva), mentre `ROUGE-2` considera i bigrammi, premiando la conservazione delle associazioni lessicali locali e una minima coerenza di fraseggio.

La seconda variante di rilievo è **ROUGE-L**, che supera i vincoli di contiguità rigida basandosi sulla *Longest Common Subsequence* (LCS). La sottosequenza comune più lunga individua la sequenza ordinata di token condivisi più estesa ammettendo inserimenti arbitrari o elementi intercalati, riconoscendo la somiglianza strutturale anche a fronte di perifrasi o ristrutturazioni sintattiche. ROUGE-L viene formalizzato come misura F bilanciata tra la recall ($R_{\text{LCS}} = \text{LCS}(y, \hat{y})/|y|$) e la precisione ($P_{\text{LCS}} = \text{LCS}(y, \hat{y})/|\hat{y}|$).

Nei compiti di Question Answering focalizzati su risposte puntuali, resi popolari da benchmark come SQuAD [37], il criterio di valutazione più rigoroso è costituito dall'**Exact Match** (EM). Si tratta di un indicatore strettamente binario ($\text{EM} \in \{0, 1\}$) che verifica la perfetta identità testuale tra la stringa predetta e il riferimento, previo strip dei segni di interpunzione, rimozione degli articoli e conversione in caratteri minuscoli. Pur rappresentando uno standard consolidato nel recupero estrattivo per entità chiuse (quali date, codici o formule numeriche), l'Exact Match si rivela eccessivamente rigido per la valutazione di enunciati a risposta aperta.

Per superare l'intransigenza binaria dell'Exact Match mantenendosi nell'alveo delle metriche lessicali, una soluzione ampiamente consolidata è la **Token-Level F1**. Questa misura modella il testo generato e il riferimento come collezioni di token individuali $T_{\hat{y}}$ e T_y: calcolando la precisione come frazione di token predetti

presenti nel riferimento e la recall come quota di token del riferimento recuperati, ne determina la media armonica bilanciata:

$$F_1 = \frac{2 \cdot |T_{\hat{y}} \cap T_y|}{|T_{\hat{y}}| + |T_y|} \tag{2.5}$$

In questo modo, la Token-F1 offre un profilo di valutazione continuo e sfumato, capace di tollerare parziali omissioni o aggiunte lessicali senza penalizzare la risposta in modo discreto.

Nonostante la loro standardizzazione e la rapidità di calcolo, l'impiego esclusivo di metriche lessicali nei moderni sistemi RAG conversazionali evidenzia criticità strutturali insormontabili. In primo luogo, come dimostrato empiricamente da Liu et al. [38] nei compiti di dialogo aperto e confermato da Callison-Burch et al. [39] nella traduzione, le metriche basate su sovrapposizione di parole presentano una correlazione pressoché nulla (e statisticamente non significativa) con il giudizio qualitativo umano espresso a livello di singolo turno conversazionale. In secondo luogo, operando unicamente sulla superficie dei caratteri, queste misure manifestano una totale cecità semantica: non riconoscono sinonimi o parafrasi e cadono nella trappola della polarità logica. Due frasi possono condividere oltre il 90% dei token ma asserire concetti opposti a causa di una singola negazione (ad esempio: *«Il triangolo è isoscele»* rispetto a *«Il triangolo non è isoscele»*), inducendo BLEU e ROUGE ad assegnare punteggi ingannevolmente prossimi all'unità a risposte concettualmente fallaci.

Infine, questo limite diventa particolarmente acuto nel tutoring didattico. Nel dialogo pedagogico non esiste una singola formulazione corretta: a fronte di un dubbio dello studente, un tutor efficace può porre una domanda maieutica, suggerire un contro-esempio o impostare una strategia di scaffolding. Misurare l'intervento dell'agente tramite la sovrapposizione di parole con una singola risposta di riferimento (*one-reference fallacy*) genera una distorsione paradossale: premia gli assistenti passivi che rivelano subito la soluzione ricopiando il testo, e penalizza gravemente le risposte didatticamente virtuose che guidano il ragionamento senza regalare il risultato finale.

2.2.2 Metriche basate su rappresentazioni dense

Per superare la rigidità intrinseca della corrispondenza lessicale, una seconda generazione di metriche ha traslato la valutazione dal piano discreto delle stringhe a quello continuo degli spazi vettoriali densi (*embedding spaces*). Con l'affermazione delle architetture Transformer [18] e dei modelli di linguaggio pre-addestrati (quali BERT [40] e le sue evoluzioni come RoBERTa e DeBERTa [41]), le parole e le frasi non vengono più trattate come simboli disconnessi, bensì mappate in vettori a dimensionalità fissa $\mathbb{R}^d$ capaci di codificare relazioni semantiche e contestuali profonde. In questo paradigma, la valutazione della risposta generata $\hat{y}$ rispetto a un riferimento y si traduce nel calcolo della prossimità geometrica nello spazio latente, riconoscendo come equivalenti formulazioni lessicalmente dissimili ma concettualmente affini.

L'approccio più immediato opera a livello di intero enunciato attraverso la **Semantic Similarity**. I modelli linguistici convenzionali richiederebbero l'elaborazione

congiunta delle due frasi tramite un'architettura cross-encoder (concatenando candidato e riferimento in un'unica sequenza separata da token speciali), una procedura computazionalmente onerosa che scala quadraticamente con il numero di confronti. Per ovviare a tale collo di bottiglia, il framework **Sentence-BERT** (SBERT), proposto da Reimers e Gurevych [42], adotta una rete siamese (*bi-encoder*) che processa ciascun testo in modo indipendente. Gli stati nascosti dell'ultimo layer vengono aggregati mediante un'operazione di pooling (tipicamente la media dei vettori dei token, o *mean pooling*) per produrre due rappresentazioni dense globali $\mathbf{v}_{\hat{y}}, \mathbf{v}_y \in \mathbb{R}^d$. La prossimità semantica viene quindi quantificata tramite la similarità del coseno, convenzionalmente riscalata nell'intervallo unitario $[0.0, 1.0]$:

$$\text{SemanticSimilarity}(y, \hat{y}) = \frac{1}{2}\left(1 + \frac{\mathbf{v}_y \cdot \mathbf{v}_{\hat{y}}}{\|\mathbf{v}_y\|_2 \, \|\mathbf{v}_{\hat{y}}\|_2}\right) \tag{2.6}$$

Questo metodo consente di determinare con eccezionale rapidità se il significato complessivo della risposta sia allineato al testo atteso, resistendo a sinonimi e ristrutturazioni sintattiche. Tuttavia, la compressione di un intero enunciato in un unico vettore denso introduce un inevitabile collo di bottiglia informativo, che tende a smussare i dettagli informativi specifici e a mascherare omissioni locali.

Per preservare la granularità analitica senza regredire al conteggio lessicale, Zhang et al. [7] hanno introdotto **BERTScore**, una metrica che trasferisce l'allineamento denso dal livello di frase a quello di singolo token contestualizzato. Anziché collassare la sequenza in un vettore globale, BERTScore preserva la successione degli embedding generati dagli strati profondi del Transformer per ciascun token del candidato $\hat{\mathbf{x}} = \langle \hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_k \rangle$ e del riferimento $\mathbf{x} = \langle \mathbf{x}_1, \ldots, \mathbf{x}_l \rangle$. Previa normalizzazione unitaria dei vettori, viene calcolata la matrice completa delle similarità del coseno a coppie $\hat{\mathbf{x}}_i^\top \mathbf{x}_j$. Mediante una procedura di allineamento avido (*greedy matching*), ogni token del riferimento viene associato al token generato con la massima similarità semantica per quantificare la recall, e specularmente ogni token generato viene abbinato al riferimento per la precisione:

$$R_{\text{BERT}} = \frac{1}{|y|} \sum_{\mathbf{x}_j \in y} \max_{\hat{\mathbf{x}}_i \in \hat{y}} \hat{\mathbf{x}}_i^\top \mathbf{x}_j, \quad P_{\text{BERT}} = \frac{1}{|\hat{y}|} \sum_{\hat{\mathbf{x}}_i \in \hat{y}} \max_{\mathbf{x}_j \in y} \hat{\mathbf{x}}_i^\top \mathbf{x}_j \tag{2.7}$$

La sintesi delle due componenti è affidata alla media armonica F_{BERT}, frequentemente calibrata attraverso pesature Inverse Document Frequency (IDF) per dare maggiore risalto ai termini informativi rari rispetto alle parole grammaticali ad alta frequenza. La letteratura ha ulteriormente arricchito questa famiglia introducendo metriche fondate sul trasporto ottimale, quali **MoverScore** [43], che modella allineamenti densi molti-a-molti tramite la *Earth Mover's Distance*, e modelli espressamente addestrati alla predizione della qualità come **BLEURT** [44].

Sebbene le metriche basate su embedding rappresentino un progresso sostanziale rispetto agli indici lessicali tradizionali, esse mostrano severe limitazioni teoriche e applicative nel contesto dei sistemi RAG e del dialogo tutoriale. In primo luogo, la vicinanza geometrica cattura la correlazione tematica e lessicale-distributiva (*cosa* viene detto), ma risulta interamente cieca di fronte alla funzione pragmatica e pedagogica dell'enunciato (*come* viene comunicato). In una sessione didattica, di

fronte a un problema matematico, una risposta di puro spoilering passivo (che fornisce direttamente il risultato finale) e una domanda socratica accuratamente calibrata (che stimola lo studente a individuare il passaggio logico mancante) condividono il medesimo intorno semantico; tuttavia, BERTScore e Sentence-BERT tenderanno a premiare sistematicamente la prima rispetto alla seconda, a causa della maggiore aderenza contenutistica con la soluzione di riferimento.

In secondo luogo, come documentato da Maynez et al. [2], gli encoder densi manifestano una pericolosa tolleranza verso discrepanze fattuali minute: inversioni di negazione, scambi di ruoli sintattici o alterazioni di singoli valori numerici producono variazioni minime nella similarità del coseno, consentendo a gravi allucinazioni concettuali di ottenere punteggi elevati. Infine, al pari degli indicatori lessicali, anche le metriche dense soffrono della dipendenza strutturale da una *ground truth* predeterminata, un vincolo insostenibile per la valutazione di interazioni tutoriali aperte e percorsi di apprendimento multi-turno non predefiniti.

2.2.3 Il paradigma LLM-as-a-Judge

Le metriche lessicali e dense valutano i testi calcolando sovrapposizioni di caratteri o distanze geometriche, ma non sono in grado di comprendere davvero il significato di una spiegazione, né di verificare se rispetti vincoli logici complessi. Per superare questa barriera, la ricerca ha introdotto il paradigma **LLM-as-a-Judge** [8]. L'idea di fondo è semplice e intuitiva: impiegare un modello linguistico di grandi dimensioni nel ruolo di esaminatore automatico, chiedendogli di leggere e giudicare una risposta proprio come farebbe un valutatore umano, seguendo criteri e rubriche esplicite fornite nel prompt. Il modello può operare confrontando due risposte alternative alla stessa domanda per decretare la migliore (*pairwise comparison*), oppure assegnando un voto diretto su una scala prefissata a un singolo testo (*single-answer scoring*).

Nel mondo del RAG, questo paradigma rappresenta una svolta metodologica fondamentale: permette di valutare il sistema senza dover disporre a priori di una risposta di riferimento scritta da esperti umani (*reference-free evaluation*). Il controllo viene infatti distribuito sulle tre dimensioni cardine della cosiddetta *RAG Triad*, che consentono di analizzare separatamente il lavoro del motore di ricerca e quello del generatore, come schematizzato nello schema concettuale di Figura 2.2:

- **Faithfulness (Fedeltà alle fonti):** controlla che la risposta generata dal modello sia rigorosamente ancorata ai documenti recuperati, impedendo il fenomeno delle allucinazioni. Operativamente, il giudice scompone la risposta in singole affermazioni atomiche e verifica, una per una, se ciascuna asserzione sia logicamente supportata dal contesto estratto, calcolando la percentuale di quelle confermate.

- **Answer Relevance (Pertinenza della risposta):** verifica quanto l'intervento risponda effettivamente al dubbio espresso dall'utente. Mentre la fedeltà guarda ai documenti di partenza, la pertinenza guarda alla domanda iniziale, penalizzando le risposte che divagano, che girano a vuoto o che aggiungono dettagli prolissi non richiesti.

- **Context Precision e Context Recall (Qualità del recupero):** valutano l'efficacia del motore di ricerca. La precisione misura quanti dei frammenti recuperati siano davvero rilevanti e se compaiano nelle prime posizioni del ranking, mentre il richiamo accerta se la ricerca abbia intercettato tutte le informazioni necessarie per poter rispondere in modo completo.

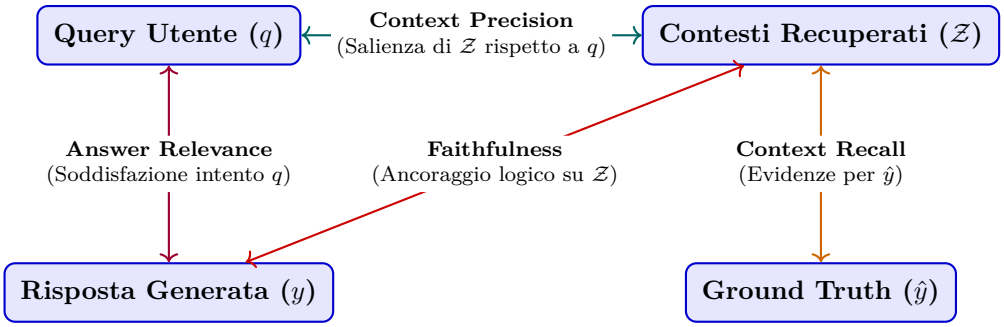


Figura 2.2: La Triade RAG estesa e le relazioni diagnostiche tra query, contesti recuperati, risposta generata e ground truth.

Nonostante la grande flessibilità e l'ottima correlazione con il giudizio umano, anche gli LLM manifestano delle debolezze sistematiche quando vestono i panni dei giudici [8, 45]. Spesso tendono a preferire risposte lunghe e articolate scambiando la prolissità per competenza (*verbosity bias*), si lasciano influenzare dall'ordine in cui vengono presentati i testi da confrontare (*position bias*), oppure finiscono per favorire i testi generati da modelli della loro stessa famiglia (*self-enhancement bias*).

Per neutralizzare questi difetti, la pratica comune adotta strategie di prompt engineering mirate: si impone al modello di spiegare il proprio ragionamento passo dopo passo prima di formulare il voto finale (*Chain-of-Thought*), si inverte l'ordine dei testi per verificare se il verdetto cambia, si adottano rubriche descrittive molto vincolanti con esempi chiari, o si ricorre a comitati formati da modelli differenti per bilanciare i punti di vista.

Tuttavia, anche se corretto e ben calibrato, il paradigma LLM-as-a-Judge convenzionale presenta un limite invalicabile nel contesto del tutoring didattico: valuta se una risposta è vera e pertinente, ma ignora del tutto se sia didatticamente efficace. Un assistente che fornisce immediatamente la soluzione completa a un esercizio otterrà un punteggio perfetto di fedeltà e pertinenza, ma dal punto di vista pedagogico avrà fallito il suo compito, privando lo studente dell'opportunità di riflettere e costruire attivamente la propria conoscenza.

2.2.4 Framework di valutazione RAG esistenti

Per tradurre le metriche sopra descritte in strumenti operativi, la comunità di ricerca e l'industria hanno sviluppato diversi framework open-source dedicati alla valutazione

sistematica delle pipeline RAG:

RAGAS (Retrieval Augmented Generation Assessment) [9] rappresenta il punto di riferimento più noto e adottato per la valutazione *reference-free*. Il framework orchestra chiamate automatiche a modelli LLM per quantificare le dimensioni fondamentali della RAG Triad: Faithfulness, Answer Relevance, Context Precision e Context Recall. Uno dei maggiori punti di forza di RAGAS risiede nella modularità dei prompt e nella capacità di generare dataset sintetici di test (*synthetic test-set generation*): partendo da un corpus di documenti grezzi, il framework può generare automaticamente terne realistiche di query, contesti rilevanti e risposte attese, facilitando la creazione di benchmark personalizzati senza dover ricorrere a costose annotazioni umane manuali.

ARES (Automated RAG Evaluation System) [10] nasce per superare il principale collo di bottiglia di RAGAS, ovvero l'onere computazionale ed economico derivante dal delegare ogni singola valutazione a chiamate API verso LLM proprietari di grandi dimensioni. ARES adotta un approccio semi-supervisionato in tre fasi: dapprima genera dati sintetici tramite LLM per coprire i diversi pattern di query e contesto; in seguito, utilizza un insieme ridotto di esempi annotati da esseri umani (poche centinaia) per addestrare classificatori dedicati e leggeri (come modelli encoder basati su DeBERTa); infine, sfrutta la tecnica statistica della *Prediction-Powered Inference* (PPI) per combinare le predizioni dei modelli leggeri con i dati umani, garantendo stime di qualità accurate corredate da formali intervalli di confidenza statistica.

TruLens si concentra sull'osservabilità e sulla diagnostica continua delle applicazioni basate su LLM. Oltre a formalizzare le metriche della RAG Triad (riformulate in Context Relevance, Groundedness e Answer Relevance), il framework si integra direttamente all'interno delle pipeline applicative (come LangChain o LlamaIndex). Attraverso strumenti di tracciamento strumentale (*instrumentation*), TruLens registra l'intero flusso di esecuzione di ogni interazione, monitorando latenze, costi operativi, consumo di token e punteggi di qualità passo dopo passo, e restituendo tali telemetrie all'interno di una dashboard interattiva pensata per facilitare il debugging rapido in fase di sviluppo.

DeepEval adotta una filosofia orientata all'ingegneria del software e alla Continuous Integration/Continuous Deployment (CI/CD). Concepito come un framework di testing in stile unit-test (integrabile nativamente con strumenti quali `pytest`), DeepEval permette agli sviluppatori di definire soglie minime di accettazione per ciascuna metrica RAG (ad esempio, imponendo che la fedeltà alle fonti non scenda al di sotto di 0.85). In questo modo, ogni modifica al prompt, all'algoritmo di chunking o al modello di embedding può essere testata automaticamente all'interno della pipeline di sviluppo software, bloccando eventuali regressioni prestazionali prima del rilascio in produzione.

Sebbene questi strumenti rappresentino contributi fondamentali per la valutazione dei sistemi RAG generici, tutti condividono un limite comune: rispondono

esclusivamente alla domanda *«La risposta è corretta, rilevante e ben fondata?»*, trascurando interamente la questione pedagogica *«La risposta promuove efficacemente l'apprendimento?»*.

2.3 Valutazione pedagogica dei sistemi conversazionali

La valutazione della qualità didattica di un agente conversazionale richiede il superamento del piano puramente informativo (correttezza della risposta) per abbracciare dimensioni interazionali, strategiche e socio-cognitive che caratterizzano l'insegnamento efficace. In questa sezione vengono analizzati i principali contributi teorici e computazionali che informano la progettazione della metrica proposta in questo lavoro.

2.3.1 Teorie dell'apprendimento di riferimento

L'efficacia di un dialogo tutoriale non può essere compresa riducendo l'insegnamento a un semplice trasferimento unidirezionale di informazioni. Al contrario, la matrice socio-costruttivista dell'apprendimento evidenzia come la costruzione della conoscenza scaturisca da una dinamica interattiva situata all'interno della **Zona di Sviluppo Prossimale** (ZPD), teorizzata da Vygotsky [11]. Tale dimensione individua lo spazio cognitivo compreso tra il livello di sviluppo attuale dello studente — ciò che è in grado di risolvere in totale autonomia — e il suo potenziale evolutivo, raggiungibile solo attraverso la mediazione di un pari o di un tutor più competente. Un intervento educativo virtuoso si colloca stabilmente lungo questo crinale: proporre compiti al di sotto della ZPD genera noia e disinteresse, mentre spingersi al di là dei confini superiori produce sovraccarico cognitivo, frustrazione e abbandono.

Per navigare operativamente all'interno della ZPD, l'interazione didattica si avvale del meccanismo di **scaffolding**, formalizzato da Wood et al. [12]. Ispirandosi alla metafora dell'impalcatura edilizia, gli autori descrivono l'insieme di aiuti flessibili e temporanei — quali indizi graduali, scomposizione del problema in sottoproblemi più semplici o suggerimenti strategici — che il tutor allestisce per consentire allo studente di eseguire compiti che eccedono le sue attuali capacità. La caratteristica qualificante dello scaffolding risiede nella sua natura dinamica: man mano che lo studente interiorizza i concetti e consolida la propria padronanza, l'impalcatura viene progressivamente ridotta e infine ritirata (*fading*), restituendo piena autonomia cognitiva al soggetto.

Sul piano conversazionale, lo scaffolding trova la propria naturale declinazione nell'**interrogazione socratica** (*Socratic questioning*) [46]. Anziché cedere alla via più immediata della spiegazione diretta (*telling*), il tutor adotta una postura maieutica: pone domande aperte, invita a considerare contro-esempi e stimola lo studente a verbalizzare le proprie riflessioni, guidandolo a scoprire autonomamente la soluzione. Questa impostazione si intreccia intimamente con il paradigma del **productive failure** [13]: l'errore, l'incertezza e il fallimento temporaneo non sono difetti da scongiurare a ogni costo, bensì tappe cognitive preziose. Lasciare che lo

studente si confronti con la difficoltà prima di ricevere un riscontro strutturato genera uno sforzo mentale che prepara il terreno a una comprensione concettuale molto più robusta e duratura rispetto alla passiva assimilazione della risposta corretta.

Questi principi convergono organicamente nella tassonomia del modello **ICAP** (*Interactive-Constructive-Active-Passive*) formulato da Chi [47], che correla la qualità degli esiti formativi con le modalità di ingaggio cognitivo dello studente. La teoria dimostra che i benefici di apprendimento crescono progressivamente man mano che lo studente passa dalla ricezione passiva di nozioni alla semplice manipolazione attiva di contenuti (come sottolineare o ripetere), fino ad approdare a modalità costruttive (generare nuove inferenze e spiegazioni) e interattive (co-costruire significato attraverso il dialogo paritario con il tutor). Ne consegue un criterio cardine per la valutazione dei tutor basati su intelligenza artificiale: il valore di un agente didattico non si misura sulla completezza enciclopedica o sull'eleganza espositiva delle sue risposte, bensì sulla sua capacità di provocare risposte costruttive e interazioni cognitive profonde da parte dello studente.

2.3.2 Conversational Uptake

Il concetto di **Conversational Uptake**, formalizzato computazionalmente da Demszky et al. [14], misura il grado in cui un interlocutore recepisce, riconosce e sviluppa i contributi del parlante precedente. Nel contesto del tutoring, un alto livello di uptake indica che il tutor sta prestando attenzione a ciò che lo studente ha detto, ricollegandosi al suo filo logico ed elaborando i concetti introdotti, anziché ignorare il suo contributo per seguire un percorso predeterminato.

Demszky et al. [14] hanno sviluppato modelli neurali per la classificazione automatica dell'uptake nelle interazioni insegnante-studente, dimostrando la sua correlazione con indicatori di engagement e apprendimento effettivo. Il loro framework, basato sulla *Next Utterance Classification* (*NUC*), valuta se l'enunciato del tutor è coerente con lo sviluppo dialogico o rappresenta una rottura tematica. Questo lavoro costituisce una delle basi metodologiche su cui la presente tesi costruisce la sotto-metrica M_1 del PES.

La ripresa conversazionale è una qualità didattica che dipende strettamente dal contesto: è preziosa quando si introducono nuovi concetti (perché permette di partire da ciò che lo studente già sa), ma può diventare controproducente quando bisogna correggere un errore, dove riprendere un'idea sbagliata dello studente rischia solo di rafforzare l'errore anziché correggerlo.

2.3.3 Readability e adattamento linguistico

L'efficacia didattica di un intervento non dipende soltanto dal contenuto informativo, ma anche dalla forma con cui viene comunicato. Un tutor efficace calibra il proprio registro linguistico in funzione del profilo dello studente, impiegando un lessico accessibile, una sintassi non eccessivamente complessa e un tono incoraggiante.

Sul piano quantitativo, la comprensibilità del testo viene tradizionalmente stimata attraverso formule statistiche di leggibilità, tra cui spicca l'indice **Flesch Reading Ease** [48]. Basandosi sul rapporto tra la lunghezza media delle frasi e il numero

medio di sillabe per parola, l'indice assegna un punteggio numerico che riflette la complessità strutturale del testo; sebbene ideato per la lingua inglese, adattamenti parametrici per altre lingue (tra cui l'italiano) ne garantiscono l'applicabilità anche al contesto formativo in esame.

Tuttavia, la mera leggibilità statistica non esaurisce la complessità dell'interazione tutoriale. La letteratura pedagogica sottolinea l'importanza di ulteriori sfumature stilistiche e pragmatiche che incidono direttamente sull'esperienza di apprendimento. In primo luogo, assume un ruolo decisivo il *linguaggio tentativo* (*tentativeness*): l'uso di formule dubitative o costrutti esplorativi (come «cosa succederebbe se provassimo a...» o «potresti considerare...»), tipici dell'interrogazione socratica, evita di calare la verità dall'alto e invita lo studente a una riflessione paritaria. A ciò si affianca la formulazione di un *feedback costruttivo*, che riconosce l'impegno dello studente e lo orienta al miglioramento, in piena sintonia con i principi del *growth mindset* [49]. Infine, l'adattamento linguistico deve farsi carico della gestione del carico cognitivo, in linea con la teoria del *Cognitive Load* di Sweller [50]: calibrare la densità delle spiegazioni e frammentare le nozioni complesse consente di non saturare la memoria di lavoro dello studente, mantenendone viva l'attenzione.

2.3.4 No Immediate Disclosure (NID)

Il principio di **No Immediate Disclosure** (NID) codifica una delle regole d'oro del tutoring efficace: un buon tutor non rivela immediatamente la soluzione o il risultato finale di un problema, ma guida lo studente a ricavarli in modo autonomo attraverso suggerimenti progressivi, domande mirate e la scomposizione del compito in passaggi più accessibili.

Questo principio affonda le proprie radici teoriche nello scaffolding [12] e trova una solida conferma empirica negli studi sul *productive failure* [13]. Fornire subito la risposta corretta — comportamento noto in letteratura come *telling* o *spoilering* — produce un'illusione di comprensione immediata, ma priva lo studente dello sforzo cognitivo necessario per riorganizzare i propri schemi mentali e sviluppare strategie di problem-solving durature.

Nel contesto dei Large Language Model, il rispetto del NID rappresenta un ostacolo particolarmente insidioso. I moderni modelli linguistici, addestrati e allineati tramite tecniche come il Reinforcement Learning from Human Feedback (RLHF), sono naturalmente ottimizzati per essere il più utili e diretti possibile (*helpfulness bias*): di fronte a una richiesta, tendono spontaneamente a fornire la risposta più esaustiva e immediata. Quando applicati all'istruzione, questo automatismo si traduce in una costante fuga di informazioni (*answer leakage*): l'agente finisce per risolvere gli esercizi al posto dell'utente anziché accompagnarlo nel ragionamento, trasformandosi da mentore didattico a mero esecutore passivo.

D'altra parte, applicare il NID non significa rifiutarsi ostinatamente di aiutare lo studente, cadendo nell'eccesso opposto di un silenzio frustrante. Una corretta applicazione pedagogica richiede una progressione calibrata di indizi (*hint progression*): si parte da domande di rilancio metacognitivo per chiarire il dubbio, si passa a suggerimenti concettuali su quale strategia adottare, e solo in caso di stallo prolungato si scompone il singolo passaggio difficile, senza mai anticipare l'esito conclusivo.

La quantificazione automatica del NID costituisce pertanto una delle sfide metodologiche più rilevanti della presente tesi. A differenza della fedeltà fattuale, che si limita a confrontare un testo con le fonti documentali, il NID esige una valutazione pragmatica sulla strategia comunicativa adottata nel tempo: la medesima risposta esatta può essere presentata in forma maieutica («Quale proprietà puoi applicare a questo punto?») oppure in forma rivelativa («Il risultato è 42»). Poiché i framework RAG tradizionali sono del tutto ciechi a questa distinzione, la formalizzazione di una metrica dedicata al NID rappresenta un tassello indispensabile per valutare l'effettivo valore educativo di un tutor conversazionale.

2.3.5 Benchmark e dataset di riferimento

La validazione empirica delle metriche pedagogiche richiede risorse specializzate, capaci di andare oltre la semplice correttezza della soluzione matematica per catturare la qualità delle strategie didattiche, l'interazione multi-turno e le dinamiche di scaffolding. Negli ultimi anni, la comunità di ricerca ha sviluppato diversi benchmark di riferimento volti a misurare il divario esistente tra la capacità di un modello di risolvere un problema (*solver*) e la sua effettiva abilità nell'insegnarlo a uno studente (*tutor*).

Nel presente lavoro, la sperimentazione e la calibrazione delle metriche si concentrano principalmente su due risorse cardine:

- **MRBench** [15]: è un benchmark specificamente progettato per valutare la capacità pedagogica dei tutor basati su LLM nella correzione degli errori matematici (*math mistake remediation*). La versione v3, adottata in questo studio, raccoglie 2,475 interazioni tutoriali generate da nove sistemi distinti (tra cui modelli commerciali di punta come GPT-4, Claude 3.5 Sonnet e Gemini 1.5 Pro, modelli open-weight quali Llama 3.1 8B/405B, Phi-3 e Mistral, nonché baseline umane esperte e novizie). Ogni dialogo è arricchito da annotazioni umane multidimensionali su aspetti cruciali quali l'identificazione dell'errore, la localizzazione della falla concettuale, il tono comunicativo e, soprattutto, la qualità della guida pedagogica (*providing guidance*).

- **MathDial** [16]: è un dataset costituito da circa 3,000 dialoghi tutoriali reali tra insegnante e studente, incentrati su problemi di ragionamento matematico multi-passo. La peculiarità fondamentale di MathDial risiede nell'annotazione fine-grained di ciascun turno del tutor secondo la tassonomia delle *Teacher Moves*: `probing` (domande socratiche di approfondimento), `focus` (guida mirata a un passaggio specifico), `generic` (interventi di apertura tematica) e `telling` (rivelazione diretta della soluzione). Questa dettagliata classificazione consente di testare con precisione la capacità di una metrica automatica di discriminare tra autentico scaffolding didattico e comportamenti passivi di spoilering.

Accanto ai dataset impiegati direttamente in questo progetto, il panorama scientifico recente ha visto emergere altri benchmark di notevole rilievo per la valutazione pedagogica degli agenti conversazionali:

- **MathTutorBench** [51]: estende la valutazione delle capacità tutoriali a compiti a risposta aperta (*open-ended*), strutturando l'analisi su tre macro-competenze fondamentali: la competenza matematica pura, la comprensione dello stato cognitivo dello studente (inclusa la diagnosi di misconcezioni latenti) e l'efficacia pedagogica complessiva. Il framework include inoltre un modello di ricompensa (*reward model*) appositamente addestrato per distinguere tra risposte didattiche formulate da insegnanti esperti rispetto a interventi formulati da novizi.

- **TutorBench** [52]: benchmark recentemente proposto per valutare in modo sistematico le abilità tutoriali degli LLM attraverso oltre un migliaio di scenari didattici curati da esperti. Il focus primario è incentrato sulla generazione di spiegazioni adattive, sulla fornitura di feedback immediatamente attuabili (*actionable feedback*) e sulla promozione attiva dell'apprendimento mediante la somministrazione progressiva di indizi, testando la capacità dell'agente di resistere all'impulso di fornire la risposta finale.

- **KMP-Bench** [53]: affronta esplicitamente il paradigma di transizione «From Solver to Tutor» nel contesto della scuola primaria e secondaria di primo grado (K-8). Il benchmark si articola in due moduli complementari: *KMP-Dialogue*, che analizza la coerenza pedagogica olistica nell'arco di lunghe sessioni multi-turno, e *KMP-Skills*, che valuta abilità atomiche mirate quali il rilevamento degli errori, la generazione di contro-esempi e la formulazione di domande maieutiche.

L'insieme di queste risorse testimonia un consenso scientifico ormai consolidato: per giudicare un tutor conversazionale intelligente non è sufficiente misurarne l'accuratezza nei compiti di Question Answering, ma è indispensabile disporre di protocolli e metriche capaci di quantificare la qualità pedagogica dell'interazione.

2.4 Tecnologie e strumenti utilizzati

Questa sezione descrive le principali tecnologie e gli strumenti software adottati nel presente lavoro, sia per lo sviluppo del framework `rag_eval` sia per la conduzione della campagna sperimentale di valutazione pedagogica.

2.4.1 Large Language Models

Il panorama dei modelli linguistici di grandi dimensioni vive una fase di espansione frenetica, scandita da rilasci continui e da una marcata polarizzazione tra soluzioni commerciali proprietarie accessibili via API e modelli *open-weight* a pesi aperti eseguibili su infrastrutture locali.

La frontiera tecnologica più recente vede il consolidarsi di architetture di nuova generazione con capacità di ragionamento profondo e autonomia agentica su orizzonti estesi. In casa OpenAI, l'evoluzione ha condotto dalla serie GPT-4o e dai primi modelli a catena di pensiero esplicita (come o1 e o3-mini) alla famiglia **GPT-5** (nello specifico la serie **GPT-5.6**), articolata su una gerarchia di capacità a tre livelli: **Sol**, modello ammiraglio orientato al ragionamento scientifico di frontiera,

alla risoluzione di problemi complessi e a flussi operativi agentici ad alta affidabilità; **Terra**, soluzione intermedia ottimizzata per il bilanciamento tra accuratezza e sostenibilità computazionale in carichi enterprise; e **Luna**, variante leggera e a bassissima latenza ideale per compiti e automazioni ad alta frequenza. Sul versante di Anthropic, la quinta generazione della famiglia Claude ha introdotto **Claude Opus 5**, modello di riferimento per il ragionamento agentico e la gestione di compiti multi-fase su contesti estesi (con finestre di contesto fino a un milione di token e meccanismi di *adaptive thinking* capaci di modulare dinamicamente lo sforzo computazionale dei token di pensiero in funzione della complessità del problema), affiancato da **Claude Sonnet 5** e dai sistemi di ricerca avanzata della serie **Fable 5.1**. Accanto a essi si collocano **Google Gemini** [54] (nelle generazioni 1.5 e 2.0 con contesto ultra-esteso e multimodalità nativa) e le innovazioni aperte di **DeepSeek-R1** [55] (basato su puro apprendimento per rinforzo) e della famiglia **Llama 3.1** (inclusa la versione da 405 miliardi di parametri, [56]) e **Llama 3.3** di Meta.

Nonostante l'attrattiva esercitata da questa frontiera in continuo mutamento, la scelta dei modelli analizzati nel presente lavoro (ed esposti nel Capitolo 5) risponde a un preciso principio di rigore metodologico e stabilità scientifica. Per garantire la confrontabilità con la letteratura e con le misurazioni di riferimento, la sperimentazione adotta il protocollo standardizzato **MRBench v3** (Maurya et al. [15]), che fissa il banco di prova su una selezione rappresentativa di architetture consolidate: i riferimenti proprietari **GPT-4** [57], **Claude 3.5 Sonnet** e **Gemini 1.5 Pro**, contrapposti ai modelli aperti **Llama 3.1** (nelle taglie 8B e 405B), all'efficiente **Mistral 7B** [58] (caratterizzato da *Sliding Window Attention* e *Grouped-Query Attention*) e alla famiglia compatta **Phi-3 Mini** [59] di Microsoft (3.8 miliardi di parametri), includendo inoltre il confronto diretto con tutor umani novizi ed esperti.

La coesistenza e il confronto sistematico tra queste famiglie assumono un'importanza cruciale per il design di `rag_eval`: il framework integra un'architettura modulare multi-provider capace di interfacciare in modo trasparente sia gli endpoint proprietari sia i modelli locali, consentendo di stabilire empiricamente se modelli leggeri o aperti possano offrire prestazioni di tutoring pedagogicamente efficaci senza dover ricorrere necessariamente a costosi giganti commerciali.

2.4.2 Modelli di embedding e Natural Language Inference

La valutazione quantitativa delle architetture RAG e delle dinamiche dialogiche richiede modelli discriminativi specializzati in due compiti complementari: la misurazione della similarità vettoriale nello spazio semantico — indispensabile per quantificare la pertinenza del contesto e la fedeltà della risposta nella Triade RAG — e l'analisi delle relazioni logico-deduttive tra turni conversazionali, necessaria per catturare l'uptake dialogico del tutor.

Nel panorama contemporaneo del *dense retrieval*, lo stato dell'arte per i modelli di embedding è tracciato da benchmark standardizzati come MTEB (*Massive Text Embedding Benchmark*, [60]), che valutano le rappresentazioni dense su decine di compiti eterogenei spaziando dalla classificazione al clustering fino al semantic search. Su questo terreno, i vertici delle classifiche sono presidiati da modelli multi-task su larga scala quali **BGE-M3** [61], la famiglia **E5** (*multilingual-e5-large*, [62]),

encoder fondati su decoder autoregressivi come GritLM e SFR-Embedding, oltre ai servizi commerciali proprietari ad altissima dimensionalità offerti da OpenAI (`text-embedding-3-large`), Cohere Embed v3 e Voyage AI. In modo del tutto analogo, nel dominio del *Natural Language Inference* (NLI), le massime accuratezze teoriche vengono raggiunte da pesanti cross-encoder o da architetture basate su DeBERTa-v3-large o xxlarge con centinaia di milioni di parametri.

Sebbene tali modelli guidino le graduatorie accademiche globali, la loro adozione all'interno di un framework di profilazione e valutazione in tempo reale come `rag_eval` comporterebbe forti limitazioni operative. Sistemi di tale mole richiedono infatti svariati gigabyte di memoria VRAM dedicata, introducono latenze nell'ordine delle centinaia di millisecondi per ciascuna invocazione oppure impongono vincoli economici e dipendenze da server remoti. Per queste ragioni, il presente lavoro adotta una strategia pragmatica orientata al miglior equilibrio tra sensibilità diagnostica, velocità di calcolo e riproducibilità deterministica.

Per il calcolo della similarità semantica e il monitoraggio della coerenza dialogica locale, la scelta è ricaduta su **Sentence-BERT** [42], nello specifico attraverso il modello compatto `all-MiniLM-L6-v2`. Con una taglia di soli 22 milioni di parametri e uno spazio vettoriale denso a 384 dimensioni, questo bi-encoder garantisce tempi di risposta sub-centisecondo — nell'ordine di pochissimi millisecondi per coppia frasale — e un'occupazione di memoria RAM inferiore a 100 MB, operando con la massima agilità direttamente su CPU generiche senza alcun bisogno di acceleratori grafici dedicati. L'assoluto determinismo del suo output assicura inoltre la piena riproducibilità dei punteggi lungo tutta la suite di test, sia su normali postazioni di sviluppo sia all'interno di pipeline automatizzate di integrazione continua (CI/CD).

Per la stima dell'uptake conversazionale (M_1), il framework si affida invece a **mDeBERTa-v3-base-mnli-xnli** [41]. Forte del meccanismo di *disentangled attention*, in cui le rappresentazioni lessicali e quelle posizionali relative sono disaccoppiate e calcolate separatamente, questo modello da circa 86 milioni di parametri è stato specificamente addestrato sui corpus multilingue MNLI e XNLI. Esso consente di classificare la relazione tra l'enunciato dello studente e la replica del tutor nelle tre categorie canoniche di implicazione, contraddizione o neutralità. In questo modo, `rag_eval` quantifica con precisione se il tutor stia accogliendo e sviluppando le premesse concettuali poste dallo studente, ottenendo un giudizio oggettivo, rapido e deterministico che evita l'onere computazionale e le oscillazioni stocastiche tipiche dei valutatori basati su modelli generativi.

2.4.3 LangChain

Nello sviluppo di applicazioni moderne alimentate da Large Language Models, **LangChain** [63] si è affermato come il punto di riferimento open-source per l'orchestrazione e l'integrazione di componenti complessi. La motivazione alla base del suo impiego risiede nella capacità di disaccoppiare la logica dell'applicazione dalle peculiarità dei singoli modelli generativi, offrendo un'infrastruttura standardizzata per collegare la generazione linguistica a basi di conoscenza esterne, indici vettoriali, memorie di dialogo e strumenti esterni.

Il cardine su cui poggia l'architettura moderna di LangChain è il **LangChain Expression Language (LCEL)**, un paradigma dichiarativo che riprende la logica di composizione a pipeline tipica degli operatori di pipe Unix (`|`). Attraverso l'astrazione unificata `Runnable`, LCEL consente di costruire flussi RAG componibili in modo lineare ed elegante, garantendo nativamente il supporto per chiamate sincrone, asincrone, streaming di token ed esecuzione parallela di rami indipendenti — come l'interrogazione concorrente dell'indice vettoriale e l'inoltro della domanda dello studente. Il framework mette inoltre a disposizione componenti modulari per ciascuna fase della pipeline, dalla suddivisione semantica dei documenti (tramite text splitter con sovrapposizione controllata) alla connessione con indici vettoriali e meccanismi di retrieval avanzati (ibridi o con compressione contestuale).

Un elemento di particolare interesse per il presente lavoro è costituito dal sistema estendibile di **callback** di LangChain. Questo meccanismo a eventi consente di osservare l'esecuzione delle catene dall'esterno: come verrà approfondito nel Capitolo 3, tale caratteristica permette a `rag_eval` di monitorare in modo del tutto trasparente l'intero ciclo di vita della richiesta — intercettando query, contesti recuperati, risposte generate e metriche temporali — senza dover modificare o appesantire il codice dell'applicazione didattica sotto esame.

2.4.4 Inferenza locale: Ollama e vLLM

L'esecuzione controllata di modelli a pesi aperti (*open-weight*) su infrastrutture locali costituisce un pilastro metodologico fondamentale del presente lavoro. Per coprire sia le esigenze di sperimentazione interattiva sia la necessità di elaborare grandi volumi di dati, il framework `rag_eval` si affida a due motori di inferenza complementari:

Ollama rappresenta la soluzione ideale per la prototipazione rapida e lo sviluppo quotidiano su singole postazioni. Basato sul motore C/C++ di `llama.cpp`, consente di eseguire con estrema semplicità modelli quantizzati in formato GGUF (tipicamente a 4 o 8 bit), riducendo l'impronta di memoria senza degradare sensibilmente la qualità delle risposte. Ollama gestisce in modo trasparente l'accelerazione hardware — supportando sia schede NVIDIA CUDA sia la memoria unificata Apple Silicon Metal — e semplifica la configurazione dei modelli tramite file dichiarativi `Modelfile`, esponendo inoltre un'API compatibile con lo standard OpenAI che ne rende immediata l'integrazione.

vLLM [64] è invece orientato al calcolo ad alte prestazioni (*high-throughput serving*) su server e cluster dedicati. Il suo punto di forza è l'algoritmo **PagedAttention**, che ottimizza radicalmente la gestione della memoria dinamica (KV-cache) durante la generazione autoregressiva, abbattendo la frammentazione tipica dei sistemi tradizionali. Questa architettura consente di processare contemporaneamente molteplici richieste con strategie di batching continuo e di distribuire modelli di grande scala su più GPU, rendendo vLLM il motore d'elezione per le estese sessioni di benchmark offline analizzate nel Capitolo 5.

La scelta congiunta di Ollama e vLLM nasce dall'esigenza di evitare i compromessi di altre piattaforme: soluzioni enterprise come Text Generation Inference (TGI) di Hugging Face impongono infatti infrastrutture più pesanti e vincoli containerizzati rigidi, mentre l'impiego diretto della riga di comando di `llama.cpp` o di motori

fortemente legati a specifiche GPU (come TensorRT-LLM) avrebbe introdotto un carico gestionale manuale o limitato la portabilità multipiattaforma del codice.

Infine, l'adozione dell'inferenza locale risponde a un requisito etico e istituzionale imprescindibile nel contesto educativo: la **sovranità e riservatezza dei dati**. Eseguire l'intera pipeline di tutoring e valutazione all'interno dell'infrastruttura locale garantisce che le interazioni, le difficoltà concettuali e i dati personali degli studenti non vengano mai inoltrati a server commerciali di terze parti, assicurando la piena conformità al Regolamento Generale sulla Protezione dei Dati (GDPR).

Capitolo 3

Architettura del Framework `rag_eval`

3.1 Principi architetturali e design pattern

La progettazione di un'infrastruttura computazionale per la valutazione sistematica di sistemi di *Retrieval-Augmented Generation* (RAG) impone vincoli ingegneristici stringenti: l'eterogeneità dei provider di modelli linguistici, la coesistenza di metriche computazionalmente dissimili (statistiche lessicali, allineamenti su vettori densi, inferenze NLI e giudici basati su LLM), la scalabilità su dataset massivi e la garanzia di riproducibilità scientifica e determinismo sperimentale.

Per rispondere a tali requisiti, il framework `rag_eval` è stato interamente ingegnerizzato in Python 3.10+ adottando un'architettura a livelli (*Layered Architecture*) fortemente disaccoppiata, modulare e orientata a contratti sui dati formalmente verificati a runtime.

3.1.1 Pattern architetturali adottati

L'impianto software fa leva su cinque design pattern fondamentali:

- **Registry Pattern (`MetricRegistry`):** implementa un meccanismo centralizzato per la registrazione e l'*auto-discovery* dinamico delle metriche. All'avvio dell'applicazione, il registro analizza il sottopacchetto `rag_eval.metrics` mediante le librerie standard `pkgutil` e `inspect`, individuando e registrando automaticamente tutte le sottoclassi concrete della classe astratta `BaseMetric`. Tale meccanismo abilita l'istanziazione dinamica *config-driven* (tramite CLI o file di configurazione YAML/JSON), eliminando l'accoppiamento rigido basato su importazioni esplicite.

- **Strategy Pattern (`BaseMetric`):** tutte le metriche implementano l'interfaccia astratta polimorfa comune `BaseMetric`, esponendo le coroutine asincrone `score(sample: TestSample) -> EvalResult` e `score_batch(samples:`

`list[TestSample]) -> list[EvalResult]`. Il motore di orchestrazione tratta ogni metrica in modo uniforme, astraendo se il calcolo sottostante sia una computazione simbolica istantanea o una pipeline neurale multi-passaggio.

- **Adapter Pattern (*Benchmark Adapters* e *LangChain Handler*):** normalizza schemi dati eterogenei provenienti da benchmark accademici (es. HotpotQA, SQuAD 2.0, MRBench) o da framework di sviluppo esterni (come LangChain LCEL) nella rappresentazione canonica di dominio `TestSample`.
- **Facade Pattern (`LLMClient` ed `EmbeddingClient`):** incapsula la complessità di cinque diversi motori di inferenza generativa (OpenAI, Anthropic, Google GenAI, Ollama, vLLM) e dei modelli di embedding densi locali. La facciata gestisce in modo trasparente meccanismi trasversali di caching crittografico, retry esponenziale, parsing resiliente di schemi JSON e monitoraggio della telemetria dei consumi.
- **Dependency Injection (Iniezione Dinamica delle Dipendenze):** le classi delle metriche non istanziano direttamente client di rete o modelli su disco, ma li ricevono tramite costruttore. La CLI sfrutta l'introspezione sulla segnatura dei metodi (`inspect.signature`) per iniettare in modo pigro (*lazy loading*) esclusivamente le risorse computazionali e le connessioni di rete necessarie alle metriche selezionate.

3.2 Modelli di dominio e contratti dati

Le strutture dati interne del framework sono formalizzate mediante modelli immutabili o semi-immutabili tramite la libreria `Pydantic v2`. Ciò garantisce una validazione dei tipi a tempo di esecuzione, serializzazione e deserializzazione ad alte prestazioni compilata in linguaggio Rust (attraverso `pydantic-core`) e consistenza dello stato interno.

3.2.1 `TestSample` — L'unità atomica di valutazione

Rappresenta un singolo punto nello spazio di valutazione, incapsulando la tupla fondamentale del paradigma RAG:

```
1 class TestSample(BaseModel):
2     question: str = Field(..., description="Query o richiesta
    formulata dall'utente.")
3     answer: str = Field(default="", description="Risposta generata
    dal sistema RAG o tutor.")
4     ground_truth: str = Field(default="", description="Risposta di
    riferimento aurea (se presente).")
5     contexts: list[str] = Field(default_factory=list, description="
    Passaggi testuali recuperati dal retriever.")
6     metadata: dict[str, Any] = Field(default_factory=dict,
    description="Metadati contestuali opzionali.")
```

Listing 3.1: Definizione del modello di dominio `TestSample`.

3.2.2 `EvalResult` — L'esito atomico della metrica

Incapsula il risultato della valutazione di una singola metrica su un campione:

```
class EvalResult(BaseModel):
    metric_name: str = Field(..., description="Identificativo univoco della metrica.")
    score: float = Field(..., ge=0.0, le=1.0, description="Punteggio scalare normalizzato in [0.0, 1.0].")
    sample_index: int = Field(default=0, description="Indice ordinale del TestSample associato.")
    reason: str = Field(default="", description="Giustificazione qualitativa o catena di ragionamento.")
    metadata: dict[str, Any] = Field(default_factory=dict, description="Dettagli intermedi e scomposizioni atomiche.")
```

Listing 3.2: Definizione del modello di dominio `EvalResult`.

3.2.3 `EvalReport` — Il container aggregato di sessione

Raccoglie la totalità degli oggetti `EvalResult` generati nel corso di una campagna di valutazione, calcola le statistiche descrittive aggregate (media, mediana, deviazione standard, percentili) e preserva la configurazione di esecuzione e i dati di consumo di token e costi:

```
class EvalReport(BaseModel):
    config: dict[str, Any] = Field(default_factory=dict)
    results: list[EvalResult] = Field(default_factory=list)
    summary: dict[str, Any] = Field(default_factory=dict)
```

Listing 3.3: Definizione del container di sessione `EvalReport`.

Il metodo statico `load_from_json(path)` garantisce la serializzazione *round-trip*: un report esportato su disco può essere deserializzato programmaticamente in qualsiasi momento per condurre ulteriori analisi statistiche, generare dashboard comparative o eseguire verifiche di regressione.

3.3 Astrazione multi-provider per modelli linguistici

Le metriche appartenenti alla famiglia *LLM-as-a-Judge*, unitamente al generatore di dataset sintetici, richiedono frequenti chiamate a modelli linguistici. L'architettura definisce l'interfaccia astratta `BaseLLMProvider`, che standardizza l'interazione attraverso due primitive asincrone:

- `complete(prompt: str, system: str, **kwargs) -> LLMResponse`
- `complete_json(prompt: str, system: str, **kwargs) -> dict[str, Any]`

3.3.1 Provider supportati nativamente

Il framework integra cinque driver di comunicazione specializzati:

- **`OpenAIProvider`:** interfaccia verso le API OpenAI ufficiali, con supporto nativo per *Structured Outputs* basati su schemi JSON rigorosi;
- **`AnthropicProvider`:** integrazione con il client asincrono di Anthropic (Claude 3.5 Sonnet), ottimizzato per compiti di ragionamento analitico multi-passaggio;
- **`GoogleProvider`:** connettore verso l'SDK Google GenAI per l'elaborazione su modelli Gemini a finestra di contesto estesa;
- **`OllamaProvider`:** driver per l'inferenza open-weight su hardware locale a costo zero tramite protocollo HTTP REST (`http://localhost:11434`), con convalida e correzione automatica di strutture JSON malformate;
- **`VLLMProvider`:** integrazione server-grade con motori vLLM distribuiti, interfacciati tramite protocollo REST ad alta efficienza.

3.3.2 Caching deterministico basato su SHA-256

Al fine di azzerare i costi computazionali ed economici legati a chiamate API ridondanti e garantire la totale riproducibilità scientifica degli esperimenti, la classe `ResponseCache` memorizza persistentemente le risposte dei modelli su disco. La chiave di memorizzazione univoca $\mathcal{K}_{\text{cache}}$ è calcolata tramite funzione di hash crittografico SHA-256:

$$\mathcal{K}_{\text{cache}} = \text{SHA256}\Big(\text{provider} \parallel \text{model} \parallel \text{system} \parallel \text{user} \parallel \text{temperature}\Big) \tag{3.1}$$

Qualora la chiave $\mathcal{K}_{\text{cache}}$ sia presente nel catalogo locale (`.cache/rag_eval/`) e il parametro di validità temporale `ttl_seconds` non risulti superato, la risposta viene restituita istantaneamente senza impegnare la scheda di rete né consumare crediti di inferenza.

3.3.3 Telemetria dei consumi e stima economica (`UsageTracker`)

La classe `UsageTracker` monitora in tempo reale i volumi cumulativi di token in ingresso (*prompt tokens*) e in uscita (*completion tokens*). Attraverso una tabella interna di riferimento (`MODEL_PRICING`), il framework associa a ciascun modello il rispettivo costo per milione di token, quantificando con precisione nel report finale l'esborso economico complessivo associato all'esperimento.

3.4 Client di embedding densi

Molte delle valutazioni effettuate dal framework non si limitano a verificare la corrispondenza esatta delle parole, ma richiedono di confrontare il significato semantico dei testi. Per calcolare metriche basate sulla similarità semantica (*Semantic Similarity*) o per suddividere i documenti in blocchi coerenti (*chunking*), è necessario convertire le frasi in vettori numerici, detti *embedding densi*. All'interno del framework, tutte queste operazioni sono gestite in modo centralizzato dalla classe `EmbeddingClient` (`src/rag_eval/utils/embeddings.py`).

Come modello predefinito è stato adottato `all-MiniLM-L6-v2` della libreria *Sentence Transformers* [42]. Questa scelta nasce da un ottimo compromesso pratico: il modello trasforma ciascun frammento di testo in un vettore di 384 numeri ed è in grado di catturare con precisione il senso delle frasi pur rimanendo molto leggero. Può infatti essere eseguito rapidamente in locale direttamente sulla CPU del computer, senza dover dipendere da schede video dedicate o da API esterne a pagamento che introdurrebbero ritardi e costi di rete. Il componente è comunque flessibile: se per casi d'uso particolari occorre una precisione maggiore, è possibile configurare modelli a dimensionalità più alta (come `all-mpnet-base-v2` a 768 dimensioni) o appoggiarsi a provider commerciali esterni come OpenAI.

Per evitare sprechi di risorse, il caricamento del modello segue la tecnica del *lazy loading* (caricamento pigro). I pesi della rete neurale non vengono caricati all'avvio del programma, ma solo nel momento in cui viene effettivamente invocato il metodo `embed()`. In questo modo, se un utente seleziona esclusivamente metriche simboliche o lessicali (quali Token F1, Exact Match o ROUGE), l'interfaccia a riga di comando si avvia all'istante senza consumare inutilmente memoria RAM.

Quando si devono analizzare grandi moli di dati, il client elabora i testi a piccoli blocchi (*batch*, impostati di default a 32 alla volta) per sfruttare al meglio il calcolo vettoriale. I vettori prodotti vengono normalizzati, consentendo di misurare quanto due testi siano affini tramite la formula della similarità coseno. Date due matrici di vettori $A \in \mathbb{R}^{m \times d}$ e $B \in \mathbb{R}^{n \times d}$, la similarità complessiva viene calcolata rapidamente sfruttando il prodotto scalare:

$$\mathrm{Sim}_{\cos}(A, B) = \frac{A\, B^{\top}}{\|A\|_2 \, \|B\|_2} \tag{3.2}$$

Inoltre, quando si confronta una risposta generata con la rispettiva soluzione di riferimento, la funzione ausiliaria `pairwise_cosine_similarity` misura la vicinanza coppia per coppia. Il valore restituito è un numero compreso tra 0 e 1 che descrive con precisione quanto i due testi siano vicini come significato, a prescindere dalle parole specifiche utilizzate.

3.5 Pipeline di valutazione e orchestrazione asincrona

L'esecuzione delle campagne di test e il coordinamento di tutte le metriche sui dataset sono affidati alla classe `Evaluator` (`src/rag_eval/pipeline/evaluator.py`). Data

la natura eterogenea delle metriche — che spaziano da rapidi confronti lessicali a chiamate verso modelli linguistici remoti o reti neurali locali — l'intero motore di orchestrazione è stato ingegnerizzato sfruttando la libreria asincrona `asyncio` di Python, garantendo elevata efficienza, parallelismo e robustezza operativa.

Il primo aspetto fondamentale della pipeline riguarda la gestione della concorrenza. L'invio simultaneo e incontrollato di centinaia di richieste a endpoint remoti (come OpenAI, Anthropic o Google) causerebbe frequenti blocchi per superamento dei limiti di frequenza (*rate limiting*, tipicamente errori `HTTP 429`), mentre l'esecuzione parallela non regolata su modelli locali (tramite Ollama o vLLM) saturerebbe rapidamente la memoria della scheda video. Per prevenire questi problemi, l'`Evaluator` regola l'accesso alle risorse tramite un semaforo asincrono (`asyncio.Semaphore`). Il parametro `max_concurrency`, personalizzabile dall'utente, è impostato di default a 5 per i provider cloud e a 1 per i modelli eseguiti in locale, stabilizzando così il flusso delle valutazioni ed evitando colli di bottiglia o interruzioni improvvise.

Un secondo elemento critico è la tolleranza ai guasti, implementata secondo una logica *fail-safe* isolata a livello di singolo campione. Nelle campagne di test composte da centinaia o migliaia di esempi, è del tutto normale imbattersi in anomalie isolate, come un timeout temporaneo della connessione o una risposta dell'LLM non conforme allo schema JSON richiesto. Anziché interrompere l'intero esperimento vanificando ore di calcolo già svolte, l'`Evaluator` intercetta le eccezioni per ciascuna combinazione di campione e metrica: in caso di errore non recuperabile, assegna un punteggio nullo (0.0) e registra dettagliatamente la motivazione diagnostica all'interno dell'oggetto `EvalResult`. In questo modo la sessione prosegue senza intoppi e l'utente può analizzare a posteriori le cause dei singoli fallimenti.

Infine, per garantire la massima sicurezza nei confronti di crash di sistema o chiusure impreviste del terminale, la pipeline integra un meccanismo di salvataggio incrementale su disco (*checkpointing*). Al termine dell'elaborazione di ciascun blocco di campioni (`batch_size`), lo stato parziale dell'`EvalReport` viene salvato automaticamente in formato JSON all'interno della cartella temporanea della sessione (`.rag_eval_cache/checkpoint_batch_{i}.json`). Qualora il processo venga interrotto forzatamente, l'esecuzione può essere ripresa dall'ultimo blocco completato senza dover ricalcolare le metriche sui campioni già processati. L'avanzamento complessivo della valutazione viene inoltre mostrato all'utente in tempo reale mediante barre di progresso interattive della libreria `Rich`, che indicano il numero di campioni elaborati, la percentuale di completamento e la stima del tempo residuo.

3.6 La Suite di Metriche RAG Implementate e Profilazione Computazionale

Oltre all'orchestrazione asincrona e alla gestione delle risorse di calcolo, un requisito fondante del framework `rag_eval` è la fornitura di una suite diagnostica completa di metriche di valutazione dello stato dell'arte. Tali indicatori consentono di disaccoppiare sistematicamente le prestazioni del componente di recupero documentale

(*retriever*) dalle qualità logico-linguistiche del modello generativo (*generator*), integrando le basi teoriche esposte nel Capitolo 2 all'interno di un'infrastruttura software modulare e uniforme.

3.6.1 Tassonomia e implementazione delle metriche

Tutte le metriche integrate nel framework ereditano dalla classe astratta `BaseMetric` (descritta nella Sezione 3.1.1) e sono suddivise in tre famiglie metodologiche complementari, come schematizzato nella Figura 3.1:

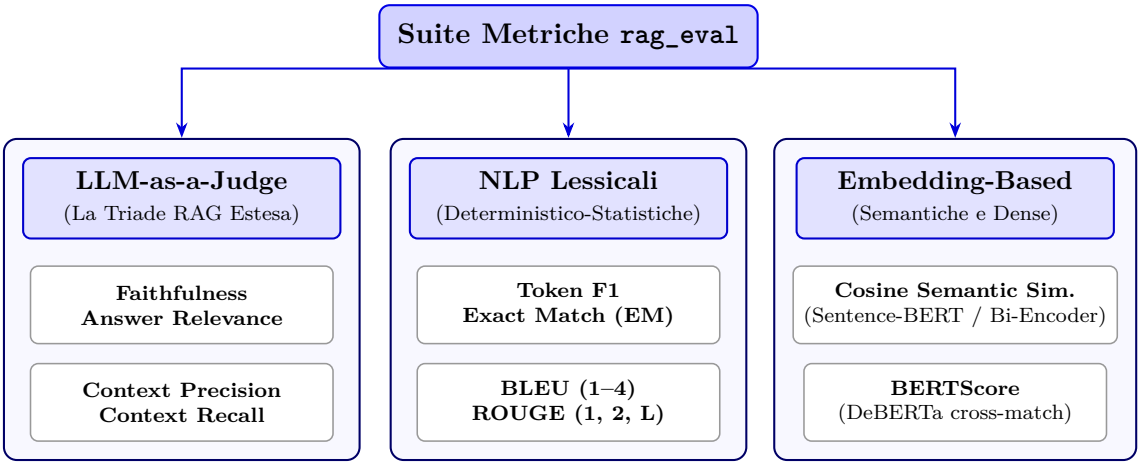


Figura 3.1: Tassonomia gerarchica delle metriche RAG integrate nel framework `rag_eval`.

1. **Famiglia LLM-as-a-Judge (`Faithfulness`, `AnswerRelevance`, `ContextPrecision`, `ContextRecall`):** formalizza le dimensioni della Triade RAG (analizzate nella Sezione 2.2.3). L'implementazione si appoggia all'astrazione `LLMClient` per interrogare modelli generativi mediante prompt strutturati e schemi JSON rigorosi. In particolare, la metrica `Faithfulness` decompone la risposta in enunciati atomici verificandone il supporto logico nei contesti estratti; `AnswerRelevance` valuta la pertinenza della risposta rispetto all'intento della query; infine, `ContextPrecision` e `ContextRecall` quantificano l'accuratezza e la completezza del recupero documentale rispetto al fabbisogno informativo;

2. **Famiglia NLP Lessicale (`TokenF1`, `ExactMatch`, `BLEU`, `ROUGE`):** implementa le metriche statistiche classiche (Sezione 2.2.1). Tali moduli non richiedono chiamate di rete né modelli neurali: i testi subiscono una pipeline locale di normalizzazione deterministica (conversione in minuscolo, rimozione di punteggiatura e articoli) prima del calcolo della sovrapposizione lessicale. BLEU

adotta la precisione modificata con penalità di brevità e lisciamento empirico [36], mentre ROUGE valuta il richiamo di n-grammi e la sequenza comune più lunga (LCS) con stemming di Porter;

3. **Famiglia Embedding-Based (`SemanticSimilarity`, `BERTScore`):** sfrutta le rappresentazioni vettoriali dense (Sezione 2.2.2). `SemanticSimilarity` si avvale dell'`EmbeddingClient` locale per calcolare la correlazione del coseno riscalata in $[0.0, 1.0]$ su vettori a 384 dimensioni generati da `all-MiniLM-L6-v2`; `BERTScore` opera a livello di singoli token contestualizzati calcolando la matrice di similarità con greedy matching.

3.6.2 Profilazione computazionale, latenze e costi

Il quadro comparativo riportato in Tabella 3.1 evidenzia i requisiti computazionali, le dipendenze software e i profili di costo associati a ciascun indicatore implementato.

Tabella 3.1: Quadro comparativo delle metriche di valutazione RAG implementate.

Metrica	Famiglia	Input	Dipendenza	Latenza Media	Costo
`Faithfulness`	LLM-as-a-Judge	$y, \mathcal{Z}$	`LLMClient`	$\sim$ 800–1500 ms	API / 0 loc.
`AnswerRelevance`	LLM-as-a-Judge	q, y	`LLMClient`	$\sim$ 500–1000 ms	API / 0 loc.
`ContextPrecision`	LLM-as-a-Judge	$q, \mathcal{Z}$	`LLMClient`	$\sim$ 800–1200 ms	API / 0 loc.
`ContextRecall`	LLM-as-a-Judge	$\hat{y}, \mathcal{Z}$	`LLMClient`	$\sim$ 800–1400 ms	API / 0 loc.
`TokenF1`	NLP Lessicale	$y, \hat{y}$	*Nessuna*	$<$ 1 ms	**Zero**
`ExactMatch`	NLP Lessicale	$y, \hat{y}$	*Nessuna*	$<$ 1 ms	**Zero**
`BLEU (1-4)`	NLP Lessicale	$y, \hat{y}$	`nltk`	$\sim$ 1–5 ms	**Zero**
`ROUGE (1,2,L)`	NLP Lessicale	$y, \hat{y}$	`rouge-score`	$\sim$ 2–10 ms	**Zero**
`SemanticSim`	Embedding	$y, \hat{y}$	`MiniLM-L6-v2`	$\sim$ 10–30 ms	**Zero**
`BERTScore`	Embedding	$y, \hat{y}$	`DeBERTa-xlarge`	$\sim$ 100–300 ms	**Zero**

Come si evince dai dati, la suddivisione modulare consente una gestione altamente calibrata delle risorse. Le metriche lessicali ed embedding operano a costo computazionale nullo e con latenze inferiori ai 30 ms, rendendosi ideali per test rapidi ed esecuzioni batch massicce. Al contrario, le metriche della Triade offrono una profondità diagnostica semantica insostituibile, ma comportano latenze tipiche comprese tra 500 e 1500 ms; l'accoppiamento con il sistema di caching SHA-256 e con provider locali (Ollama e vLLM) garantisce tuttavia la sostenibilità economica e la totale riproducibilità sperimentale anche per queste valutazioni ad alta intensità.

3.6.3 Criteri di selezione e albero di instradamento operativo

In virtù delle eterogeneità di runtime e costo, l'impiego della suite deve essere parametrato sugli obiettivi specifici dell'analisi. La Figura 3.2 sintetizza l'albero decisionale raccomandato per la selezione del profilo di valutazione ottimale all'interno del framework.

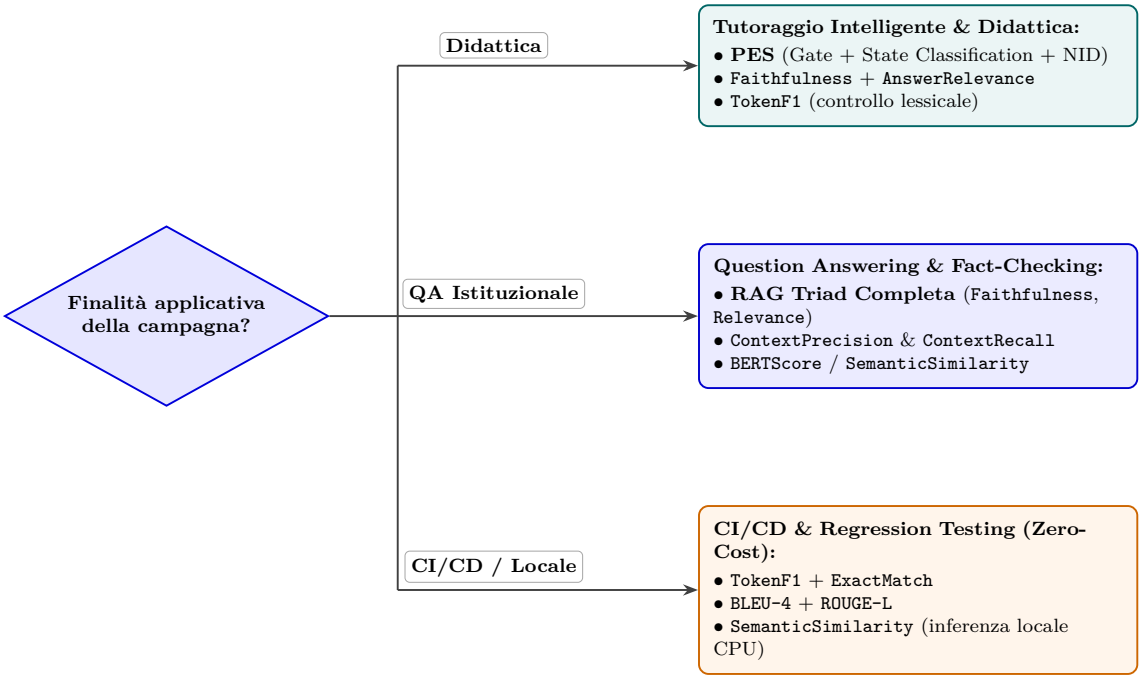


Figura 3.2: Albero di instradamento per la selezione del profilo di valutazione ottimale.

Si delineano tre macro-scenari operativi:

- **Continuous Integration e Regression Testing (Zero-Cost):** durante lo sviluppo continuo del codice o la modifica delle funzioni di chunking, l'utente può eseguire test automatici a costo zero sfruttando le metriche deterministiche (`TokenF1`, `ExactMatch`, `BLEU`, `ROUGE`) e la `SemanticSimilarity` calcolata localmente da Sentence-BERT, ottenendo riscontri sub-secondo;
- **Question Answering Istituzionale e Fact-Checking:** per la verifica di sistemi RAG generalisti focalizzati sulla correttezza documentale, la suite attiva la Triade RAG completa combinata a `BERTScore`, isolando le imperfezioni del retriever da quelle del generatore;
- **Tutoraggio Intelligente e Ambienti di Apprendimento:** nei contesti didattici, l'analisi esige di superare la staticità delle metriche convenzionali mediante l'attivazione del **Pedagogical Evaluation Score (PES)**, la cui teoria e formalizzazione analitica costituiscono il focus del Capitolo 4.

3.7 Generazione sintetica di dataset

Per ovviare alla scarsità di dataset annotati nei domini specialistici, il componente `SyntheticDataGenerator` (`src/rag_eval/datasets/synthetic.py`) implementa una pipeline per la generazione automatica di benchmark di test a partire da documenti non strutturati.

3.7.1 Partizionamento semantico dinamico (`ChunkingService`)

Nei sistemi RAG, prima di salvare un documento nel database vettoriale è necessario suddividerlo in porzioni più piccole, comunemente chiamate *chunk*. Molti strumenti tradizionali eseguono questo taglio contando semplicemente un numero fisso di caratteri o di parole; questo metodo, tuttavia, finisce spesso per spezzare le frasi a metà o separare concetti strettamente collegati tra loro. Per risolvere questo problema, il componente `ChunkingService` raggruppa il testo in base alla coerenza del significato:

1. Il documento di partenza viene innanzitutto suddiviso nelle sue singole frasi mediante espressioni regolari ben calibrate, facendo attenzione a non considerare come punto di fine frase i punti presenti in abbreviazioni comuni (come *es.*, *dott.*), date o numeri decimali;

2. Ciascuna frase s_i viene convertita in un vettore semantico utilizzando il modello `paraphrase-MiniLM-L6-v2`;

3. Le frasi vengono analizzate una alla volta in sequenza: a ogni passaggio, il sistema calcola la similarità coseno tra il vettore della nuova frase $\mathbf{e}_i$ e il centroide $\mathbf{c}_J$ (cioè il vettore medio) delle frasi già inserite nel blocco corrente J:

$$\mathrm{Sim}(\mathbf{c}_J, \mathbf{e}_i) = \frac{\left(\frac{1}{|J|}\sum_{j\in J}\mathbf{e}_j\right)\cdot \mathbf{e}_i}{\left\|\frac{1}{|J|}\sum_{j\in J}\mathbf{e}_j\right\|_2 \, \|\mathbf{e}_i\|_2} \tag{3.3}$$

4. Finché la nuova frase continua a parlare dello stesso argomento la similarità rimane alta, per cui la frase viene aggiunta al blocco. Non appena la similarità scende al di sotto della soglia empirica $\theta = 0.50$ (e il blocco ha raggiunto una dimensione minima di almeno 100 caratteri), il sistema comprende che l'argomento è cambiato: chiude quindi il chunk corrente e ne apre uno nuovo, mantenendo una leggera sovrapposizione di alcune frasi per non perdere il filo del contesto tra blocchi adiacenti.

3.7.2 Generazione multi-tipo e iniezione di contesti distrattori

Il generatore sintetico guida il modello LLM attraverso prompt strutturati per generare quesiti distribuiti su quattro classi cognitive distinte:

- **`factual`:** estrazione diretta di evidenze esplicite contenute nel frammento;
- **`multi-hop`:** sintesi di informazioni dislocate in posizioni distanti del passaggio;
- **`reasoning`:** quesiti che impongono deduzioni logiche non verbalizzate direttamente;
- **`comparative`:** comparazione di proprietà ed entità parallele.

A ciascuna domanda generata vengono affiancati passaggi distrattori (*distractor contexts*) estratti da porzioni ortogonali del documento originale, consentendo di misurare la precisione del modulo di recupero in presenza di rumore informativo.

3.8 Adattatori per benchmark standard

Per consentire confronti oggettivi e riproducibili, il framework supporta nativamente i principali benchmark della letteratura scientifica attraverso una famiglia di moduli adattatori. Ciascun adattatore eredita dalla classe base astratta `BaseBenchmarkAdapter`, il cui compito è normalizzare le strutture dati eterogenee dei vari dataset (spesso distribuiti su HuggingFace o in formato JSON/JSONL) e convertirle nella rappresentazione interna standard `TestSample`:

```
1 class BaseBenchmarkAdapter(ABC):
2     @abstractmethod
3     def load(self, split: str = "test", max_samples: int | None =
      None) -> list[TestSample]:
4         """Carica il dataset e restituisce una lista normalizzata
      di TestSample."""
5         pass
```

Listing 3.4: Interfaccia astratta `BaseBenchmarkAdapter`.

Di seguito vengono descritti gli otto benchmark integrati, evidenziando per ciascuno l'obiettivo di valutazione, la struttura del record originale e la relativa mappatura:

- **HotpotQA [65]:** ideato per testare il ragionamento a più passaggi (*multi-hop reasoning*). Ogni record richiede di combinare informazioni distribuite su più pagine di Wikipedia per giungere alla risposta.
  - *Struttura del record:* include i campi `question` (es. *"Were Scott Derrickson and Ed Wood of the same nationality?"*), `answer` (es. *"yes"*) e una lista di 10 paragrafi in `context`, di cui solo due contengono i fatti di supporto reali (*supporting facts*) e gli altri otto fungono da distrattori.
  - *Mappatura:* la domanda e la risposta popolano rispettivamente `question` e `ground_truth`, i paragrafi formano la lista `contexts`, mentre la tipologia (*bridge* o *comparison*) e la difficoltà (*level*) vengono memorizzate in `metadata`.

- **SQuAD 2.0 [66]:** uno dei riferimenti classici per la comprensione del testo (*reading comprehension*). Rispetto alla prima versione, include oltre 50.000 domande prive di risposta formulata nel testo (*unanswerable questions*).
  - *Struttura del record:* contiene un paragrafo `context`, una `question` e una lista `answers` con gli indici di inizio carattere nel testo; se la domanda non ha risposta, `answers` è vuoto.
  - *Mappatura:* l'adattatore mappa il passaggio in `contexts`, la domanda in `question` e la prima risposta disponibile in `ground_truth`, filtrando o etichettando opportunamente i casi non rispondibili per testare il rifiuto da parte del RAG.
- **Natural Questions [67]:** raccoglie query reali inserite dagli utenti sul motore di ricerca Google, associate a intere pagine di Wikipedia.
  - *Struttura del record:* presenta la query spontanea in `question_text`, l'intero testo HTML della pagina Wikipedia, una `long_answer` (il paragrafo di riferimento) e una o più `short_answers` (l'entità o la data precisa).
  - *Mappatura:* la query diventa la `question`, i passaggi rilevanti vengono isolati e inseriti in `contexts`, mentre la risposta sintetica forma la `ground_truth`.
- **TriviaQA [68]:** benchmark di cultura generale con oltre 95.000 terne composte da domande, risposte e testi di supporto non curati artificialmente, tratti da pagine web e voci enciclopediche.
  - *Struttura del record:* contiene il quesito `Question`, la risposta certificata con la lista di tutti i suoi sinonimi e alias validi in `Answer`, e frammenti testuali estratti dal web in `SearchResults`.
  - *Mappatura:* il quesito viene normalizzato in `question`, i frammenti web popolano `contexts`, mentre la risposta principale e i suoi alias vengono salvati in `ground_truth` e nei metadati per consentire una valutazione flessibile.
- **RGB (*Retrieval-Augmented Generation Benchmark*) [69]:** benchmark specialistico progettato per misurare quattro abilità critiche del RAG: tolleranza al rumore informativo, rifiuto in assenza di prove, resistenza a contesti negativi e robustezza verso informazioni controfattuali.
  - *Struttura del record:* comprende una `query`, la `ground_truth` e una serie di documenti `docs` appositamente manipolati per contenere fatti veritieri, rumore o affermazioni palesemente false.
  - *Mappatura:* la query e i documenti vengono mappati in `question` e `contexts`, mentre la condizione sperimentale (es. presenza di rumore o necessità di rifiuto) viene tracciata in `metadata`.
- **RECALL [70]:** valuta la capacità di un modello linguistico di aggiornare le proprie conoscenze pregresse quando il contesto recuperato dal RAG fornisce

informazioni nuove o contrastanti, anziché affidarsi ciecamente alla propria memoria parametrica pre-addestrata.

  - *Struttura del record:* presenta un quesito fattuale controfattuale `question`, il documento di supporto contenente la nuova informazione in `context` e la risposta attesa aggiornata in `ground_truth`.
  - *Mappatura:* popola i rispettivi campi di `TestSample` permettendo di quantificare la fedeltà del generatore al contesto effettivo.

- **MRBench v3 [15]:** benchmark dedicato al tutoraggio didattico-matematico multi-turno, corredato da dettagliate annotazioni qualitative fornite da esperti educatori.

  - *Struttura del record:* include l'esercizio matematico, lo storico della conversazione tra studente e tutor (`conversation_history`), la risposta candidata formulata dal tutor e le valutazioni umane su dimensioni pedagogiche quali `providing_guidance` (con etichette *"Yes"*, *"To some extent"*, *"No"*) e `actionability`.
  - *Mappatura:* lo storico della conversazione fino all'intervento dello studente forma la `question`, la risposta del tutor popola `answer`, mentre le annotazioni umane vengono convertite in punteggi numerici $(1.0, 0.5, 0.0)$ all'interno dei `metadata` per consentire la validazione della metrica PES.

- **MathDial [16]:** dataset di riferimento nell'ambito dell'AI educativa, costituito da circa 2.861 dialoghi reali di ripetizioni matematiche tra docenti e studenti, incentrati sulla diagnosi e correzione di errori concettuali.

  - *Struttura del record:* ciascun dialogo è segmentato in turni delimitati dal marcatore `|EOM|`. Per ogni risposta dell'insegnante è annotata la specifica intenzione pedagogica (*teacher move*), classificata in:
    * `probing`: domanda socratica che stimola lo studente a scoprire l'errore in autonomia (massima qualità pedagogica, score = 1.0);
    * `focus`: intervento che guida l'attenzione su un passaggio specifico del calcolo (score = 0.75);
    * `generic`: frase introduttiva o incoraggiamento generico (score = 0.5);
    * `telling`: spiegazione diretta della soluzione corretta che priva lo studente dello sforzo cognitivo (score = 0.0).
  - *Mappatura:* l'adattatore isola i singoli turni del docente. Il problema matematico e i turni precedenti del dialogo vengono incapsulati nella `question`, la replica del docente costituisce la `answer`, mentre la mossa pedagogica e il punteggio associato vengono memorizzati nei metadati per testare la correlazione con la sotto-metrica di scaffolding (M_3) del PES.

3.9 Integrazione LangChain

Nello sviluppo applicativo moderno, LangChain rappresenta uno dei framework più utilizzati per la prototipazione e l'orchestrazione di pipeline RAG attraverso il paradigma LCEL (*LangChain Expression Language*). Quando si desidera valutare un'applicazione già esistente realizzata con questo strumento, uno dei problemi più frequenti è dover modificare il codice sorgente per estrarre a mano le domande degli utenti, i frammenti documentali recuperati dai retriever e le risposte generate dall'LLM.

Per eliminare questo attrito e rendere la valutazione del tutto non invasiva, il framework fornisce la classe dedicata `RagEvalCallbackHandler` (`src/rag_eval/integrations/langchain.py`), che eredita direttamente da `BaseCallbackHandler` del pacchetto ufficiale `langchain_core`.

Il funzionamento del modulo sfrutta il sistema a eventi nativo di LangChain. Senza alterare in alcun modo la logica della catena, lo sviluppatore deve unicamente passare un'istanza del gestore al momento dell'invocazione (tramite il parametro di configurazione `callbacks=[handler]`). Durante l'esecuzione della catena, il gestore ascolta silenziosamente gli eventi del ciclo di vita:

1. All'avvio dell'esecuzione (`on_chain_start`), intercetta e memorizza la richiesta o la domanda iniziale dell'utente;
2. Non appena il componente di ricerca conclude il proprio lavoro (`on_retriever_end`), estrae tutti i documenti recuperati, ne conserva il testo e i metadati di provenienza (come titolo, fonte o identificativi di pagina) e rimuove in modo trasparente eventuali passaggi duplicati;
3. Al termine della generazione da parte del modello linguistico (`on_llm_end`), cattura la risposta testuale finale prodotta.

Tutti i dati raccolti durante il passaggio vengono automaticamente associati all'identificativo univoco della sessione (`root_id`). A fine esecuzione, è sufficiente richiamare il metodo `handler.get_sample()` (oppure `handler.get_samples()` nel caso di invocazioni multiple o in lotti) per ottenere direttamente uno o più oggetti `TestSample` già pronti e strutturati. Questi campioni possono essere immediatamente passati all'`Evaluator` per il calcolo delle metriche, consentendo di collaudare e monitorare pipeline RAG in produzione con pochissime righe di codice aggiuntivo.

3.10 Sistema di reporting multi-formato

La classe `ReportGenerator` (`src/rag_eval/reports/generator.py`) converte la struttura aggregata `EvalReport` in quattro formati specializzati:

1. **Rich Console Reporter:** visualizzazione su terminale con tabelle formattate a colori, barre di distribuzione Unicode e sinottico economico sui token utilizzati;
2. **JSON Exporter:** esportazione formale conforme a schema, comprensiva di metadati di sistema, timestamp e iperparametri di esecuzione;

3. **CSV Exporter (Wide Format):** matrice tabellare in cui ogni riga identifica un `TestSample` affiancato dalle colonne `{metric}_Score` e `{metric}_Reason` per tutte le metriche computate, pronta per l'importazione in ambienti di calcolo statistico come Pandas, R o SPSS;

4. **Interactive Standalone HTML Dashboard:** genera una dashboard HTML monolitica interattiva a pagina singola:

   - *Zero Dipendenze Esterne:* i sorgenti CSS e JavaScript di Bootstrap 5 e Chart.js sono memorizzati localmente e incorporati (*inlined*) direttamente nel sorgente HTML generato, garantendo la corretta fruizione grafica anche in ambienti isolati (*air-gapped*);
   - *Visualizzazioni Statistiche:* integra grafici radar del profilo delle metriche, istogrammi delle distribuzioni, tabelle con ordinamento dinamico e viste di dettaglio per singolo record con evidenziazione dei claim supportati e dei fallimenti del gate di fedeltà.

3.11 Command Line Interface (CLI)

L'interfaccia a riga di comando `rag-eval` (`src/rag_eval/cli/main.py`) è implementata tramite la libreria `Typer` ed espone quattro comandi:

- `evaluate`: esegue la pipeline di valutazione su un dataset o benchmark di riferimento;
- `generate`: avvia la sintesi di campioni QA a partire da file di testo non strutturati;
- `report`: rigenera le viste di reportistica (HTML, CSV, terminale) partendo da un report JSON esistente;
- `metrics`: interroga il `MetricRegistry` mostrando l'elenco delle metriche registrate e le rispettive dipendenze computazionali.

3.11.1 Iniezione intelligente delle dipendenze (`_instantiate_metrics`)

L'istanziazione delle metriche richieste dall'utente (es. tramite il flag `-metrics faithfulness,semantic_similarity,token_f1`) è gestita da una procedura factory:

1. Interroga `MetricRegistry` recuperando le definizioni di classe corrispondenti;
2. Ispeziona la segnatura del costruttore `__init__` tramite `inspect.signature`;
3. Istanzia e inietta `LLMClient` unicamente se la classe richiede esplicitamente il parametro `llm_client`;
4. Istanzia e inietta `EmbeddingClient` solo in presenza del parametro `embed_client`;

5. Istanzia direttamente le metriche lessicali prive di dipendenze pesanti.

Tale accorgimento assicura che l'invocazione di metriche puramente deterministiche mantenga tempi di esecuzione istantanei senza allocare risorse di rete o caricare tensori in memoria.

Capitolo 4

Il Pedagogical Evaluation Score (PES)

Questo capitolo presenta il contributo metodologico centrale della presente tesi: la progettazione, l'architettura computazionale e la formalizzazione analitica del **Pedagogical Evaluation Score (PES)** nella sua formulazione **Dinamico-Adattiva**. La trattazione è organizzata come segue: la Sezione 4.1 approfondisce i presupposti didattici e le motivazioni alla base della proposta, evidenziando i limiti strutturali delle tradizionali metriche RAG nei contesti educativi; la Sezione 4.2 descrive l'architettura a tre livelli della metrica e il flusso di elaborazione asincrono; le Sezioni 4.3, 4.4 e 4.5 illustrano in dettaglio i tre stadi operativi (il gate di sicurezza sulla fedeltà documentale, il classificatore dello stato pedagogico a pesatura adattiva e le sotto-metriche modulari); la Sezione 4.6 introduce la formulazione matematica unificata del PES Dinamico-Adattivo, analizzandone le proprietà formali, la non-compensatorietà e i gradienti di sensibilità; la Sezione 4.7 propone un'analisi qualitativa comparativa su turni dialogici reali; infine, la Sezione 4.8 sintetizza i trade-off computazionali e gli aspetti implementativi della soluzione.

4.1 Motivazione teorica e fondamenti didattici

4.1.1 I limiti delle metriche RAG tradizionali nei contesti educativi

La ricerca quantitativa nell'ambito della *Retrieval-Augmented Generation* ha prodotto un insieme consolidato di indicatori per valutare l'affidabilità dei sistemi generativi [9, 10]. Tali metriche — in particolare la fedeltà al contesto (*faithfulness*), la rilevanza della risposta (*answer relevance*) e la precisione del recupero (*context precision/recall*) — si concentrano principalmente sulla correttezza delle informazioni fornite e sull'accuratezza nell'estrazione dei passaggi documentali rilevanti.

Tuttavia, quando l'architettura RAG viene trasferita dall'ambito dell'interrogazione informativa aziendale al dominio dell'**apprendimento guidato dall'intelligenza artificiale**, tali dimensioni si dimostrano **strutturalmente insufficienti**. In un

contesto di tutoring, l'efficacia di un assistente virtuale non può essere misurata unicamente in base all'esattezza o all'esaustività della risposta. Al contrario, un modello linguistico che, a fronte del dubbio di uno studente, fornisca istantaneamente una soluzione completa, dettagliata e pronta all'uso, disattende uno dei cardini didattici della pedagogia moderna: il supporto maieutico progressivo o *scaffolding* [12].

Tale fenomeno, definito nel presente studio **The Spoiler Effect**, comporta una netta riduzione dell'impegno cognitivo dello studente. L'assimilazione profonda e duratura di concetti complessi non deriva dalla passiva ricezione di risposte preconfezionate, ma richiede un percorso attivo di confronto con la difficoltà (*productive struggle* o *productive failure*) [13]. L'erogazione diretta della soluzione priva il discente dell'opportunità di formulare ipotesi, verificare congetture e correggere autonomamente le proprie idee errate all'interno della Zona di Sviluppo Prossimale (ZPD) teorizzata da Vygotsky [11].

4.1.2 Principi pedagogici per la valutazione

La qualità pedagogica di un dialogo didattico non è una caratteristica monolitica riducibile a una misura di similarità semantica; essa costituisce un costrutto **multidimensionale e gerarchico**, che esige la convergenza armoniosa di cinque proprietà fondamentali:

1. **Accuratezza concettuale e ancoraggio documentale:** il tutor non deve generare allucinazioni concettuali o fattuali rispetto ai materiali didattici certificati ($\mathcal{Z}$). Un errore teorico fornito da un tutor autorevole rischia di consolidare nello studente convinzioni errate difficili da correggere in seguito;

2. **Adattamento dinamico all'intento tutoriale:** l'azione pedagogica deve conformarsi alle diverse fasi del dialogo: introdurre un nuovo concetto teorico richiede chiarezza espositiva ed esempi mirati; guidare la risoluzione di un errore di calcolo esige indizi graduati; consolidare le competenze acquisite richiede quesiti maieutici stimolanti;

3. **Non-rivelazione immediata e scaffolding maieutico (*No Immediate Disclosure*):** il sistema deve astenersi dal fornire direttamente la soluzione o i passaggi risolutivi conclusivi, proponendo invece domande guida e chiarimenti progressivi calibrati sul carico cognitivo dell'allievo [50];

4. **Continuità dialogica e ripresa delle risposte (*Conversational Uptake*):** il tutor artificiale deve dimostrare di recepire e valorizzare attivamente i contributi e i passaggi logici formulati dallo studente (*uptake*), evitando repliche stereotipate o disconnesse dal percorso svolto [14, 71];

5. **Registro comunicativo e tono costruttivo:** il linguaggio didattico deve coniugare un'elevata leggibilità con l'impiego di formulazioni non impositive che invitano alla riflessione (*socratic tentativeness*) e feedback costruttivi, promuovendo nello studente una mentalità di crescita (*growth mindset*) [49].

Per tradurre questo insieme di requisiti in un modello quantitativo rigoroso e direttamente integrabile nelle pipeline di valutazione automatica, in questo lavoro viene introdotto il **Pedagogical Evaluation Score (PES)**: una metrica gerarchica a tre livelli, potenziata da un meccanismo di arresto di sicurezza (*safety gating*) e da una modulazione dinamica dei pesi condizionata allo stato pedagogico del dialogo.

4.1.3 Metriche RAG statiche vs. Metriche Didattiche Dinamico-Adattive

Dall'analisi comparativa tra la suite di indicatori convenzionali integrati nel framework (descritti nella Sezione 3.6) e i requisiti dell'interazione tutoriale emerge una discrepanza metodologica fondamentale:

1. **La natura statica delle metriche RAG standard:** indicatori quali `Faithfulness`, `Answer Relevance`, `ROUGE` o `BERTScore` operano secondo un paradigma invariante: a parità di risposta y e contesto documentale $\mathcal{Z}$, la funzione di costo restituisce un valore identico, astraendo totalmente dal ruolo conversazionale del modello e dagli obiettivi cognitivi del discente. In un contesto didattico, tale staticità produce gravi distorsioni: una risposta che fornisce la derivata completa o la soluzione algebrica a fronte di una svista dello studente massimizza la pertinenza e la fedeltà documentale, ma distrugge il processo di apprendimento maieutico (*spoiler effect*).

2. **Il paradigma Dinamico-Adattivo del PES:** Il Pedagogical Evaluation Score formalizzato in questo capitolo supera questo limite non rifiutando le metriche convenzionali, ma integrandole gerarchicamente. Il PES assorbe la fedeltà documentale trasformandola in un vincolo di sicurezza non compensativo (Livello 0), e introduce al contempo la modulazione adattiva dei pesi vettoriali $\mathbf{w}(\mathcal{S})$ condizionata allo stato pedagogico (Livello 1).

La Tabella 4.1 riassume la complementarità operativa tra le metriche di valutazione convenzionali e il PES Dinamico-Adattivo.

4.2 Architettura gerarchica a tre livelli del PES

4.2.1 Pipeline computazionale e flusso informativo

L'infrastruttura di calcolo del PES organizza la stima della qualità tutoriale secondo una sequenza a tre livelli operativi coordinati, culminanti in uno stadio di aggregazione finale, illustrata analiticamente nel diagramma TikZ di Figura 4.1:

- **Livello 0 — Faithfulness Safety Gate:** filtro di sicurezza iniziale sulla fedeltà documentale. Valuta la conformità logica della risposta generata y rispetto all'insieme dei frammenti di contesto recuperati $\mathcal{Z}$. Qualora l'indice di aderenza scenda al di sotto della soglia critica τ, la computazione viene interrotta e il punteggio complessivo forzato a zero;

Tabella 4.1: Confronto dimensionale tra metriche RAG tradizionali e il PES Dinamico-Adattivo.

Dimensione	**Metriche Standard**	**PES Dinamico-Adattivo**	**Razionale Didattico**
Ponderazione Pesi	Statica / Uniforme	**Dinamica e Adattiva** ($\mathbf{w}(\mathcal{S})$)	Calibra la severità maieutica sullo stato contingente ($\mathcal{S}_A, \mathcal{S}_B, \mathcal{S}_C$).
Trattamento Spoiler	Neutro o Premiato	**Penalizzato** ($M_3 \to 0$)	Penalizza l'anticipazione non richiesta della soluzione corretta.
Uptake dell'Errore	Premiato per pertinenza	**Azzerato** ($w_1 = 0$ in $\mathcal{S}_B, \mathcal{S}_C$)	Evita che il tutor amplifichi e confermi i fraintendimenti dell'allievo.
Allucinazione Fattuale	Degrado lineare	**Gate di blocco** (PES = 0)	Barriera epistemica assoluta contro disinformazione didattica.
Costo Computazionale	Zero-cost o Medio	Composito parallelo	Ottimizzato via caching SHA-256 e micro-LLM locali.

- **Livello 1 — Pedagogical State Intent Classifier:** modulo di classificazione contestuale che analizza congiuntamente la richiesta dello studente q e la risposta del tutor y per identificare la fase didattica dello scambio, associandola a uno stato pedagogico discreto $\mathcal{S} \in \{\mathcal{S}_A, \mathcal{S}_B, \mathcal{S}_C\}$ (insegnamento concettuale, correzione dell'errore o valutazione socratica). Questa classificazione assegna dinamicamente il vettore di ponderazione $\mathbf{w}(\mathcal{S}) = \Big(w_1(\mathcal{S}), w_2(\mathcal{S}), w_3(\mathcal{S})\Big)$, calibrando l'importanza di ciascuna sotto-metrica in funzione degli specifici obiettivi educativi del turno;
- **Livello 2 — Computazione Parallela delle Sotto-metriche:** stima modulare e disaccoppiata delle tre componenti primarie della qualità didattica:
  - M_1 (*Conversational & Context Elaboration / Uptake*);
  - M_2 (*Linguistic Adaptation*);
  - M_3 (*Scaffolding & No Immediate Disclosure*);
- **Livello 3 — Aggregazione Geometrica Ponderata e Lisciamento:** sintesi non lineare delle sotto-metriche mediante una media geometrica pesata, integrata con un fattore di regolarizzazione (ϵ-smoothing). Rispetto a una media aritmetica, questa formulazione garantisce la proprietà di *non-compensatorietà*: se anche una sola dimensione pedagogica rilevante risulta fallimentare, l'intero punteggio collassa verso lo zero, impedendo che gravi carenze didattiche vengano mascherate da altri aspetti positivi. Il parametro ϵ assicura al contempo la stabilità computazionale, prevenendo singolarità logaritmiche quando un indicatore si annulla.

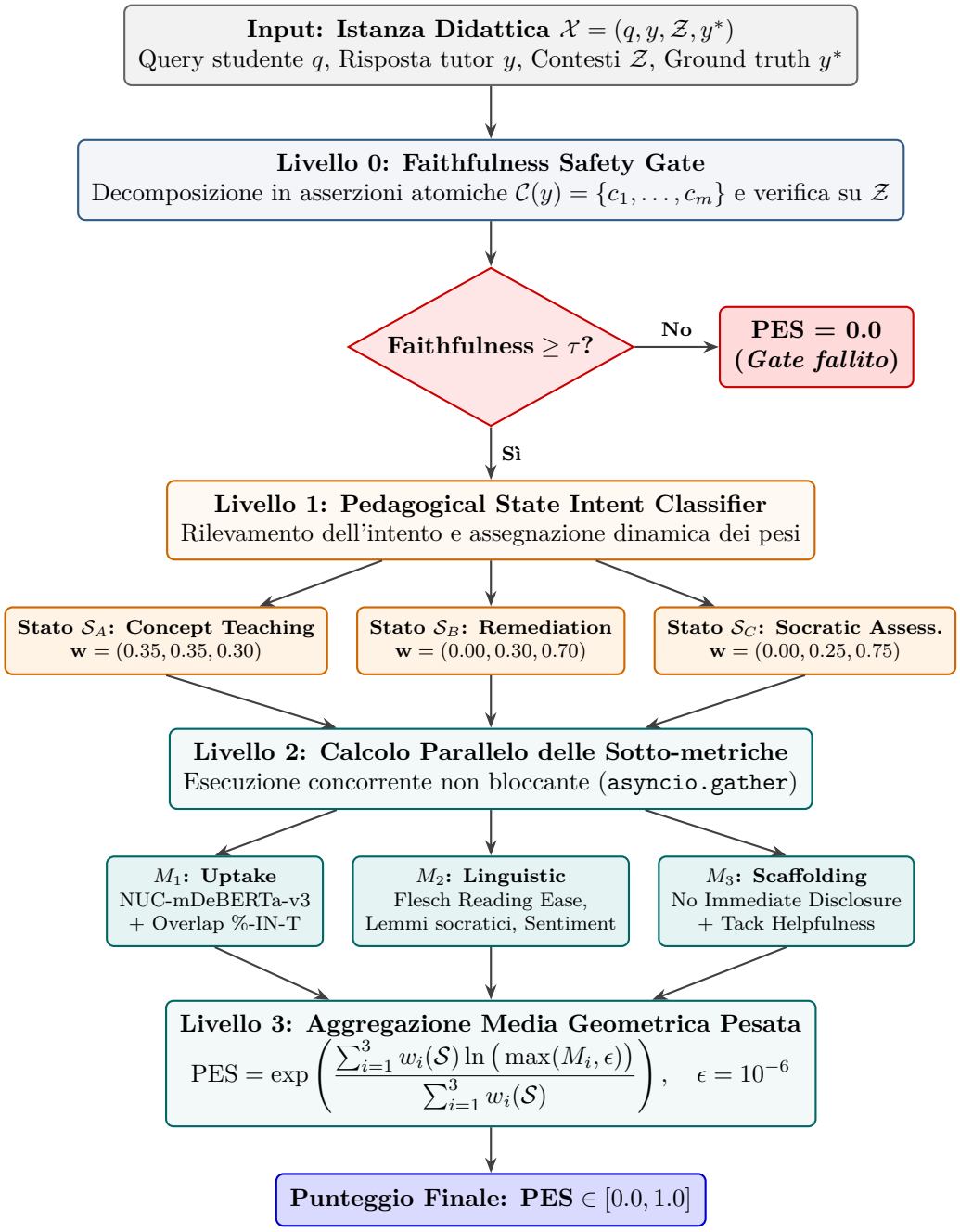


Figura 4.1: Diagramma architetturale completo della metrica composita Pedagogical Evaluation Score (PES), con evidenziazione dei tre livelli operativi, del gate di sicurezza documentale e dell'aggregazione non lineare.

4.2.2 Algoritmo di valutazione asincrona parallela

Sul piano implementativo, la valutazione del PES è orchestrata in Python sfruttando le coroutine asincrone native di `asyncio`. L'esecuzione delle sotto-metriche al Livello 2 avviene in parallelo mediante la primitiva non bloccante `asyncio.gather`, riducendo la latenza complessiva dell'operazione al limite superiore imposto dal componente più oneroso:

$$T_{\text{totale}} \approx T_{\text{gate}} + T_{\text{state}} + \max\left(T_{M1}, T_{M2}, T_{M3}\right) + T_{\text{agg}} \tag{4.1}$$

Il formalismo operativo è specificato in dettaglio nell'Algoritmo 1.

Algorithm 1 Computazione asincrona del Pedagogical Evaluation Score (PES)

Require: Campione didattico $\mathcal{X} = (q, y, \mathcal{Z}, y^*)$; soglia faithfulness τ; lisciamento $\epsilon = 10^{-6}$; mappatura pesi $\{\mathcal{S} \mapsto \mathbf{w}\}$.
Ensure: Risultato strutturato $\mathcal{R}_{\text{PES}} = \langle \text{score}, \text{reason}, \mathcal{M} \rangle$.

Inizializza dizionario metadati $\mathcal{M} \leftarrow \emptyset$
if $y = \emptyset \vee \text{length}(y) = 0$ **then**
    return $\langle 0.0, \text{'Risposta assente'}, \mathcal{M} \rangle$
end if
▷ — **Livello 0: Faithfulness Gate** —
$s_{\text{faith}} \leftarrow \textsc{EvaluateFaithfulness}(y, \mathcal{Z})$
$\mathcal{M}[\texttt{faithfulness}] \leftarrow s_{\text{faith}}$
if $s_{\text{faith}} < \tau$ **then**
    $\mathcal{M}[\texttt{gate_failed}] \leftarrow$ **true**
    return $\langle 0.0, \text{'Gate fallito: allucinazione rilevata'}, \mathcal{M} \rangle$
end if
$\mathcal{M}[\texttt{gate_failed}] \leftarrow$ **false**
▷ — **Livello 1: Intent State Classification** —
$\mathcal{S}, \text{conf} \leftarrow \textsc{ClassifyPedagogicalState}(q, y)$
$(w_1, w_2, w_3) \leftarrow \textsc{ResolveWeights}(\mathcal{S})$
$\mathcal{M}[\texttt{state}] \leftarrow \langle \mathcal{S}, \text{conf}, (w_1, w_2, w_3) \rangle$
▷ — **Livello 2: Computazione Parallela Sotto-Metriche (asyncio.gather)** —
$r_1 \leftarrow \textsc{ComputeUptake}(q, y, \mathcal{Z})$ ▷ Sotto-metrica M_1
$r_2 \leftarrow \textsc{ComputeLinguistic}(y)$ ▷ Sotto-metrica M_2
$r_3 \leftarrow \textsc{ComputeScaffolding}(q, y)$ ▷ Sotto-metrica M_3
$M_1 \leftarrow r_1.\text{score}, \quad M_2 \leftarrow r_2.\text{score}, \quad M_3 \leftarrow r_3.\text{score}$
$\mathcal{M}[M_1] \leftarrow r_1, \quad \mathcal{M}[M_2] \leftarrow r_2, \quad \mathcal{M}[M_3] \leftarrow r_3$
▷ — **Livello 3: Aggregazione Geometrica Lisciata** —
$W \leftarrow w_1 + w_2 + w_3$
$\lambda \leftarrow \frac{1}{W}\Big(w_1 \ln(\max(M_1, \epsilon)) + w_2 \ln(\max(M_2, \epsilon)) + w_3 \ln(\max(M_3, \epsilon))\Big)$
$\text{score} \leftarrow \min\Big(1.0, \max\Big(0.0, \exp(\lambda)\Big)\Big)$
Formatta stringa diagnostica reason
return $\langle \text{round}(\text{score}, 4), \text{reason}, \mathcal{M} \rangle$

4.3 Livello 0 — Faithfulness Safety Gate

4.3.1 Motivazione del vincolo non compensativo sulla fedeltà

La maggior parte delle metriche composite in letteratura adotta schemi di aggregazione lineari o additivi (come medie aritmetiche ponderate), nei quali un deficit marcato su una dimensione può essere compensato da prestazioni eccellenti su un'altra. In ambito educativo, tale compensazione non è ammissibile: una spiegazione affetta da errori concettuali o allucinazioni rispetto ai testi di riferimento è **altamente fuorviante e dannosa**. Uno studente alle prese con nozioni complesse non possiede ancora gli schemi concettuali necessari per riconoscere un'inesattezza sottile mascherata da un tono empatico e sicuro [17].

Pertanto, il PES introduce un vincolo a **priorità assoluta**: la fedeltà concettuale ai documenti è un prerequisito inderogabile. Se il tutor genera allucinazioni, il punteggio complessivo dell'intervento scende istantaneamente a zero, indipendentemente dall'eleganza stilistica o dalla gradualità pedagogica mostrata.

4.3.2 Decomposizione in asserzioni atomiche e verifica logica

Operativamente, la metrica di *Faithfulness* estrae dal testo della replica y l'insieme delle asserzioni atomiche, autoconsistenti e verificabili:

$$\mathcal{C}(y) = \{c_1, c_2, \ldots, c_m\} \tag{4.2}$$

Ciascun claim c_i viene confrontato semanticamente contro i passaggi di contesto documentale $\mathcal{Z} = \{z_1, z_2, \ldots, z_k\}$ attraverso un LLM-as-a-Judge vincolato da uno schema di output JSON rigoroso:

$$\mathbb{I}\Big(\text{Support}(c_i, \mathcal{Z})\Big) = \begin{cases} 1 & \text{se } \exists\, z \subseteq \mathcal{Z} \text{ tale che } z \models c_i \\ 0 & \text{altrimenti (asserzione allucinata o non comprovata)} \end{cases} \tag{4.3}$$

La misura scalare di fedeltà è formalizzata dal rapporto:

$$\text{Faithfulness}(y, \mathcal{Z}) = \begin{cases} 1.0 & \text{se } |\mathcal{C}(y)| = 0 \\ \dfrac{1}{|\mathcal{C}(y)|} \displaystyle\sum_{i=1}^{|\mathcal{C}(y)|} \mathbb{I}\Big(\text{Support}(c_i, \mathcal{Z})\Big) & \text{se } |\mathcal{C}(y)| > 0 \end{cases} \tag{4.4}$$

La condizione $|\mathcal{C}(y)| = 0$ gestisce costrutti puramente socratici o frasi di raccordo comunicativo (ad esempio, *«Cosa accadrebbe se provassi a isolare l'incognita a primo membro?»*): non enunciando asserzioni fattuali da verificare, a questi interventi viene assegnata per definizione la piena conformità (1.0).

4.3.3 Soglia di arresto e configurazione del gate

La funzione di gating condizionale è formalizzata come segue:

$$\text{Gate}(y, \mathcal{Z}; \tau) = \mathbb{I}\Big(\text{Faithfulness}(y, \mathcal{Z}) \geq \tau\Big) \tag{4.5}$$

$$\text{Gate}(y, \mathcal{Z}; \tau) = 0 \implies \text{PES}(y, q, \mathcal{Z}) = 0.0 \tag{4.6}$$

La scelta del parametro τ è cruciale:

- **Soglia nominale standard ($\tau = 0.50$):** impone che la maggioranza assoluta dei claim enunciati trovi diretto riscontro nelle fonti;
- **Soglia permissiva calibrata ($\tau = 0.25$):** adottata nella sperimentazione intensiva su modelli compatti (es. Ollama Llama 3.2) per mitigare i falsi positivi generati da una parafrasi lessicalmente divergente ma didatticamente corretta rispetto al contesto.

4.4 Livello 1 — Classificazione dello Stato Pedagogico e Ponderazione Dinamico-Adattiva

4.4.1 Tassonomia degli stati pedagogici e il paradigma dinamico-adattivo

L'elemento di innovazione metodologica introdotto dalla presente tesi risiede nella formulazione del **PES Dinamico-Adattivo**. Mentre le tradizionali metriche composite adottano pesi statici e universali — presupponendo implicitamente che i criteri di qualità siano invarianti rispetto alla fase del dialogo —, il PES riconosce che l'efficacia didattica di un'interazione tutoriale è strettamente contingente: pretendere che il tutor ponga quesiti maieutici quando lo studente esplicita di non conoscere affatto una definizione introduttiva genera frustrazione; per converso, offrire spiegazioni estese quando l'allievo ha commesso una mera svista di calcolo impedisce lo sviluppo dell'auto-diagnosi.

Per formalizzare matematicamente questa flessibilità maieutica, il Livello 1 classifica ciascun turno didattico in uno di tre stati discreti mutuamente esclusivi $\mathcal{S} \in \{\mathcal{S}_A, \mathcal{S}_B, \mathcal{S}_C\}$, riparametrizzando dinamicamente il vettore dei pesi $\mathbf{w}(\mathcal{S}) = \Big(w_1(\mathcal{S}), w_2(\mathcal{S}), w_3(\mathcal{S})\Big)$:

Stato $\mathcal{S}_A$ — Concept Teaching (Insegnamento Concettuale): Si attiva quando la conversazione verte sull'acquisizione di nuove nozioni, principi teorici o quadri terminologici. In questo scenario, la funzione primaria del tutor consiste nell'esporre con rigore, chiarezza e accessibilità i contenuti. Il vettore dei pesi assegna un'importanza bilanciata all'elaborazione concettuale (M_1) e all'adeguatezza espositiva (M_2), riducendo l'enfasi sulla non-rivelazione (M_3):

$$\mathbf{w}(\mathcal{S}_A) = \Big(w_1 = 0.35,\, w_2 = 0.35,\, w_3 = 0.30\Big) \tag{4.7}$$

> **Studente q:** «Che cos'è una matrice invertibile e qual è la condizione necessaria e sufficiente affinché l'inversa esista?»
>
> **Tutor y:** «Una matrice quadrata $A \in \mathbb{R}^{n \times n}$ si dice invertibile se esiste una matrice B tale che $AB = BA = I_n$, dove I_n è la matrice identità. La condizione necessaria e sufficiente è che il determinante sia non nullo, ovvero $\det(A) \neq 0$, il che equivale ad affermare che le sue righe e colonne costituiscano un insieme di vettori linearmente indipendenti.»

Stato $\mathcal{S}_B$ — Error Remediation (Correzione dell'Errore): Subentra allorché l'allievo formula un passaggio errato, una congettura fallace o manifesta un fraintendimento concettuale esplicito. L'obiettivo didattico risiede nella guida verso l'auto-correzione autonoma. È imperativo evitare lo svelamento diretto della soluzione corretta (M_3), mantenendo un registro accogliente e non giudicante (M_2). La ripresa discorsiva dell'errore (M_1) viene soppressa per scongiurare che il sistema insista sterilmente sulla fallacia dell'allievo:

$$\mathbf{w}(\mathcal{S}_B) = \Big(w_1 = 0.00,\, w_2 = 0.30,\, w_3 = 0.70\Big) \tag{4.8}$$

> **Studente** q**:** «Nell'equazione $2x + 6 = 14$, ho sottratto 6 e ho ottenuto $2x = 8$. Poi ho sottratto 2 da entrambi i lati e trovo $x = 6$. È giusto?»
>
> **Tutor** y**:** «Molto bene il primo passaggio: hai impostato correttamente $2x = 8$. Nel passaggio successivo, però, osserva come il coefficiente 2 sia legato alla x da un prodotto. Qual è l'operazione inversa della moltiplicazione che ti consentirebbe di isolare l'incognita?»

Stato $\mathcal{S}_C$ — Socratic Assessment (Valutazione Socratica): Si applica quando l'input dello studente è corretto, neutrale o interlocutorio. Il tutor assume la veste maieutica: saggia la tenuta della comprensione mediante domande di riflessione metacognitiva, estensioni concettuali o controesempi controllati. Lo scaffolding (M_3) assurge a cardine dominante assoluto:

$$\mathbf{w}(\mathcal{S}_C) = \Big(w_1 = 0.00,\, w_2 = 0.25,\, w_3 = 0.75\Big) \tag{4.9}$$

> **Studente** q**:** «Ho calcolato la derivata di $f(x) = x^2e^x$ applicando la regola del prodotto e ho ottenuto $f'(x) = 2xe^x + x^2e^x$.»
>
> **Tutor** y**:** «Esatto, la derivata è corretta! Ora prova a fare un passo in avanti: raccogliendo a fattor comune i termini simili, in quali punti la derivata prima si annulla? Che cosa suggerisce questa informazione sull'andamento della funzione?»

Stato di Fallback: Utilizzato qualora il classificatore non raggiunga un margine di confidenza accettabile:

$$\mathbf{w}(\text{fallback}) = \Big(w_1 = 0.33,\, w_2 = 0.33,\, w_3 = 0.34\Big) \tag{4.10}$$

La sintesi strutturale dei profili di pesatura è compendiata nella Tabella 4.2.

Tabella 4.2: Tassonomia degli stati pedagogici, definizioni didattiche e profili di pesatura adattiva.

Stato	Denominazione	Caratterizzazione Didattica	w_1	w_2	w_3
$\mathcal{S}_A$	**Concept Teaching**	Esposizione teorica, introduzione definizioni, esemplificazione di quadri concettuali generali.	**0.35**	**0.35**	**0.30**
$\mathcal{S}_B$	**Error Remediation**	Rilevazione svista concettuale o procedurale; indirizzamento maieutico all'auto-correzione.	**0.00**	**0.30**	**0.70**
$\mathcal{S}_C$	**Socratic Assessment**	Input studente corretto/neutro; stimolo metacognitivo, posa di controesempi e quesiti guida.	**0.00**	**0.25**	**0.75**
—	*Fallback / Default*	Configurazione uniforme per fallita convergenza o ambiguità di intento.	0.33	0.33	0.34

Nota: w_1: Uptake (M_1), w_2: Linguistic Adaptation (M_2), w_3: Scaffolding / NID (M_3).

4.4.2 Motivazione dell'azzeramento di w_1 negli Stati B e C

La scelta architetturale di fissare $w_1(\mathcal{S}_B) = 0.00$ e $w_1(\mathcal{S}_C) = 0.00$ scaturisce da una precisa evidenza teorica ed empirica: nelle fasi di correzione di un errore o di stimolazione socratica, incoraggiare il modello linguistico a massimizzare l'uptake discorsivo (M_1) comporta che il tutor ripeta o rielabori l'argomentazione (anche fallace) appena enunciata dallo studente. Tale ripetizione sovraccarica la memoria di lavoro dell'allievo e devia l'attenzione dal focus risolutivo. Come confermato sperimentalmente dai risultati dello studio di ablazione condotto nel Capitolo 5 (Sezione 5.4), l'inclusione di $w_1 > 0$ negli stati $\mathcal{S}_B$ e $\mathcal{S}_C$ induce una correlazione negativa con la qualità percepita dagli educatori umani.

4.5 Livello 2 — Tassonomia delle sotto-metriche

4.5.1 M_1 — Uptake

In una conversazione didattica reale, un buon insegnante non parla mai nel vuoto e non recita un copione predefinito. Al contrario, pratica un ascolto attivo e attento: coglie le parole esatte dello studente, ne intercetta i dubbi e le intuizioni parziali, e costruisce la propria spiegazione ripartendo proprio da lì. In linguistica e nelle scienze dell'educazione, questo fenomeno prende il nome di *conversational uptake* [71]: è il segnale tangibile con cui il tutor dimostra all'allievo che il suo intervento è stato ascoltato, compreso e valorizzato, trasformando lo scambio in un dialogo autentico anziché in un monologo unidirezionale.

Trasferire questo principio nella valutazione di un assistente artificiale significa verificare se il modello linguistico sappia ricollegarsi in modo fluido a quanto espresso dallo studente, oppure se tenda a ignorarne il filo logico per produrre risposte generiche o decontestualizzate. L'indicatore M_1 quantifica esattamente questo grado di coesione dialogica, formalizzando il legame tra la replica del tutor, l'input dell'allievo e i documenti didattici di supporto.

Dal punto di vista operativo, la valutazione dell'uptake deve saper distinguere due situazioni profondamente diverse:

1. **Input dello studente articolato ($|T_q| \geq 5$ token):** quando l'allievo esprime un pensiero compiuto, un passaggio di calcolo o un dubbio specifico, il tutor ha il dovere pedagogico di agganciarsi direttamente alle sue parole. In questo caso si calcola l'*uptake conversazionale* (U_{conv}), che combina la coerenza logico-semantica della risposta con la continuità lessicale rispetto ai termini chiave usati dal discente.

2. **Input dello studente breve o interlocutorio ($|T_q| < 5$ token):** se l'allievo si limita a risposte telegrafiche (come *«Sì»*, *«Non capisco»* o *«E adesso?»*), pretendere che il tutor ne riprenda le parole sarebbe privo di senso e penalizzante. In queste circostanze, l'attenzione della metrica si sposta sull'*uptake contestuale* (U_{ctx}), misurando quanto la risposta del tutor si ancora saldamente alle nozioni presenti nei frammenti documentali di riferimento ($\mathcal{Z}$).

Per modellare computazionalmente questa duplice forma di connessione, il PES adotta il paradigma del *Next Utterance Classification* (NUC) teorizzato da Demszky et al. [14], implementandolo tramite il trasformatore multilingue pre-addestrato `mDeBERTa-v3-base-mnli-xnli` [41]. Il modello valuta se la replica t rappresenti la naturale prosecuzione o conseguenza logica (*entailment*) dell'enunciato sorgente s, calcolando la probabilità:

$$P_{\text{NUC}}(s, t) = \text{Softmax}\Big(\mathbf{W}_{\text{NUC}} \cdot \text{Encoder}(s \parallel t)[\text{CLS}]\Big)_{\text{entailment}} \tag{4.11}$$

La computazione complessiva di M_1 si articola quindi secondo la seguente logica condizionale, che discrimina in base all'ampiezza lessicale della query dello studente

($|T_q|$):

$$M_1(q, y, \mathcal{Z}) = \begin{cases} U_{\text{conv}}(q, y) & \text{se } |T_q| \geq 5 \\ U_{\text{ctx}}(\mathcal{Z}, y) & \text{se } |T_q| < 5 \wedge \mathcal{Z} \neq \emptyset \\ 0.0 & \text{altrimenti} \end{cases} \tag{4.12}$$

dove le due componenti U_{conv} e U_{ctx} sono definite formalmente come:

$$U_{\text{conv}}(q, y) = \alpha \cdot P_{\text{NUC}}(q, y) + (1 - \alpha) \cdot \text{Overlap}_{\%\text{-IN-T}}(q, y) \tag{4.13}$$

$$U_{\text{ctx}}(\mathcal{Z}, y) = \max_{z \in \mathcal{Z}} P_{\text{NUC}}(z, y) \tag{4.14}$$

con $\alpha = 0.70$. Il termine $\text{Overlap}_{\%\text{-IN-T}}$ rappresenta la percentuale di parole a contenuto informativo (esclusi gli elementi grammaticali e le *stop-words*) condivise tra la richiesta dello studente e la replica del tutor, garantendo così sia la pertinenza concettuale sia il riconoscimento esplicito del lessico impiegato dall'allievo.

4.5.2 M_2 — Linguistic Adaptation

Nell'insegnamento, il *modo* in cui si comunica un concetto è tanto determinante quanto il contenuto del concetto stesso. Una spiegazione scientificamente ineccepibile può rivelarsi del tutto inefficace se formulata con un linguaggio inutilmente oscuro, se espressa con un tono dogmatico che scoraggia le domande, oppure se corregge un errore in maniera rigida e demoralizzante. In un dialogo tutoriale, la lingua non è un semplice veicolo di dati, ma uno strumento pedagogico che deve accogliere, motivare e guidare.

La sotto-metrica M_2 (*Linguistic Adaptation*) valuta proprio questa sensibilità comunicativa. Un buon tutor artificiale non deve limitarsi a produrre frasi corrette, ma deve saper modulare il proprio registro stilistico lungo tre dimensioni fondamentali:

1. **Semplicità e accessibilità sintattica (S_{read}):** il linguaggio deve risultare fluido e immediatamente comprensibile, evitando periodi eccessivamente involuti o tecnicismi non necessari che finirebbero per sovraccaricare la memoria di lavoro dell'allievo;

2. **Apertura socratica e spirito d'indagine (S_{tent}):** l'insegnamento più efficace non cala la verità dall'alto come un dogma. Attraverso l'uso di formulazioni ipotetiche e forme verbali modali (ad esempio *«Proviamo a considerare...»*, *«Cosa succederebbe se...»*, *«Potrebbe essere che...»*), il tutor invita lo studente a congetturare, esplorare e sentirsi protagonista della scoperta;

3. **Tono propositivo e critica costruttiva (S_{crit}):** di fronte a un errore, il riscontro deve essere rassicurante e orientato al futuro. L'obiettivo è stimolare una mentalità di crescita (*growth mindset*) [49], dove lo sbaglio non è motivo di frustrazione ma un trampolino per il ragionamento.

Questi tre requisiti vengono sintetizzati nell'indicatore M_2 mediante una combinazione convessa che bilancia leggibilità, maieutica e incoraggiamento:

$$M_2(y) = 0.40 \cdot S_{\text{read}}(y) + 0.30 \cdot S_{\text{tent}}(y) + 0.30 \cdot S_{\text{crit}}(y) \tag{4.15}$$

Nei paragrafi seguenti viene dettagliata la modellazione analitica di ciascuna delle tre componenti.

1. Leggibilità Sintattico-Strutturale ($S_{\mathbf{read}}$)

Per stimare l'accessibilità del testo si impiega l'indice *Flesch Reading Ease* (FRE) [48], un classico della psicolinguistica che quantifica la facilità di lettura correlando la lunghezza media delle frasi con la complessità sillabica dei singoli vocaboli. Il punteggio, calcolato mediante la libreria multilingue `textstat`, viene normalizzato linearmente sull'intervallo unitario $[0.0, 1.0]$:

$$S_{\text{read}}(y) = \min\left(1.0,\ \max\left(0.0,\ \frac{\text{FRE}(y)}{100.0}\right)\right) \tag{4.16}$$

In tal modo, risposte scorrevoli e dal periodare snello ottengono valori prossimi a 1.0, mentre testi involuti e artificiosi vengono opportunamente penalizzati.

2. Tentativeness e Apertura Socratica ($S_{\mathbf{tent}}$)

Questa componente premia la capacità del tutor di esprimersi in forma non categorica ed esplorativa. Dal punto di vista computazionale, il testo generato y viene analizzato con il lemmatizzatore multilingue `simplemma`, previa identificazione della lingua tramite `langdetect`. Si calcola la frequenza con cui compaiono lemmi socratici ed espressioni dubitative (T_{tent}) rispetto al totale dei token ($|T_y|$), rapportando il risultato a una soglia di saturazione didattica ottimale fissata al 5% ($\theta_{\text{sat}} = 0.05$):

$$S_{\text{tent}}(y) = \min\left(1.0,\ \frac{|T_{\text{tent}}(y)|}{|T_y| \cdot \theta_{\text{sat}}}\right) \tag{4.17}$$

Il dizionario di riferimento include verbi modali, congiunzioni ipotetiche e avverbi di possibilismo: in lingua italiana, ad esempio, comprende termini quali *forse, potrebbe, ipotizziamo, consideriamo, parrebbe*; in lingua inglese, forme come *perhaps, might, could, suppose, consider*. La saturazione a θ_{sat} impedisce che un uso eccessivo o artificiale di dubitativi gonfi indebitamente il punteggio.

3. Critica Costruttiva e Tono Propositivo ($S_{\mathbf{crit}}$)

La terza dimensione valuta se il tutor sappia formulare osservazioni benevole, capaci di correggere senza mortificare. Per catturare questa sfumatura affettiva e pragmatica, si impiega la rete neurale multilingue pre-addestrata `nlptown/bert-base-multilingual-uncased-sentiment`, che classifica il testo su una scala da 1 a 5 stelle, proiettata linearmente in $s_{\text{bert}} \in [0.0, 1.0]$. A questa stima di base viene aggiunto un bonus di $+0.10$ qualora il discorso contenga connettivi avversativi propositivi (quali *tuttavia, benché, nondimeno, pur tuttavia*):

$$S_{\text{crit}}(y) = \min\left(1.0,\ s_{\text{bert}}(y) + 0.10 \cdot \mathbb{I}(\text{ConnettivoAvversativo} \in y)\right) \tag{4.18}$$

La presenza di tali connettivi rivela una specifica attitudine tutoriale: riconoscere ciò che di buono c'è nel tentativo dell'allievo prima di indirizzarlo con delicatezza verso la corretta traiettoria di pensiero.

4.5.3 M_3 — Scaffolding, NID & Helpfulness

La sotto-metrica M_3 tocca il nodo più critico e delicato dell'intero processo educativo mediato da modelli linguistici: il fragile equilibrio tra il sostenere l'allievo e il lasciarlo pensare da solo.

I moderni modelli di intelligenza artificiale sono addestrati per essere assistenti massimamente servizievoli ed efficienti: se interrogati su un problema, la loro risposta naturale è fornire immediatamente la soluzione completa, corretta e rifinita nei minimi particolari. In classe o in una sessione di studio, però, questo comportamento genera quello che viene identificato come *Spoiler Effect*: consegnare subito la risposta pronta cancella lo sforzo cognitivo autonomo dell'allievo (*productive struggle*) e vanifica il principio dello scaffolding all'interno della Zona di Sviluppo Prossimale [11, 12]. Senza sforzo e senza il superamento graduale delle difficoltà, l'apprendimento rischia di rimanere superficiale ed effimero.

Al tempo stesso, un assistente che si limitasse a rifiutare le risposte o a porre domande oscure e sconclusionate non farebbe pedagogia, ma genererebbe soltanto smarrimento e frustrazione. Il ruolo del tutor ideale risiede dunque in un compromesso virtuoso tra due esigenze:

1. **Non rivelare prematuramente la soluzione (*No Immediate Disclosure — NID*):** resistere all'impulso di dare la risposta finale, limitandosi a offrire indizi, chiarimenti concettuali o domande guida proporzionati al livello di difficoltà incontrato dall'allievo;

2. **Fornire un supporto realmente utile ed efficace (*Helpfulness*):** assicurarsi che l'indizio o la domanda offerta non siano fini a se stessi, ma forniscano allo studente una leva concreta e azionabile per sbloccare il proprio ragionamento e compiere il passo successivo in autonomia.

Per riflettere questa duplice esigenza, la sotto-metrica M_3 combina con pari dignità la disciplina della non-rivelazione con l'efficacia didattica dell'intervento:

$$M_3(q, y) = 0.50 \cdot S_{\text{NID}}(q, y) + 0.50 \cdot S_{\text{help}}(q, y) \tag{4.19}$$

1. No Immediate Disclosure (S_{NID})

Il punteggio S_{NID} misura la capacità del tutor di trattenersi dallo svelamento diretto della soluzione. La valutazione è demandata a un componente *LLM-as-a-Judge*, istruito con un prompt ad alta specificità pedagogica che assegna alla risposta uno di quattro livelli operativi discreti, formalizzati nella Tabella 4.3.

Tabella 4.3: Rubrica formale di attribuzione del punteggio No Immediate Disclosure (S_{NID}).

S_{NID}	**Livello Operativo**	**Comportamento Tutoriale Osservato**
0.00	**Full Disclosure / Spoiler**	Il tutor esplicita direttamente il valore numerico finale, la formula di risoluzione o il frammento di codice conclusivo. Disinnesca completamente il carico cognitivo.
0.33	**Major Leakage**	Il tutor risolve autonomamente il passaggio algebrico o concettuale più critico e impegnativo dell'esercizio, lasciando all'allievo solo computazioni banali.
0.67	**Guided Hinting**	Il tutor delinea la sequenza dei passi concettuali necessari (*solution path*) o suggerisce quale teorema invocare, senza eseguire calcoli né anticipare il risultato.
1.00	**Pure Scaffolding / Socratic**	Il tutor non rivela alcun dato risolutivo; stimola l'allievo esclusivamente con domande di riflessione metacognitiva mirate a far emergere la contraddizione.

Come emerge dalla rubrica, il punteggio pieno (1.00) premia esclusivamente gli interventi autenticamente socratici che lasciano all'allievo l'intera responsabilità risolutiva; al contrario, lo svelamento della soluzione finale porta a un azzeramento istantaneo (0.00) dell'indicatore.

2. Helpfulness Pedagogica (S_{help})

La sola astensione dallo svelamento della soluzione non basta a garantire una buona didattica: un silenzio ostinato o un suggerimento incomprensibile otterrebbero il massimo del punteggio NID pur essendo pedagogicamente disastrosi. Per questa ragione, il punteggio S_{help} valuta l'effettiva capacità della risposta di chiarire il dubbio e stimolare la prosecuzione del lavoro.

Adottando il protocollo psicometrico validato da Tack e Piech [35], un LLM-as-a-Judge qualifica l'utilità didattica su una scala Likert a 5 punti (da 1, «totalmente inutile o fuorviante», a 5, «pedagogicamente eccellente e risolutivo per l'impasse»), normalizzata linearmente sull'intervallo $[0.0, 1.0]$:

$$S_{\text{help}}(q, y) = \frac{\text{Score}_{\text{Likert}}(q, y) - 1.0}{4.0} \tag{4.20}$$

In questo modo, risposte che guidano efficacemente lo studente verso la comprensione senza sostituirsi a lui ottengono valutazioni elevate in entrambi i fattori, massimizzando M_3.

4.6 Formulazione matematica unificata e proprietà analitiche

4.6.1 Equazione maestra con lisciamento ϵ-smoothing

Unificando il filtro booleano del Livello 0, la configurazione dei pesi dipendente dallo stato didattico del Livello 1 e le tre sotto-metriche del Livello 2, l'equazione analitica definitiva del Pedagogical Evaluation Score è formalizzata come:

$$\mathrm{PES}(y, q, \mathcal{Z}) = \begin{cases} 0.0 & \text{se Faithfulness}(y, \mathcal{Z}) < \tau \\ \exp\left(\dfrac{\sum_{i=1}^{3} w_i(\mathcal{S}) \cdot \ln\left(\max(M_i, \epsilon)\right)}{\sum_{i=1}^{3} w_i(\mathcal{S})} \right) & \text{se Faithfulness}(y, \mathcal{Z}) \geq \tau \end{cases} \tag{4.21}$$

dove $\epsilon = 10^{-6}$ definisce la soglia inferiore di lisciamento computazionale introdotta per garantire la differenziabilità numerica e prevenire singolarità asintotiche ($\ln(0) \to -\infty$) nel logaritmo.

4.6.2 Proprietà matematiche: non-compensatorietà e confronto tra medie

La scelta della media geometrica pesata risponde a precisi requisiti assiomatici di teoria della misurazione didattica:

1. **Proprietà di Non-Compensatorietà:** A differenza della media aritmetica, nella quale una prestazione insufficiente ($M_k \to 0$) può essere mascherata da punteggi eccellenti negli altri indicatori, la media geometrica vincola il risultato globale al prodotto pesato delle componenti:

$$\lim_{M_k \to 0} \mathrm{PES} = \epsilon^{\frac{w_k}{\sum w_i}} \approx 0.0 \quad (\text{se } w_k > 0) \tag{4.22}$$

Esempio di fallimento della media aritmetica: Si consideri un tutor artificiale operante in Stato $\mathcal{S}_C$ (Socratic Assessment, con pesi $w_2 = 0.25, w_3 = 0.75$) che eroghi una risposta stilisticamente impeccabile ($M_2 = 0.95$), ma contravvenga totalmente al vincolo di scaffolding fornendo lo spoiler integrale della soluzione ($M_3 = 0.01$). Sotto media aritmetica pesata, l'intervento riceverebbe una valutazione ampiamente sufficiente:

$$\mathrm{PES}_{\text{aritmetica}} = 0.25 \cdot 0.95 + 0.75 \cdot 0.01 = 0.245$$

mentre per il PES geometrico il punteggio collassa inesorabilmente a:

$$\begin{aligned} \mathrm{PES}_{\text{geometrico}} &= \exp\Big(0.25 \ln(0.95) + 0.75 \ln(0.01)\Big) \\ &= 0.95^{0.25} \cdot 0.01^{0.75} \approx 0.987 \cdot 0.0316 \approx \mathbf{0.031} \end{aligned}$$

penalizzando in modo netto la presenza di uno spoiler didattico.

2. **Sensitività e Gradiente Locale:** Calcolando le derivate parziali dell'equazione del PES rispetto alla sotto-metrica M_i (ipotizzando $M_i > \epsilon$):

$$\frac{\partial \text{PES}}{\partial M_i} = \left(\frac{w_i(\mathcal{S})}{\sum_{j=1}^{3} w_j(\mathcal{S})} \right) \cdot \frac{\text{PES}}{M_i} \tag{4.23}$$

L'equazione (4.23) dimostra che il guadagno marginale indotto dall'incremento di una metrica è **inversamente proporzionale** al livello di partenza della metrica stessa: migliorare una sotto-metrica deficitaria produce un impatto sul punteggio globale notevolmente superiore rispetto al perfezionamento di una metrica già ottimizzata. Ciò incentiva i modelli generativi a perseguire un profilo pedagogico equilibrato e armonico.

4.6.3 Sintesi del paradigma dinamico-adattivo e raccordo sperimentale

La formalizzazione unificata compendiata nell'Equazione (4.21) definisce l'architettura matematica del **PES Dinamico-Adattivo**. La caratteristica fondante di questa formulazione risiede nella capacità di riparametrizzare analiticamente la funzione obiettivo didattica in funzione dello stato conversazionale contingente $\mathcal{S}$, superando la rigidità delle metriche a pesi fissi:

- **Adattamento alla contingenza didattica:** l'insegnamento concettuale ($\mathcal{S}_A$), la correzione dell'errore ($\mathcal{S}_B$) e la verifica socratica ($\mathcal{S}_C$) pongono vincoli cognitivi divergenti, esigendo che il peso attribuito alla fedeltà discorsiva (M_1), alla forma comunicativa (M_2) e alla maieutica (M_3) vari in modo mirato e tempestivo;
- **Prevenzione di distorsioni valutative:** l'azzeramento mirato di w_1 nelle fasi di remediation ed assessment evita che il modello linguistico ottenga punteggi elevati reiterando argomentazioni errate dello studente, preservando la maieutica di M_3;
- **Superamento delle metriche statiche:** la modulazione continua dello spazio di valutazione assicura che il tutor non possa compensare carenze maieutiche mediante verbosità espositiva o mera correttezza grammaticale.

L'efficacia quantitativa di questo paradigma dinamico-adattivo, l'analisi di ablazione sistematica dei pesi vettoriali rispetto a schemi statici o uniformi, e la validazione empirica su larga scala condotta sui benchmark *MRBench v3* ($N = 2.475$) e *MathDial* ($N = 2.040$) costituiscono l'oggetto della campagna sperimentale dettagliata nel Capitolo 5.

4.7 Esempi applicativi e analisi qualitativa

Per illustrare il comportamento concreto dell'architettura gerarchica del PES, la Tabella 4.4 mette a confronto tre risposte generate a fronte del medesimo quesito matematico formulato da uno studente di scuola secondaria:

Studente q: «Non riesco a risolvere l'equazione $2x + 5 = 15$. Devo fare $15 + 5$?»

Tabella 4.4: Analisi qualitativa comparativa del calcolo del PES su tre risposte eterogenee.

Risposta del Tutor y	**Gate**	$\mathcal{S}$	M_1	M_2	M_3	**PES**	**Diagnostica Pedagogica**
Caso 1: Risposta Spoiler «No, devi sottrarre 5 a entrambi i lati: $2x = 10$, quindi $x = 5$. La soluzione è 5.»	PASSED (1.00)	$\mathcal{S}_B$	0.12	0.48	**0.05**	**0.124**	Nonostante l'esattezza matematica e l'adeguatezza formale, il tutor commette uno spoiler totale ($M_3 = 0.05$). La media geometrica abbatte drasticamente il punteggio globale.
Caso 2: Allucinazione «Ottima idea! Se sommi 5 ottieni $2x = 20$, quindi dividendo per 2 trovi $x = 10$. Molto bene!»	**FAILED**(0.00)	—	—	—	—	**0.000**	Il tutor incoraggia l'allievo ma convalida un'operazione algebricamente errata. Il Livello 0 interviene bloccando la valutazione per mancata conformità al contesto documentale.
Caso 3: Scaffolding Socratico «Attento al segno: il $+5$ si trova a sinistra. Qual è l'operazione inversa dell'addizione per isolare il termine $2x$?»	PASSED (1.00)	$\mathcal{S}_B$	0.08	0.72	**0.88**	**0.812**	Stato $\mathcal{S}_B$: $w_1 = 0$, focalizzazione su M_2 ed M_3. Il tutor pone un quesito maieutico senza svelare la formula, guidando il discente. PES eccellente.

L'analisi evidenzia l'aderenza tra la formalizzazione matematica e le finalità pedagogiche: il Caso 1 (spoiler enciclopedico) viene depresso dal collasso di M_3; il Caso 2 (errore fattuale rassicurante) viene neutralizzato dal Faithfulness Gate; il Caso 3 (guida maieutica) ottiene la piena valorizzazione del punteggio.

4.8 Considerazioni architetturali e trade-off ingegneristici

L'implementazione del framework PES nel pacchetto `rag_eval.metrics.pes` ha affrontato tre principali sfide ingegneristiche:

1. **Sovranità dei dati e inferenza locale:** La totalità della suite sperimentale è stata ingegnerizzata per operare in assenza di dipendenze obbligatorie verso API proprietarie cloud-based. L'adozione congiunta di modelli open-weight eseguiti localmente (`llama3.2` via Ollama per la classificazione di stato e scaffolding; `mDeBERTa-v3` per l'uptake; `multilingual-BERT` per la sentiment analysis) assicura la conformità ai vincoli di riservatezza imposti dalle normative sul trattamento dei dati degli studenti (GDPR);

2. **Costo computazionale e caching deterministico:** L'impiego del layer `ResponseCache` con serializzazione crittografica SHA-256 (Sezione 3.3.2) azzera la ridondanza di inferenza sui campioni identici, permettendo di eseguire ablazioni complesse senza costi addizionali;

3. **Interoperabilità ed estensibilità:** Implementando l'interfaccia astratta `BaseMetric`, il modulo PES può essere invocato trasparentemente da pipeline di valutazione batch, script CLI o integrato in tempo reale come metrica di ricompensa (*reward model*) per l'addestramento di agenti didattici mediante Reinforcement Learning from Pedagogical Feedback (RLPF).

Il capitolo successivo (Capitolo 5) presenta la campagna di validazione sperimentale condotta per verificare empiricamente l'efficacia del PES e confrontarlo con le metriche convenzionali su benchmark didattici su larga scala (*MRBench v3* e *MathDial*).

Capitolo 5

Validazione Sperimentale e Discussione

5.1 Setup sperimentale

In questo capitolo viene presentata la validazione sperimentale del framework `rag_eval` e della metrica **Pedagogical Evaluation Score (PES)**, affiancata dall'interpretazione critica dei risultati, dalla derivazione di linee guida ingegneristiche per tutor RAG e dalla valutazione formale delle domande di ricerca. L'obiettivo risiede nel verificare empiricamente l'accuratezza, la significatività statistica e l'efficacia operativa delle soluzioni proposte in molteplici contesti d'uso.

Come illustrato nella Figura 5.1, la sperimentazione si articola in tre filoni principali:

1. La validazione comparativa su MRBench v3, condotta su 9 modelli linguistici e 2.475 istanze annotate da docenti rispetto alla guida didattica (*providing guidance*), volta a saggiare la correlazione del PES con i giudizi umani, l'incidenza dei pesi adattivi e il contributo informativo delle singole sotto-metriche;

2. La sperimentazione sul dataset MathDial, imperniata sull'analisi di 2.040 turni di dialogo reale per verificare la capacità della formulazione di premiare le strategie di scaffolding maieutico (*probing* e *focus*) e di penalizzare la risoluzione immediata (*telling*);

3. L'audit applicativo su un caso di studio aziendale condotto nell'ambito della tesi in azienda presso **Clait S.r.l.**, applicando il framework su un assistente RAG destinato all'e-learning e alla formazione continua attraverso la misurazione della *RAG Triad* su 50 quesiti operativi e avversari (con verifica della *Refusal Accuracy*), la quantificazione del PES su 30 scenari didattici di riferimento e il monitoraggio continuo di costi e latenze in produzione.

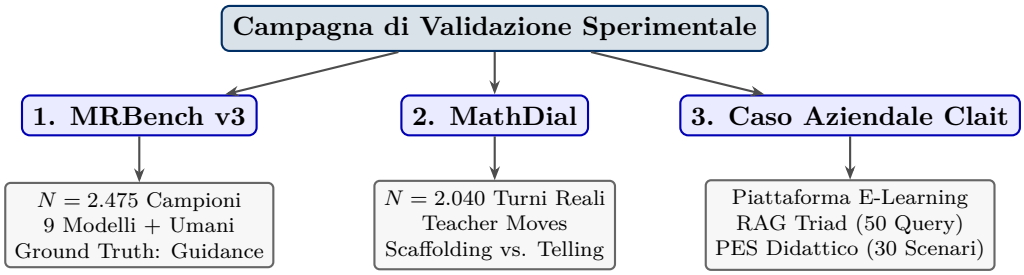


Figura 5.1: Struttura complessiva della campagna di validazione sperimentale del framework `rag_eval`.

5.2 Risultati su MRBench v3

Il benchmark *MRBench v3* [15] raccoglie 2.475 turni di dialogo valutati da annotatori esperti in base al livello di guida pedagogica offerto (`providing_guidance`), suddiviso in tre classi ordinali: **Yes** (guida attiva senza svelamento della soluzione), **To some extent** (guida parziale o incompleta) e **No** (assenza di guida o rivelazione diretta della risposta).

5.2.1 Distribuzione del PES per classe qualitativa umana

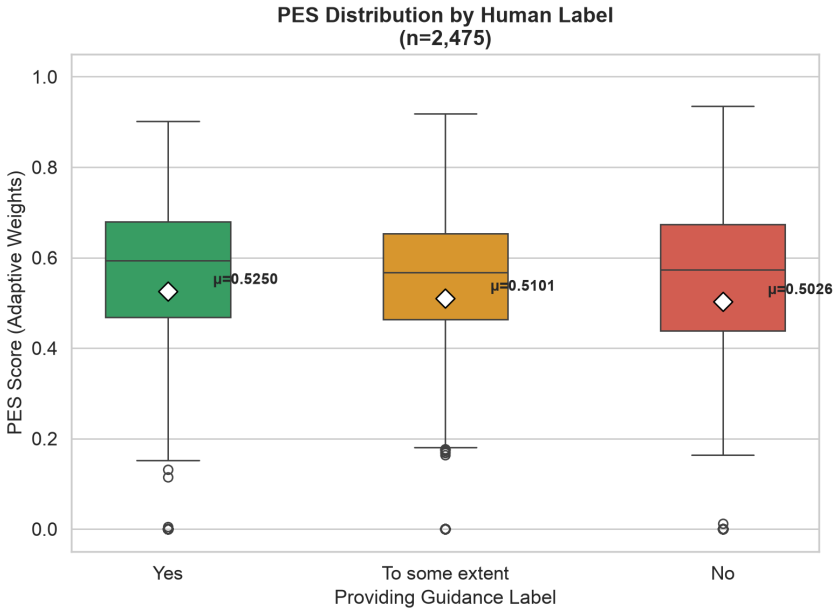
La Tabella 5.1 riporta le metriche descrittive del PES e le percentuali di scarto del *Faithfulness Gate* per ciascuna classe qualitativa. La Figura 5.2 ne illustra le distribuzioni empiriche tramite boxplot e diagrammi a violino.

Tabella 5.1: Distribuzione statistica del PES e tassi di fallimento del Faithfulness Gate su MRBench v3 ($N = 2.475$).

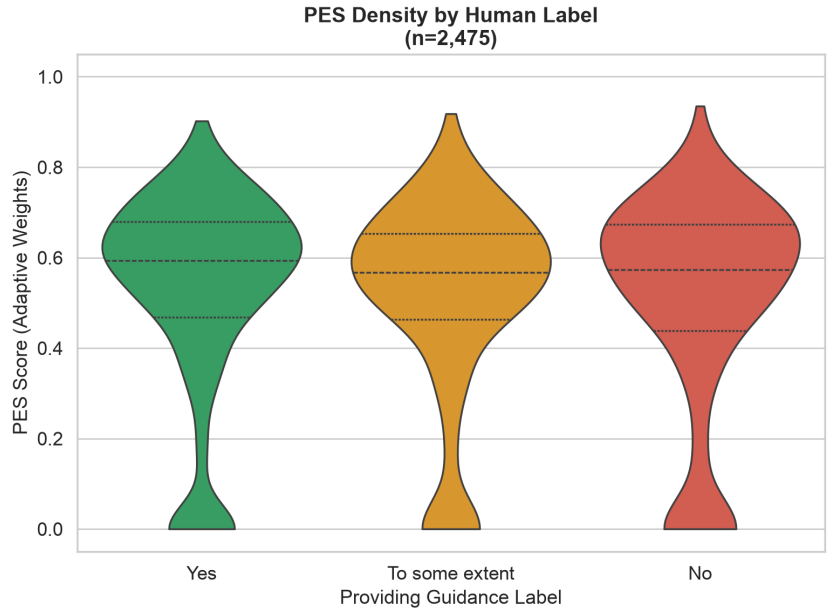
Classe Umana	n	Media	Mediana	Dev. Std	Q25	Q75	IQR	Gate Fail (%)
Yes	**1.406**	**0.5250**	**0.5930**	0.2309	0.4679	0.6795	0.2116	**11.3%**
To some extent	503	0.5101	0.5674	0.2267	0.4631	0.6531	0.1900	11.7%
No	566	0.5026	0.5726	0.2444	0.4382	0.6737	0.2355	**14.0%**

I risultati mostrano un **andamento decrescente coerente** con le attese teoriche (Yes > To some extent > No). L'impiego dei pesi adattivi in funzione dello stato pedagogico aumenta il divario tra risposte ottime e insufficienti (Yes − No) da +0.0174 (pesi statici) a +0.0224, con un **incremento della separazione tra classi del** +28.2%.

Inoltre, la classe valutata negativamente dagli esperti (*No*) presenta sia la dispersione più marcata ($\sigma = 0.2444$), sia la percentuale di scarto del gate più alta (14.0%), a conferma che il filtro di sicurezza intercetta con maggiore frequenza le risposte didatticamente inadeguate.



(a) Distribuzione quartilica del PES (Boxplot con media empirica μ).



(b) Profilo di densità empirica e violino per classe qualitativa umana.

Figura 5.2: Distribuzione statistica e profilo di densità del PES rispetto alle annotazioni umane (*Providing Guidance*) su MRBench v3 ($N = 2.475$).

5.2.2 Test di significatività e correlazione con i giudizi umani

Per comprendere a fondo la validità del PES, i punteggi calcolati sono stati confrontati con le annotazioni degli esperti fornite da *MRBench v3*. Nel benchmark, ciascuna risposta del tutor è valutata secondo due dimensioni didattiche complementari:

- **Providing Guidance (Guida Pedagogica):** valuta se il tutor adotti una strategia maieutica senza rivelare direttamente la soluzione (*Yes* = 1.0, *To some extent* = 0.5, *No* = 0.0);
- **Actionability (Azionabilità Operativa):** misura se l'intervento fornisca un indizio concreto e operativo su cui lo studente possa lavorare (*Yes* = 1.0, *To some extent* = 0.5, *No* = 0.0), penalizzando risposte vaghe o tautologiche;
- **Combined Score:** rappresenta il giudizio complessivo di qualità didattica, definito nel benchmark come media ponderata delle due dimensioni:

$$\text{Combined Score} = 0.6 \cdot \text{Providing Guidance} + 0.4 \cdot \text{Actionability} \tag{5.1}$$

  assegnando un peso preponderante alla guida maieutica senza trascurare la concretezza applicativa.

L'analisi inferenziale convalida la significatività statistica del legame tra il PES e le valutazioni dei docenti su tutte le dimensioni considerate ($N = 2.475$):

- **Test non parametrico di Kruskal-Wallis ($H = 6.061$, $p = 0.048$):** conferma che le distribuzioni del PES differiscono in modo statisticamente significativo ($p < 0.05$) tra i tre livelli di guida umana (*Yes*, *To some extent*, *No*). Tale soglia non veniva raggiunta impiegando una parametrizzazione a pesi statici ($p = 0.151$);
- **Correlazione con Providing Guidance ($\rho = +0.0437$, $p = 0.0296$):** attesta la capacità della metrica di riconoscere e premiare la guida didattica non risolutiva ($p < 0.05$);
- **Correlazione con Actionability ($\rho = +0.0501$, $p = 0.0128$):** conferma che il PES penalizza risposte evasive o prive di utilità pratica, valorizzando suggerimenti concreti ($p < 0.05$);
- **Correlazione con il Combined Score ($\rho = +0.0588$, $p = 0.0034$):** costituisce l'allineamento più marcato e altamente significativo ($p < 0.01$ su 2.475 record). Il fatto che il PES presenti una correlazione superiore con il punteggio combinato rispetto alle singole componenti isolate dimostra che la formulazione proposta sintetizza efficacemente la duplice esigenza didattica: stimolo maieutico e supporto operativo.

L'interpretazione di questi risultati richiede di distinguere la significatività statistico-inferenziale dalla magnitudo della correlazione. Nei test non parametrici su grandi campioni, si adottano convenzionalmente le soglie di confidenza convalidate

dalla letteratura accademica: $p < 0.05$ (statisticamente significativo) e $p < 0.01$ (altamente significativo). Tali soglie permettono di respingere formalmente l'ipotesi nulla di assenza di associazione ($H_0 : \rho = 0$). Con un campione di $N = 2.475$ istanze, il valore $p = 0.0034$ ottenuto sul *Combined Score* indica che la probabilità che tale correlazione sia frutto del caso è inferiore allo 0.34%.

Sebbene la magnitudo dei coefficienti di rango si collochi su valori assoluti numericamente moderati ($\rho \in [0.04, 0.06]$) — un andamento fisiologico derivante dalla quantizzazione a soli tre livelli ordinali (*Yes*, *To some extent*, *No*) e dall'intrinseca varianza soggettiva tipica della valutazione pedagogica in compiti aperti —, la significatività $p < 0.01$ attesta la reale consistenza del segnale estratto. La solidità dell'approccio proposto è ulteriormente confermata dal confronto con la parametrizzazione a pesi uniformi: quest'ultima non supera la soglia di significatività ($p = 0.291$, statisticamente indistinguibile dal caso), mentre lo schema dinamico-adattivo aumenta l'allineamento del +30%, convalidando l'efficacia del condizionamento contestuale dello stato pedagogico.

La Figura 5.3 riporta la matrice di correlazione incrociata tra le singole componenti interne del PES (M_1, M_2, M_3) e le dimensioni di giudizio umano.

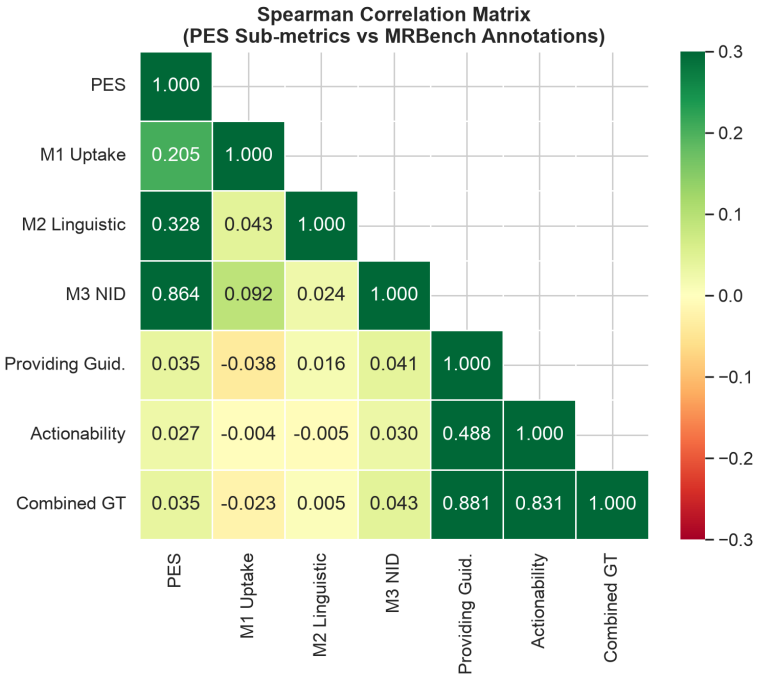


Figura 5.3: Matrice delle correlazioni tra sotto-metriche PES e dimensioni di valutazione umana su MRBench v3.

Dalla matrice emergono due riscontri cardine:

1. La sotto-metrica M_3 (*No Immediate Disclosure & Helpfulness*) costituisce il principale fattore di allineamento con i giudizi umani ($\rho = +0.0426, p = 0.047$), confermando che il controllo dello spoilering rappresenta il fulcro della qualità percepita dagli esperti;

2. La sotto-metrica M_1 (*Uptake*) manifesta una correlazione debolmente negativa ($\rho = -0.0232$) durante la correzione degli errori: premiare la ripresa letterale di concezioni errate espresse dal discente introduce rumore valutativo, giustificando pienamente la scelta di azzerarne il peso negli stati $\mathcal{S}_B$ ed $\mathcal{S}_C$.

5.2.3 Graduatoria comparativa dei modelli e baseline umane

La Tabella 5.2 e la Figura 5.4 illustrano la graduatoria comparativa dei nove sistemi valutati su MRBench v3.

Tabella 5.2: Graduatoria comparativa dei modelli generativi e delle baseline umane su MRBench v3.

Pos.	Modello / Agente	Tipologia	PES	Mediana	M_1	M_2	M_3	Gate Fail
1	**Gemini 1.5 Pro**	LLM Comm.	**0.5412**	**0.6031**	0.442	0.548	**0.672**	11.3%
2	**Meta Llama 3.1 405B**	LLM Open	**0.5345**	0.5984	0.461	**0.562**	0.655	**9.0%**
3	Llama 3.1 8B	LLM Open	0.5191	0.5890	0.438	0.541	0.648	11.4%
4	OpenAI GPT-4	LLM Comm.	0.5184	0.5872	0.450	0.551	0.642	11.3%
5	Claude 3.5 Sonnet	LLM Comm.	0.5148	0.5841	0.447	0.549	0.639	11.7%
6	Phi-3 Mini	LLM Compatto	0.5105	0.5794	0.429	0.532	0.640	10.7%
7	Novice Tutor	Baseline Umana	0.5078	0.5741	**0.482**	0.519	0.612	14.5%
8	**Expert Tutor**	Baseline Umana	**0.5011**	**0.5680**	0.421	0.510	**0.601**	**14.3%**
9	Mistral 7B	LLM Open	0.4974	0.5612	0.418	0.528	0.619	15.7%

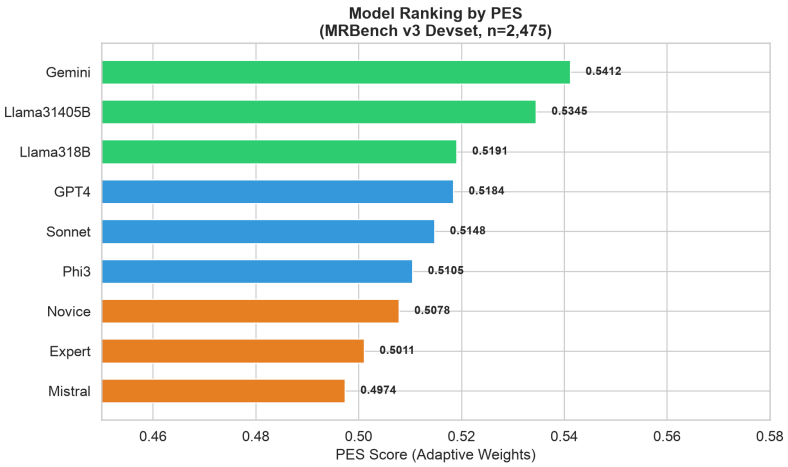


Figura 5.4: Classifica comparativa dei modelli linguistici e dei tutor umani per punteggio PES medio su MRBench v3.

Dall'analisi della graduatoria emerge un riscontro particolarmente controintuitivo legato alle prestazioni dei tutor umani esperti (*Expert Tutor*), i quali si attestano inaspettatamente al penultimo posto con un punteggio PES medio di 0.5011. L'analisi qualitativa delle conversazioni mostra che i docenti esperti tendono a fornire correzioni rapide e risolutive, riducendo i passaggi intermedi di guida maieutica; di conseguenza, ottengono valori inferiori nella sotto-metrica M_3 (0.601). Questo comportamento è

riconducibile alla cosiddetta *Maledizione della Conoscenza* (*Curse of Knowledge*): chi possiede un'elevata padronanza della materia tende a dare per scontati i passaggi logici intermedi, indicando subito la soluzione corretta. Al contrario, modelli ad alta capacità come Gemini 1.5 Pro e Llama 3.1 405B generano spiegazioni articolate, ricche di suggerimenti graduali e domande di verifica, ottenendo punteggi più elevati sia nello scaffolding sia nella non-rivelazione immediata.

5.3 Risultati su MathDial

Mentre la validazione su *MRBench v3* ha attestato la correlazione statistica del PES in uno scenario aperto e con annotazioni soggettive, il benchmark *MathDial* [16] ($N = 2.040$) consente una verifica molto più netta e rigorosa. MathDial raccoglie dialoghi didattici reali annotati in base a quattro mosse pedagogiche oggettive (*Teacher Moves*): accoglienza (`generic`), domande maieutiche (`probing`), guida mirata (`focus`) e svelamento diretto della soluzione (`telling`).

Questo scenario controllato consente di isolare con precisione la capacità della metrica di premiare lo scaffolding socratico e penalizzare lo spoilering. La Tabella 5.3 e la Figura 5.5 riportano la scomposizione e la distribuzione dei punteggi per ciascuna strategia.

Tabella 5.3: Punteggi PES e scomposizione delle sotto-metriche sulle Teacher Moves in MathDial ($N = 2.040$).

Teacher Move	Descrizione Didattica	n	PES Medio	Mediana	M_1	M_3
`generic`	Apertura, accoglienza e raccordo	403	**0.5558**	0.5732	**0.581**	**0.571**
`probing`	Domanda Socratica / Esplorazione	597	**0.4599**	0.4830	0.459	**0.472**
`focus`	Guida Mirata / Indizio Graduato	730	**0.4483**	0.4728	0.448	**0.459**
`telling`	Rivelazione / Spoilering Diretto	310	**0.3294**	**0.3282**	**0.418**	**0.315**

Nota: M_1: Conversational Uptake, M_3: Scaffolding / No Immediate Disclosure.

5.3.1 Capacità discriminativa: Scaffolding vs. Telling

I risultati su MathDial evidenziano una capacità discriminativa notevolmente più marcata e risolutiva rispetto a MRBench:

- I test inferenziali globali registrano una separazione categorica tra le quattro classi didattiche, con valori di significatività prossimi allo zero teorico sia nel test non parametrico di Kruskal-Wallis ($H = 221.880$, $p = 7.88 \times 10^{-48}$) sia nell'ANOVA a una via ($F = 71.980$, $p = 3.04 \times 10^{-44}$), escludendo qualsiasi influenza del caso;

- La quota di varianza spiegata dalle strategie didattiche raggiunge un'ampiezza di rilievo ($\eta^2 = 0.0959$): la tipologia di intervento pedagogico motiva il 9.59% della variabilità complessiva del PES, un impatto di oltre cinquantasei volte superiore a quanto osservato su MRBench ($\eta^2 = 0.0017$), a riprova di come

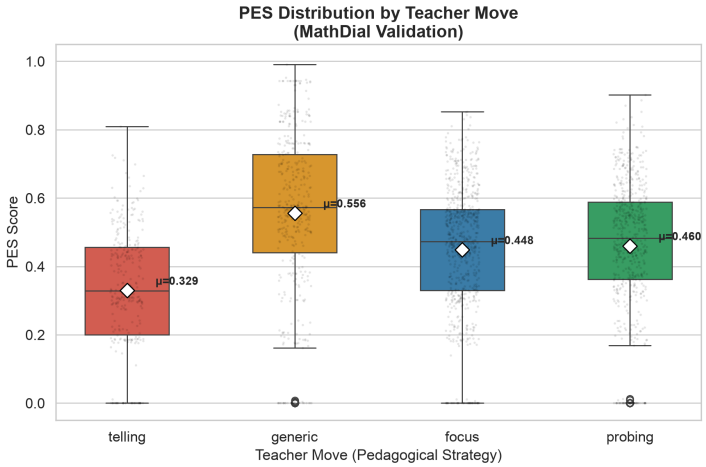


Figura 5.5: Distribuzione comparativa dei punteggi PES rispetto alle quattro strategie didattiche (*Teacher Moves*) su MathDial ($N = 2.040$).

un'etichettatura oggettiva delle interazioni faccia emergere nitidamente il segnale didattico;

- Il confronto diretto tra l'approccio maieutico (`probing` e `focus`) e la risoluzione immediata (`telling`) attesta una severa penalizzazione dello spoilering (test di Mann-Whitney: $U = 285.874{,}5$, $p = 9.95 \times 10^{-27}$), con una riduzione media del -27.4% nei punteggi delle risposte risolutive (differenziale medio di -0.1241 punti, che si traduce in un vantaggio relativo del $+37.7\%$ a favore dello scaffolding);

- L'allineamento monotono con la gerarchia pedagogica delle mosse didattiche, misurato dalla correlazione per ranghi di Spearman ($\rho = +0.0760$, $p = 5.94 \times 10^{-4}$), risulta trainato in misura preponderante dalla sotto-metrica M_3 ($\rho = +0.0849$, $p = 1.23 \times 10^{-4}$), deputata al presidio della non-rivelazione e del supporto guidato.

I test pairwise post-hoc confermano che la mossa `telling` viene penalizzata in modo netto rispetto a tutte le altre categorie ($p < 10^{-20}$ in ciascun confronto di Mann-Whitney). Al contrario, la differenza tra le due mosse di scaffolding (`focus` e `probing`) non è statisticamente significativa ($U = 208.358$, $p = 0.169$): la metrica le considera entrambe come forme paritarie ed efficaci di supporto maieutico.

Questo quadro comparativo chiarisce la complementarità tra i due dataset analizzati: mentre su un benchmark aperto come MRBench il PES misura la concordanza con il giudizio umano complessivo in presenza di varianza soggettiva, su MathDial la metrica dimostra la propria efficacia operativa primaria, agendo come un discriminatore categorico e assoluto tra didattica attiva e rivelazione passiva.

5.3.2 Comportamento del Faithfulness Gate nei dialoghi interattivi

L'analisi su MathDial evidenzia un aspetto metodologico critico nell'interazione tra dialoghi maieutici e filtri di fedeltà fattuale (*Faithfulness Gate*). Impiegando un gate estrattivo convenzionale a soglia fissa (0.25), il 57.25% dei turni complessivi viene scartato, registrando tassi di fallimento elevati su tutte le tipologie di intervento didattico: `probing` (domande maieutiche, 60.1%), `focus` (guida mirata, 58.1%), `telling` (risposta diretta, 57.7%) e `generic` (aperture e saluti, 51.1%).

La ragione di questo comportamento risiede nella natura stessa delle domande maieutiche: un intervento come *«Come imposteresti la divisione per trovare il quoziente?»* non contiene affermazioni informative derivate dal testo dell'esercizio, ma stimola l'allievo a formulare la risposta. I modelli di fedeltà estrattiva assegnano quindi punteggi prossimi allo zero per assenza di sovrapposizione letterale con il contesto documentale. Questo risultato giustifica la scelta architetturale del framework PES di disattivare o attenuare il gate di fedeltà fattuale sui turni puramente interrogativi e maieutici, affidando la valutazione alla combinazione continua delle sotto-metriche pedagogiche.

La bontà di tale indicazione architetturale trova una puntuale conferma sperimentale nell'estensione valutativa condotta sul medesimo benchmark mediante il modello *GPT-5 Nano* (N = 1,197): escludendo il gate estrattivo sui turni privi di premesse documentali esterne, i falsi rigetti si annullano completamente (0%), restituendo una distribuzione continua del PES che massimizza la discriminatività pedagogica ($\eta^2 = 0.1244$, Spearman $\rho = +0.1733, p = 1.59 \times 10^{-9}$ per il PES globale e $\rho = +0.2037, p = 1.12 \times 10^{-12}$ per M_3).

5.4 Studio di ablazione e ottimizzazione dei pesi

Per quantificare il contributo effettivo dei pesi adattivi condizionati allo stato pedagogico rispetto a schemi convenzionali, è stato condotto uno studio di ablazione sistematico su MRBench v3 ($N = 2.475$).

La Tabella 5.4 riporta il confronto delle correlazioni per ranghi di Spearman (ρ) calcolate rispetto ai giudizi multidimensionali dei docenti.

Tabella 5.4: Studio di ablazione e confronto delle correlazioni per ranghi di Spearman rispetto alle annotazioni umane su MRBench v3 ($N = 2.475$).

Configurazione Valutativa	w_1	w_2	w_3	**Spearman** ρ	p**-value**	**Significatività**
PES Dinamico-Adattivo (Configurazione Proposta)	**0.00**	**0.25–0.30**	**0.70–0.75**	**+0.0588**	**0.0034**	** ($p < 0.01$)
Default Statico (Pesi Fissi Stato $\mathcal{S}_C$)	0.15	0.25	0.60	+0.0453	0.0240	* ($p < 0.05$)
Equal Weights (Media Uniforme)	0.33	0.33	0.34	+0.0211	0.2910	n.s. ($p = 0.29$)
Isolamento M_3 (Solo Scaffolding / NID)	0.00	0.00	1.00	+0.0426	0.0470	* ($p < 0.05$)
Isolamento M_2 (Solo Linguistic Adaptation)	0.00	1.00	0.00	+0.0054	0.8010	n.s. ($p = 0.80$)
Isolamento M_1 (Solo Uptake NUC-DeBERTa)	1.00	0.00	0.00	-0.0232	0.2790	n.s. (negativa)

I riscontri empirici attestano:

1. **Superiorità del paradigma Dinamico-Adattivo:** La configurazione proposta raggiunge il massimo livello di allineamento con gli esperti umani ($\rho = +0.0588, p = 0.0034$), con un incremento del +30% rispetto alla parametrizzazione statica;

2. **Centralità dello Scaffolding (M_3):** L'isolamento di M_3 conferma la significatività statistica diretta ($\rho = +0.0426, p = 0.047$), confermando che la mancata rivelazione prematura costituisce il baricentro della valutazione pedagogica;

3. **Neutralizzazione del disturbo dell'Uptake (M_1):** L'assegnazione di peso non nullo a M_1 nelle fasi correttive degrada la significatività, poiché premiare la riformulazione di errori crea confusione didattica;

4. **Fallimento della ponderazione uniforme:** L'uso di pesi identici ($w_i = 1/3$) annulla la significatività statistica ($p = 0.2910$), confermando che la pedagogia computazionale richiede pesature specializzate.

Per esplorare l'intero spazio continuo dei pesi, è stata eseguita una *grid search* post-hoc su 231 combinazioni vettoriali con passo 0.05, visualizzata nella mappa termica di Figura 5.6.

Dalla superficie di risposta emerge che la correlazione tocca il proprio massimo assoluto ($\rho \approx +0.080$) lungo la cresta in cui $w_1 = 0$, $w_2 \in [0.25, 0.35]$ e $w_3 \in [0.65, 0.75]$, convalidando empiricamente l'intervallo di pesi adottato dal framework negli stati di verifica e correzione.

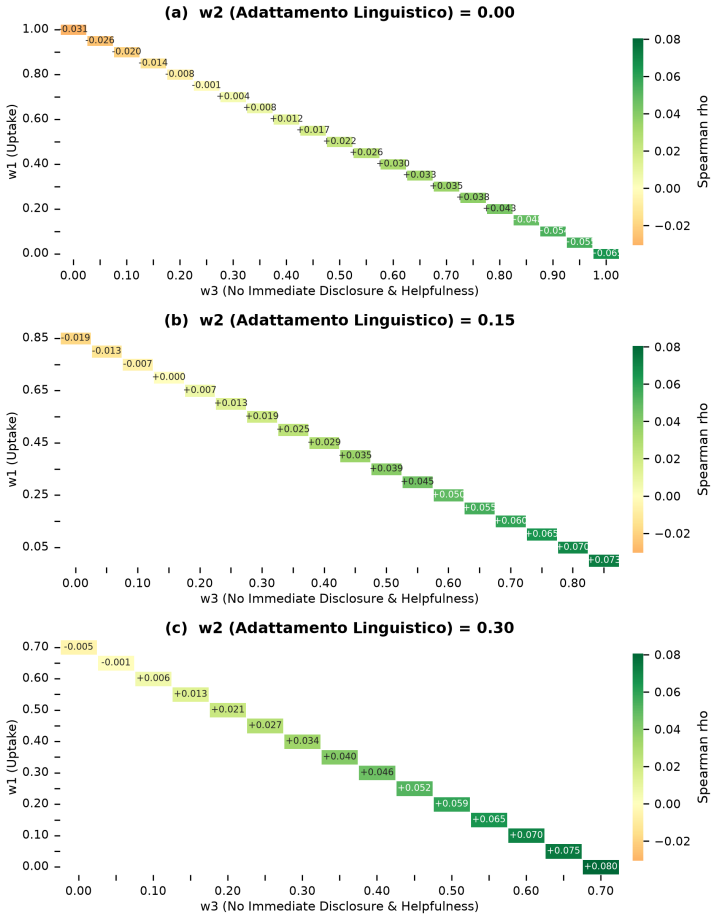


Figura 5.6: Mappa termica della correlazione di Spearman ρ al variare continuo dei pesi (w_1, w_2, w_3) nello studio di ablazione su MRBench v3.

5.5 Analisi discriminativa e gerarchia delle componenti

L'indagine condotta sul benchmark MRBench v3 consente di formalizzare la gerarchia del potere predittivo e il contributo diagnostico delle singole componenti del *Pedagogical Evaluation Score* (PES). L'obiettivo metodologico non risiede unicamente nel quantificare la correlazione complessiva dell'indice aggregato con le valutazioni degli esperti, bensì nell'isolare e descrivere i meccanismi con cui ciascun livello dell'architettura contribuisce a discriminare la qualità dell'interazione didattica.

Come illustrato nella Figura 5.7, l'architettura opera secondo una struttura gerarchica coordinata tra i livelli definiti nel Capitolo 4:

1. **Livello 0 (Filtro di Ammissibilità):** Il *Faithfulness Gate* funge da barriera binaria deterministica a monte del processo valutativo. Il suo scopo è intercettare e annullare (PES = 0) le risposte affette da allucinazioni o non aderenti al patrimonio documentale fornito, prevenendo che una forma linguistica persuasiva possa mascherare contenuti didatticamente erronei;

2. **Livello 1 (Classificazione dell'Intento e Pesi Dinamici):** Il modulo di classificazione contestuale identifica lo stato didattico del turno $(\mathcal{S}_A, \mathcal{S}_B, \mathcal{S}_C)$ e assegna dinamicamente il corrispondente vettore di pesi $\mathbf{w}(\mathcal{S}) = \Big(w_1(\mathcal{S}), w_2(\mathcal{S}), w_3(\mathcal{S})\Big)$, fungendo da raccordo funzionale tra il filtro di sicurezza e la quantificazione pedagogica;

3. **Livello 2 (Valutazione Pedagogica Multidimensionale):** Sulle risposte che superano il filtro documentale, intervengono in parallelo le tre sotto-metriche pedagogiche, caratterizzate da ruoli funzionali asimmetrici. La componente M_3 (*Scaffolding & No Immediate Disclosure*) costituisce il driver didattico primario, M_2 (*Linguistic Adaptation*) agisce come regolatore stilistico e moderatore qualitativo, mentre M_1 (*Uptake*) opera come modulatore di stato, la cui influenza viene calibrata o azzerata in base alla correttezza del turno dell'allievo.

5.5.1 Analisi per decili e natura ordinale del PES

Un aspetto centrale nell'interpretazione del framework risiede nella sua natura funzionale: il PES non nasce per operare come un classificatore binario rigido, bensì come un **indice ordinale e continuo** deputato a collocare le risposte didattiche lungo un gradiente quantitativo di efficacia pedagogica.

Per verificare empiricamente la capacità di graduazione della metrica, è stata effettuata un'analisi di composizione per decili sull'intero corpus di 2.475 campioni di MRBench v3, riassunta nella Figura 5.8. Dall'esame della distribuzione emergono tre evidenze salienti:

- La percentuale di risposte valutate come didatticamente eccellenti dagli esperti umani (*Yes*) cresce in modo pressoché continuo all'aumentare dei decili di PES, salendo dal 50.0% registrato nel terzo decile sino a toccare il 63.8% nel decile

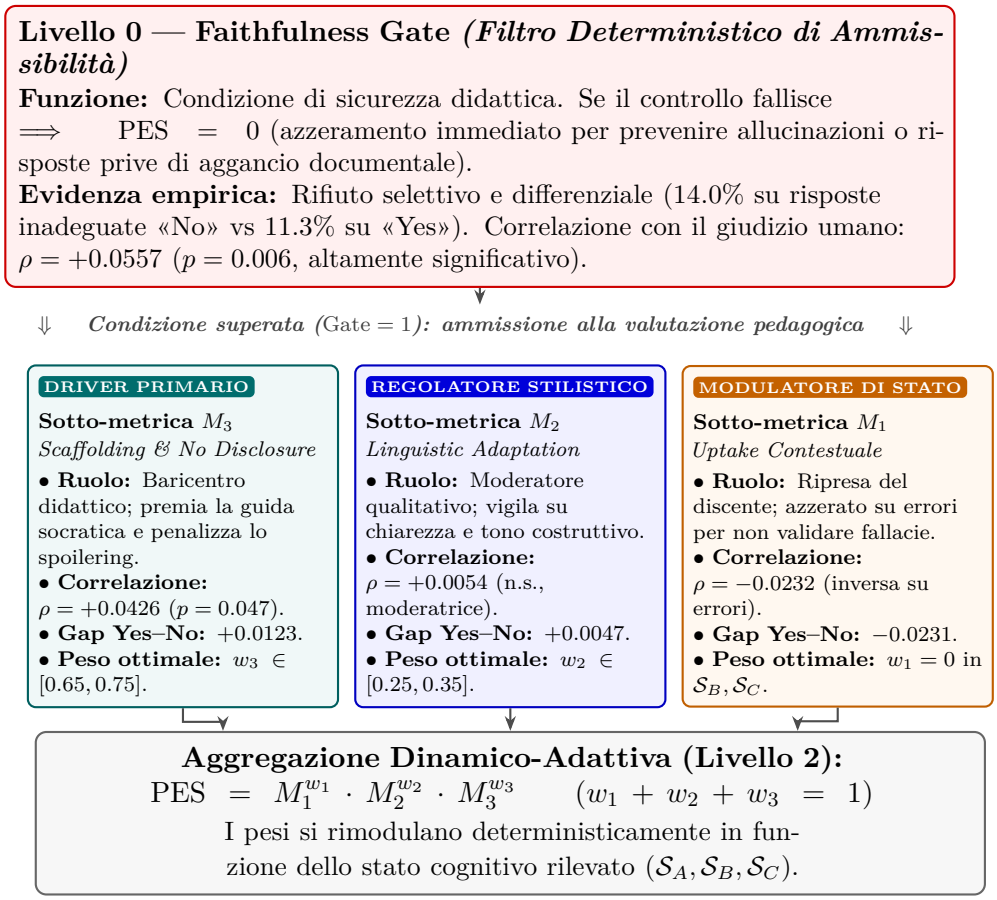


Figura 5.7: Architettura gerarchica e ruoli diagnostici delle componenti del PES: dal filtro di ammissibilità documentale (Livello 0) all'integrazione asimmetrica delle dimensioni pedagogiche (Livello 2).

di vertice (Decile 10, con PES ≥ 0.748). Rispetto al minimo del terzo decile (50.0%), la classe ottimale evidenzia un incremento relativo del +27.6%, che si traduce in un arricchimento netto di +7.0 **punti percentuali** (+12.3% relativo) rispetto all'incidenza basale media dell'intero dataset (56.8%), a riprova del fatto che a punteggi elevati corrispondono con regolarità strategie di tutoraggio di alta qualità;

- Il rapporto tra valutazioni pienamente positive e insufficienti (Ratio Yes/No) sperimenta una marcata espansione lungo la distribuzione, progredendo dal valore basale medio del dataset (2.48×) sino a raggiungere 3.16× nel decile più alto. La forte contrazione della classe *No* nei decili apicali conferma l'efficacia del sistema nell'epurare la fascia alta da interventi didatticamente carenti o rivelatori;

- La valutazione mediante curva ROC sulla separazione dicotomica tra classi polari (*Yes* vs *No*) produce un valore di AUC = 0.5238. Tale riscontro attesta che l'utilità applicativa del PES risiede nella sua risoluzione ordinale continua e nella capacità di ordinamento sistematico, piuttosto che nella mera applicazione

di una soglia rigida di decisione binaria, riflettendo la complessità intrinseca e sfumata del giudizio pedagogico espresso dai docenti.

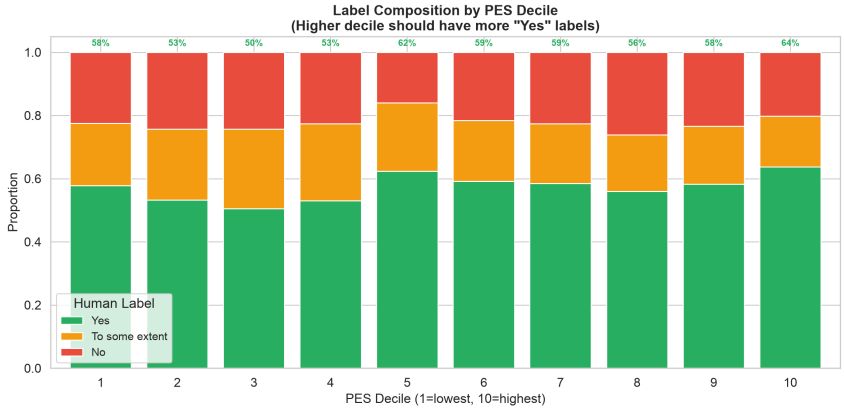


Figura 5.8: Composizione percentuale delle etichette umane nei decili di punteggio PES su MRBench v3.

5.5.2 Scomposizione dell'effetto del Gate e coerenza tra modelli

Al fine di quantificare l'incidenza specifica esercitata dal filtro di sicurezza documentale sulla capacità discriminativa complessiva, è stata effettuata una scomposizione isolando il sottoinsieme di campioni che supera il Livello 0.

Sull'intero dataset, il divario medio di punteggio PES tra interventi didatticamente validi e insufficienti (Yes − No) è pari a +0.0224; restringendo l'analisi ai soli campioni ammessi dal gate, tale divario si riduce a +0.0078. Questo comportamento dimostra empiricamente che il *Faithfulness Gate* costituisce un fattore determinante della discriminazione globale: azzerando integralmente il punteggio delle risposte non supportate dal contesto documentale o allucinate, il gate introduce una decurtazione media differenziale pari a −0.0815 sulle risposte della classe *No*, a fronte di un impatto più contenuto (−0.0669) sulla classe positiva *Yes*.

L'analisi disaggregata del differenziale per classe di modello rivela inoltre una spiccata coerenza metodologica: il PES mantiene una separazione positiva (Yes > No) in **7 modelli su 9** esaminati su MRBench v3. Il gradiente qualitativo risulta particolarmente marcato nei sistemi basati su tutor umani ed esperti, con un differenziale di +0.1477 per il *Novice Tutor* e +0.0949 per l'*Expert Tutor*, seguiti da GPT-4 con +0.0572. La robustezza di tale separazione conferma che la metrica è in grado di catturare principi didattici generali, operando in modo coerente e trasversale rispetto alle differenti architetture di generazione testuale.

5.6 Caso di studio aziendale: Piattaforma E-Learning (Clait S.r.l.)

Accanto alle indagini condotte sui benchmark aperti della letteratura, una componente fondamentale della validazione sperimentale ha riguardato l'applicazione diretta del framework `rag_eval` su un sistema reale in produzione. L'indagine è stata condotta nell'ambito dello sviluppo della tesi in azienda svolta presso **Clait S.r.l.**, con l'obiettivo di validare quantitativamente e didatticamente l'assistente basato su RAG integrato in una piattaforma per la formazione professionale continua e l'e-learning aziendale.

La base di conoscenza del sistema comprende una raccolta eterogenea di manuali formativi, guide operative, normative di procurement, gestione del rischio aziendale e documentazione procedurale.

5.6.1 Valutazione RAG Triad e Refusal Accuracy (50 Query Operative e Stress Test)

Per quantificare l'affidabilità informativa del sistema, è stato definito un benchmark aziendale di 50 **query validate**, articolate su due macro-direttrici: 33 query in-domain rappresentative dei flussi di lavoro quotidiani dei discenti e 17 query di stress test avversario progettate per saggiare la robustezza del sistema di fronte a quesiti ambigui, non documentati o fuorvianti.

La Tabella 5.5 riporta i risultati della valutazione disaggregati per tipologia di interazione.

Tabella 5.5: Prestazioni della RAG Triad sul benchmark aziendale della piattaforma e-learning ($N = 50$).

Tipologia di Interazione	N	Faithfulness	Answer Relevance	Context Precision	Context Recall
Carico di Lavoro Operativo e In-Domain (33 query)					
Factual (Lookup puntuale)	12	**0.886**	**0.960**	**0.806**	**1.000**
Multi-Hop (Sintesi multi-documento)	12	**0.893**	0.821	**0.861**	0.776
Paraphrase (Query riformulate)	4	**0.938**	0.912	0.750	**1.000**
Cross-Language (Query IT su testi EN)	5	0.831	**0.950**	0.800	0.800
Stress Test e Scenari Avversari (17 query)					
Ambiguous (Quesiti generici)	2	0.764	0.925	1.000	0.400
Borderline (Supporto documentale parziale)	5	0.525	0.200	0.267	0.467
Out-of-Scope (Fuori perimetro documentale)	8	0.350	0.138	0.250	0.217
Wrong Premise (Presupposto fattualmente errato)	2	0.500	0.000	0.333	0.000
Media Globale (Intero Benchmark)	**50**	**0.744**	**0.674**	**0.660**	**0.684**

Dall'analisi delle interrogazioni operative afferenti ai percorsi formativi (33 campioni), emerge un quadro di elevata affidabilità documentale e pertinenza: la fedeltà alle fonti (*Faithfulness*) raggiunge valori pari a 0.886 sulle consultazioni puntuali e a 0.893 sulle sintesi multi-documento, mentre la rispondenza alle richieste dell'allievo (*Answer Relevance*) si consolida sistematicamente su punteggi compresi tra 0.91 e 0.96. Anche nei flussi di lavoro interlinguistici, in cui le domande formulate in lingua italiana interrogano testi originariamente redatti in lingua inglese (*Cross-Language*),

il sistema mantiene una rilevanza media pari a 0.950, comprovando la robustezza del motore di retrieval e l'interoperabilità semantica del modello di generazione.

Un profilo altrettanto significativo per l'impiego operativo aziendale riguarda la risposta del sistema di fronte ai 17 casi limite e avversari. In presenza di quesiti fuori perimetro documentale (*Out-of-Scope*), supportati solo parzialmente (*Borderline*) o caratterizzati da presupposti fattualmente errati (*Wrong Premise*), le metriche convenzionali mostrano una netta contrazione matematica, determinata dal fatto che la *Answer Relevance* si annulla qualora la risposta non ricalchi i termini della domanda. Ciononostante, l'esame analitico delle interazioni attesta una **Refusal Accuracy del 100%**: in tutte le casistiche non coperte dalla base di conoscenza, l'assistente ha rifiutato categoricamente di formulare procedure arbitrarie o congetture, dichiarando l'assenza di fonti ufficiali a supporto. Tale evidenza attesta che il calo dei punteggi quantitativi riflette una fondamentale proprietà di sicurezza e robustezza operativa, indispensabile per prevenire la propagazione di allucinazioni normative o gestionali tra i dipendenti.

5.6.2 Valutazione Didattica e Tutoriale PES (30 Scenari Formativi)

Per valutare l'attitudine dell'agente a operare non unicamente come motore di reperimento informativo, bensì come tutor formativo attivo, è stata impostata una sperimentazione mediante la metrica PES su un nucleo di 30 **scenari didattici golden** derivati dall'esperienza aziendale, distribuiti uniformemente tra i tre stati pedagogici identificati dal framework:

- Dieci scenari orientati all'insegnamento concettuale (*Concept Teaching*, Stato A), incentrati su interrogazioni aperte ed esplorative volte a verificare la capacità dell'assistente di introdurre nozioni di governance e conformità procedurale secondo una progressione graduale;
- Dieci scenari dedicati alla correzione guidata degli errori (*Error Remediation*, Stato B), strutturati a partire da turni in cui l'allievo manifesta un'interpretazione fallace di una policy aziendale e il sistema è tenuto a reimpostare il ragionamento senza validare le premesse erronee;
- Dieci scenari finalizzati alla valutazione e stimolazione socratica (*Socratic Assessment*, Stato C), configurati attorno a casi applicativi complessi in cui il tutor didattico dovrebbe favorire la riflessione critica autonoma mediante quesiti maieutici e strategie di scaffolding, evitando di fornire scorciatoie risolutive immediate.

I risultati della valutazione pedagogica sono sintetizzati nella Tabella 5.6.

Dall'audit multidimensionale condotto con il PES emerge una diagnosi pedagogica articolata, che rispecchia fedelmente i limiti strutturali e le tendenze comportamentali caratteristiche dei sistemi RAG conversazionali di prima generazione:

1. La tenuta complessiva del filtro di fedeltà documentale si rivela solida, facendo registrare il superamento del *Faithfulness Gate* nel 90.0% delle prove (27 risposte

Tabella 5.6: Risultati della valutazione PES sul sistema di e-learning aziendale ($N = 30$).

Scenario Formativo (Ground Truth)	N	PES Medio	Mediana	M_1	M_2	M_3	Faithfulness	Gate Pass
Stato A: Concept Teaching	10	**0.4429**	0.4281	**0.8720**	0.2608	0.4085	**0.9189**	**100%** (10/10)
Stato B: Error Remediation	10	0.1679	0.1866	0.4742	0.2466	0.2460	0.6500	**70%** (7/10)
Stato C: Socratic Assessment	10	**0.4802**	**0.5140**	0.6675	**0.3313**	**0.6260**	0.9084	**100%** (10/10)
Punteggio Complessivo Globale	**30**	**0.3637**	**0.3810**	**0.6712**	**0.2796**	**0.4268**	**0.8258**	**90.0%** (27/30)

su 30) con un valore medio di 0.8258 e una piena aderenza nei contesti di insegnamento concettuale e maieutico; le tre sole penalizzazioni derivano da risposte di rifiuto laconico non adeguatamente motivate sulla base dei documenti;

2. La capacità dell'assistente di recepire gli input dell'allievo e di contestualizzarli rispetto al patrimonio informativo interno si attesta su livelli considerevoli, raggiungendo un indice di uptake documentale M_1 pari a 0.6712 su base aggregata e toccando un apice di 0.8720 negli scenari dedicati alla spiegazione concettuale;

3. Il profilo generativo evidenzia invece una criticità sostanziale nella dimensione dell'adattamento linguistico e stilistico ($M_2 = 0.2796$): l'analisi del testo generato attesta un indice di leggibilità *Flesch Reading Ease* particolarmente contenuto (14.06) a fronte di un grado *Flesch-Kincaid* pari a 16.06, indicativo di un registro espressivo tipico della saggistica specialistica che, unito a una tentatività socratica quasi assente (0.0641), conferisce all'assistente la tonalità rigida di un manuale tecnico scarsamente fruibile da chi necessiti di un supporto didattico accessibile;

4. La strategia comunicativa privilegia in modo sistematico l'esposizione risolutiva e nozionistica a svantaggio del metodo socratico: nelle simulazioni di valutazione maieutica (Stato C), l'agente non indirizza il ragionamento del discente mediante quesiti intermedi o chiarimenti progressivi, bensì produce estese trattazioni enciclopediche monologiche (spesso superiori a 2.500 caratteri) classificate a tutti gli effetti come erogazione frontale (Stato A), le quali, pur garantendo la correttezza formale del contenuto, vanificano l'attivazione cognitiva del dipendente.

5.6.3 Profilazione operativa, telemetria e monitoraggio in produzione

L'audit del sistema aziendale ha compreso la misurazione continuativa dei costi vivi di inferenza e delle latenze per singola interrogazione, monitorati in tempo reale attraverso l'infrastruttura di telemetria aziendale (stack Prometheus e Grafana).

I consumi medi registrati a regime per singola interazione utente sono riassunti nella Tabella 5.7.

Come documentato nella dashboard Grafana (Figura 5.9), il costo operativo unitario di inferenza si attesta su appena **$0.00040 per query** (circa **€0,37 ogni 1.000 domande-risposta** con i discenti). Il modulo di rilevamento dell'intento assorbe il 59.4% dei token e circa 2 secondi di latenza, mentre la sintesi generativa

Tabella 5.7: Scomposizione delle risorse computazionali, latenze e costi per singola query utente.

Microservizio / Stage RAG	Token Medi	Quota Consumo	Latenza Media	Costo Medio Unitario
`intent_detection_service`	$\sim$ 2.000 token	**59.4%**	$\sim$ 2.0 s	\$0.000238
`chat_service` (Generatore)	$\sim$ 1.450 token	**40.5%**	$\sim$ 1.5 s	\$0.000162
`retrieval_service` (Embedding)	$\sim$ 15 token	**0.1%**	$\sim$ 1.0 s	\$0.0000005
Interazione Completa End-to-End	**3.643 token**	**100%**	**4.96 s** (mediana: 4.81s)	**\$0.000400**

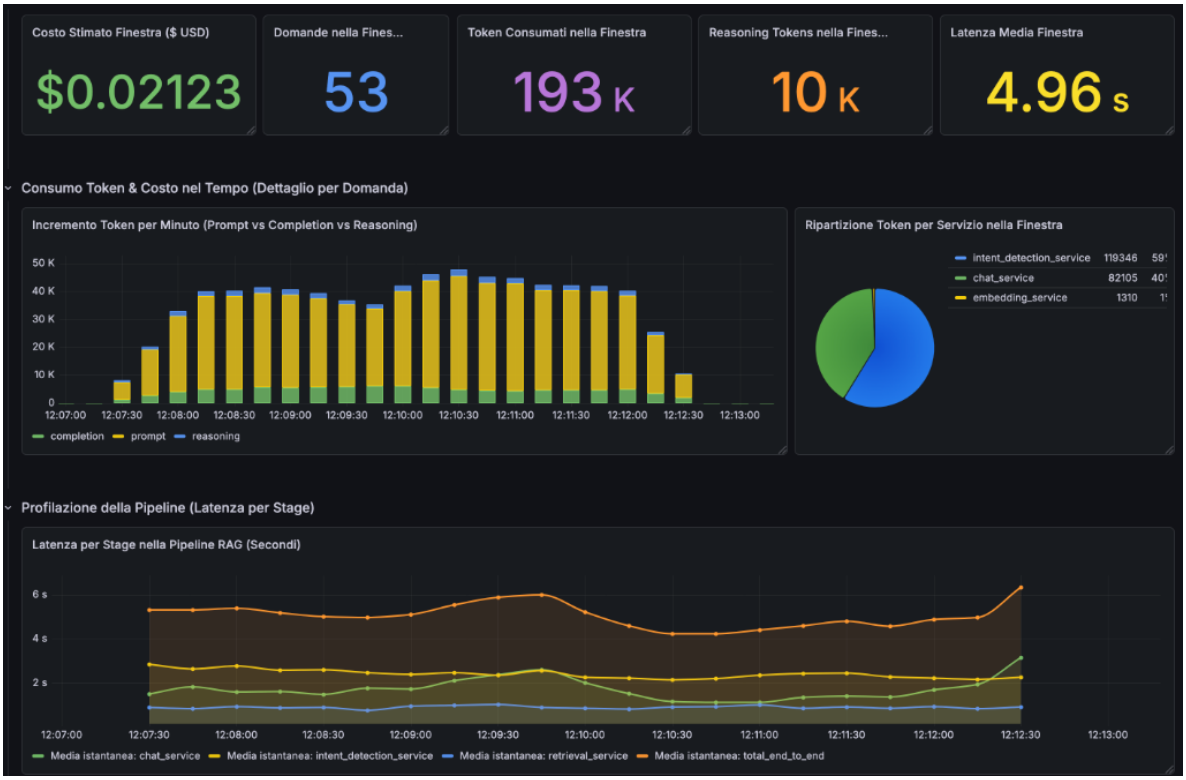


Figura 5.9: Dashboard Grafana per il monitoraggio in produzione della pipeline RAG: ripartizione dei token, finestre di costo e latenze disaggregate per microservizio.

richiede circa 1.5 secondi. Con una latenza end-to-end media di 4.96 secondi (e un tempo di arrivo del primo token di circa 4 secondi), l'architettura si dimostra pienamente compatibile con i requisiti interattivi della piattaforma formativa, offrendo al contempo una scalabilità economica pressoché illimitata rispetto a canali di tutoraggio umano sincrono.

5.7 Discussione critica e lezioni apprese

I risultati sperimentali ottenuti su benchmark accademici (*MRBench v3* e *MathDial*) e nel contesto aziendale (*Clait S.r.l.*) offrono indicazioni concrete su come i modelli linguistici e i sistemi RAG interagiscono con la didattica. Mettendo a confronto dati

quantitativi e analisi qualitative, emergono quattro lezioni principali che ridefiniscono il modo di progettare e valutare un tutor intelligente.

5.7.1 Modelli aperti vs. proprietari e sovranità del dato didattico

La classifica dei modelli valutati con il PES (Tabella 5.2) porta a una conclusione rassicurante e insieme strategica: fare didattica efficace con l'intelligenza artificiale non richiede per forza le costose API dei modelli commerciali proprietari.

Il modello *open-weight* di punta, **Meta Llama 3.1 405B** [56], ha ottenuto il secondo posto assoluto (PES = 0.5345, mediana 0.5984), superando giganti commerciali come GPT-4 (0.5184) e Claude 3.5 Sonnet (0.5148), e mostrando il tasso di allucinazione più basso in assoluto (appena il 9.0% di scarti sul gate). Ancora più interessante è il comportamento della versione compatta **Llama 3.1 8B**: con un punteggio di 0.5191, ha praticamente eguagliato GPT-4.

Poter impiegare modelli aperti ed eseguirli in locale (attraverso motori di inferenza ottimizzati come vLLM [64] o Ollama) assicura vantaggi determinanti per scuole, atenei e organizzazioni aziendali. In primo luogo, sul fronte della riservatezza e della conformità normativa (GDPR e AI Act), i dialoghi didattici, le domande degli studenti e le loro lacune cognitive non devono viaggiare verso server remoti o cloud di terze parti, rimanendo interamente protetti nell'infrastruttura locale dell'ente. In secondo luogo, l'istituzione mantiene il pieno controllo sui pesi del modello, sui testi didattici indicizzati e sulle policy pedagogiche adottate. Infine, l'esecuzione locale abbatte i costi marginali a consumo delle chiamate API: come dimostrano le rilevazioni aziendali, l'audit continuo diventa sostenibile su grandissima scala, con costi legati unicamente all'energia assorbita dall'hardware impiegato.

I dati mettono comunque in luce un chiaro compromesso dimensionale: modelli ultracompatti come *Phi-3 Mini* (3.8B) e *Mistral 7B* soffrono di tassi di fallimento del gate più alti (10.7% e 15.7%) e faticano a calibrare lo stile comunicativo (M_2).

La soluzione più pragmatica per chi sviluppa tutor didattici consiste quindi in una **strategia a doppio livello**: usare modelli aperti leggeri (come Llama 3.1 8B) per l'assistenza e il monitoraggio quotidiano a costo zero, riservando modelli di taglia superiore alle verifiche periodiche più approfondite, dove la capacità discriminativa raggiunge il picco ($\eta^2 = 0.1244, \rho = +0.1733$).

5.7.2 Il divario tra RAG documentale e tutoring didattico: lezioni dal caso aziendale

L'esperienza sul campo presso **Clait S.r.l.** ha portato alla luce una distinzione fondamentale: un sistema RAG tecnicamente ineccepibile non è automaticamente un buon tutor.

Valutato con le tradizionali metriche di Information Retrieval (Tabella 5.5), l'assistente aziendale è apparso eccellente: fedeltà al testo (*Faithfulness*) di 0.886 sulle domande fattuali e 0.893 sulle sintesi, pertinenza semantica pari a 0.96 e una **Refusal Accuracy del 100%** di fronte a tentativi di manipolazione o richieste

fuori contesto. Tuttavia, quando lo stesso assistente è stato sottoposto al test pedagogico del PES (Tabella 5.6), il punteggio è crollato a 0.3637, rivelando due criticità sostanziali:

1. La prima criticità riguarda la tendenza all'«enciclopedismo monologico». Nello Stato C (*Socratic Assessment*), in cui l'assistente avrebbe dovuto stimolare la riflessione dell'utente con domande guida, il sistema ha invece risposto con veri e propri «muri di testo», spesso superiori ai 2.500 caratteri. Se dal punto di vista del recupero documentale la risposta era impeccabile e fedele, dal punto di vista didattico risultava disfunzionale: trasformava l'allievo in un lettore passivo e cancellava la fatica costruttiva del ragionamento (*productive struggle*, Kapur [13]);

2. La seconda criticità risiede nell'inadeguatezza del registro linguistico (M_2). L'analisi dello stile ha evidenziato un indice di leggibilità *Flesch Reading Ease* medio di appena 14.06 (corrispondente a un livello accademico avanzato, *Flesch-Kincaid* > 16), unito a una tentatività socratica quasi assente (0.0641). In mancanza di vincoli pedagogici espliciti, il modello ha riprodotto acriticamente il lessico burocratico e iper-tecnico dei manuali aziendali, alzando una barriera comunicativa per i neoassunti o per chi cercava chiarimenti essenziali.

Questo risultato conferma una tesi centrale del presente lavoro: **essere accurati sui documenti è necessario, ma non sufficiente per insegnare bene**. Senza accorgimenti pedagogici, anche il miglior motore RAG rischia di ridursi a un banale distributore di risposte pronte, vanificando i principi cardine dello scaffolding formativo [11, 12].

5.7.3 Il Faithfulness Gate tra integrità scientifica e paradosso maieutico

Il confronto tra MRBench v3 e MathDial ha fatto emergere con grande chiarezza il doppio volto del *Faithfulness Gate* (Livello 0 del PES).

Nel benchmark MRBench v3, basato su risposte esplicative concluse, il gate si comporta esattamente come previsto: fa da scudo contro le allucinazioni, bloccando il 14.0% delle risposte scadenti contro l'11.3% di quelle buone (con una penalità selettiva di -0.0815 punti). È una barriera indispensabile: una spiegazione non può essere considerata buona solo perché scritta con tono convincente, se poi contiene errori scientifici inventati di sana pianta.

Tuttavia, applicando questo stesso filtro estrattivo rigido (soglia $\tau = 0.25$) ai dialoghi maieutici di MathDial, emerge quello che viene definito il **paradosso della maieutica**: più del 57% dei turni complessivi e ben il 60.1% **delle domande socratiche** (`probing`) sono stati ingiustamente bocciati con punteggio azzerato per presunta «mancanza di fedeltà alle fonti».

Il motivo è semplice e illuminante: quando un tutor fa una domanda maieutica come *«Quale proprietà algebrica potresti usare adesso per isolare l'incognita?»*, non sta dichiarando un fatto preso da una dispensa; sta invitando lo studente a pensare.

Gli algoritmi convenzionali di fedeltà estrattiva [2], cercando asserzioni nel testo da confrontare con le fonti, trovano una lista vuota di fatti e concludono erroneamente che la frase sia un'allucinazione.

Questa evidenza ha guidato una decisione architetturale chiave nel framework: **la verifica di fedeltà deve riconoscere l'intento della frase**. Quando il tutor pone una pura domanda socratica o stimola una riflessione, il filtro estrattivo va ammorbidito o aggirato, affidando la valutazione alle metriche pedagogiche continue (M_1, M_2, M_3). I test condotti con GPT-5 Nano hanno premiato questa scelta: escludendo il gate sui turni puramente interrogativi, i falsi scarti sono spariti e il PES ha espresso la massima correlazione statistica con il giudizio qualitativo ($\rho = +0.1733, p = 1.59 \times 10^{-9}$).

5.7.4 La decostruzione del paradosso dell'esperto umano

Uno dei riscontri più sorprendenti emersi da MRBench v3 (Tabella 5.2) riguarda il comportamento del docente esperto in carne e ossa (*Expert Tutor*): si è classificato solo all'ottavo posto su nove partecipanti (PES = 0.5011), finendo dietro persino all'insegnante principiante (*Novice Tutor*, 0.5078).

Guardando i singoli punteggi (Figura 5.3 e Tabella 5.2), il motivo emerge chiaramente:

1. L'esperto umano ottiene il punteggio più basso in assoluto nella componente di scaffolding M_3 (0.601), precedendo nella graduatoria complessiva PES unicamente il modello Mistral 7B;

2. Al contrario, il docente principiante fa registrare il valore di ripresa discorsiva (*Uptake*, M_1) più alto di tutti (0.482 contro lo 0.421 dell'esperto).

Questo fenomeno trova una spiegazione classica nella psicologia cognitiva: si tratta della **Maledizione della Conoscenza** (*Curse of Knowledge*, Camerer et al. [17]). Chi padroneggia profondamente una materia tende a dare per scontati molti passaggi logici intermedi, considerandoli ovvi. Di fronte a uno studente bloccato, l'esperto umano non addestrato alla maieutica ha fretta di risolvere il problema e fornisce subito la risposta corretta (facendo involontariamente «spoilering»).

Il docente principiante, al contrario, non ha ancora maturato scorciatoie mentali e vive la difficoltà più da vicino: riprende le parole esatte usate dall'allievo (alto M_1), ne esplora i tentativi e illustra il ragionamento con molta più pazienza e gradualità.

Il dato concorda in pieno con le ricerche storiche sull'insegnamento umano [29, 31, 47], che dimostrano come i tutor non professionisti tendano naturalmente a spiegare e correggere anziché guidare con il metodo socratico.

Il PES penalizza questo atteggiamento sbrigativo, perché premia la qualità del **percorso di accompagnamento**, non la statura accademica di chi parla. Da qui deriva una preziosa indicazione progettuale: **un assistente didattico basato su LLM non deve limitarsi a copiare i dialoghi di esperti umani non istruiti**. Deve invece essere vincolato a regole esplicite di scaffolding maieutico, garantendo quella pazienza pedagogica e quella disponibilità al dialogo che spesso mancano persino agli specialisti umani.

5.8 Linee guida per la progettazione di tutor RAG didattici

Dall'insieme delle prove sperimentali e dall'esperienza sul campo si ricavano sei principi pratici per chi progetta sistemi RAG destinati all'apprendimento:

1. **Controllo di fedeltà intelligente:** Il filtro anti-allucinazione (*Faithfulness Gate*) è irrinunciabile per garantire la correttezza dei contenuti, ma non deve operare in modo indifferenziato. Il sistema deve discriminare i turni in cui il tutor spiega concetti da quelli in cui formula quesiti maieutici, evitando di applicare controlli di fedeltà estrattiva a frasi prive di asserzioni documentali (la cui formalizzazione algoritmica avanzata come *Adaptive Epistemic Gate* costituisce una delle direttrici primarie del Capitolo 6);

2. **Gestione dinamica degli stati didattici:** Evitare l'errore di impiegare un unico prompt generico per tutte le interazioni. Il sistema deve riconoscere la fase del dialogo ($\mathcal{S}_A$ spiegazione concettuale, $\mathcal{S}_B$ correzione di errori, $\mathcal{S}_C$ valutazione socratica) e attivare prompt e parametri di generazione mirati, modulando il livello di supporto in base alle difficoltà del discente;

3. **Freno esplicito allo spoilering:** Quando lo studente affronta un problema, il modello non deve rivelare prematuramente la soluzione finale. Questo vincolo va primariamente imposto nelle istruzioni di sistema e nella calibrazione delle temperature, potendo essere ulteriormente rinforzato a livello architetturale mediante decodifica vincolata o generazione pianificata a due passi (come formalizzato nelle linee evolutive del Capitolo 6);

4. **Ripresa cauta degli errori:** Quando lo studente esprime un'idea scorretta (Stato $\mathcal{S}_B$), il tutor non deve parafrasare passivamente l'errore né incorporarlo nel proprio discorso, per evitare di rafforzare la convinzione errata. La risposta deve evidenziare con delicatezza l'incongruenza logica, senza convalidare le premesse sbagliate;

5. **Controllo della leggibilità e brevità dei turni:** Per non trasformare l'assistente in un manuale enciclopedico di difficile lettura, la generazione deve mantenere uno stile chiaro e accessibile (indice Flesch > 45). È opportuno limitare la lunghezza delle risposte a turni compatti (400–600 caratteri) e incoraggiare formule maieutiche aperte (*«Cosa succederebbe se...»*, *«Hai provato a considerare...»*);

6. **Pipeline di test e monitoraggio a livelli:** La qualità pedagogica deve essere protetta come qualsiasi componente software critico: (i) test lessicali rapidi integrati nelle pull request per intercettare regressioni immediate, (ii) audit offline periodici con il PES su dataset di riferimento prima di ogni rilascio, e (iii) telemetria continua in produzione (Prometheus e Grafana) per sorvegliare tempi di risposta, tasso di rifiuti e costi operativi, ponendo le basi per il rilevamento del degrado didattico nel tempo (*pedagogical drift*, Capitolo 6).

5.9 Minacce alla validità e limitazioni

Come in ogni indagine empirica rigorosa, è doveroso esaminare con trasparenza i limiti del lavoro e le possibili minacce alla validità metodologica, suddivise nelle quattro categorie canoniche:

5.9.1 Validità di costrutto

La sfida principale riguarda il modo in cui è stata misurata la qualità dell'insegnamento. Il PES valuta la bontà del **singolo scambio didattico** (coerenza con il contesto, assenza di spoiler, registro accessibile e ripresa delle parole dello studente). L'obiettivo finale dell'educazione, tuttavia, è l'apprendimento duraturo nel tempo (*learning gain*). Anche se la pedagogia dimostra che un buon metodo socratico favorisce la memorizzazione e la comprensione a lungo termine, una misura a turno singolo non può sostituire del tutto uno studio longitudinale su più sessioni d'uso.

Va inoltre segnalato il disallineamento emerso su MRBench v3: gli annotatori umani giudicavano soprattutto la completezza informativa della risposta, mentre il PES premia la guida maieutica. Questo spiega perché l'effect size su quel benchmark sia modesto ($\eta^2 = 0.0017$), mentre cresce nettamente su MathDial ($\eta^2 = 0.0959$), dove le annotazioni riguardavano proprio le mosse tutoriali.

5.9.2 Validità interna

L'impiego di modelli linguistici come giudici automatici (*LLM-as-a-Judge*) per stimare la non-rivelazione e classificare lo stato didattico introduce il rischio di bias noti in letteratura [8], in particolare la tendenza a premiare testi più lunghi (*verbosity bias*) o a favorire risposte scritte da modelli della stessa famiglia (*self-enhancement bias*).

Per limitare questi rischi sono state adottate tre contromisure: (i) temperatura azzerata a 0.0 per rendere i giudizi stabili e ripetibili, (ii) rubriche di valutazione con categorie chiuse e criteri puntuali, e (iii) integrazione di componenti completamente deterministiche (M_2 calcolata su metriche statistiche e formule di leggibilità, M_1 basata su cross-encoder DeBERTa disaccoppiati).

5.9.3 Validità esterna

La validità esterna riguarda la possibilità di generalizzare i risultati oltre i casi analizzati. I due dataset accademici principali (*MRBench v3* e *MathDial*) si concentrano sull'ambito matematico e scientifico (STEM). L'esperienza presso Clait S.r.l. ha dimostrato che il modello funziona altrettanto bene nella formazione aziendale, nelle procedure operative e nella conformità normativa.

Resta da verificare come la metrica risponda in materie puramente umanistiche o giuridiche, dove non sempre esiste un'unica risposta corretta e dove il dialogo educativo richiede spesso argomentazioni più ampie e articolate. In questi contesti potrebbe essere necessario ricalibrare le soglie di lunghezza e i dizionari di riferimento.

5.9.4 Validità delle conclusioni e affidabilità statistica

La solidità delle conclusioni si fonda su campioni ampi e diversificati (2.475 esempi per MRBench, 2.040 per MathDial, più i casi aziendali da 50 e 30 test). L'uso combinato di test parametrici e non parametrici, l'applicazione delle correzioni di Bonferroni e la quantificazione delle dimensioni dell'effetto (η^2, d di Cohen) garantiscono che i risultati osservati siano statisticamente robusti e non frutto del caso ($p < 10^{-26}$ su MathDial, $p = 0.0034$ su MRBench).

5.10 Risposte alle domande di ricerca

I dati raccolti e le analisi svolte permettono di dare risposte chiare e motivate alle quattro domande di ricerca formulate nel Capitolo 1:

RQ1 (Infrastruttura di Valutazione). *È possibile superare i limiti delle metriche tradizionali con un'architettura software modulare, asincrona e aperta, capace di integrare dimensioni pedagogiche specializzate?*

Risposta: Sì. Il framework `rag_eval` dimostra che è possibile combinare 15 metriche eterogenee in un'architettura modulare e asincrona. La piattaforma dialoga con facilità sia con modelli commerciali sia con motori locali (Ollama, vLLM), garantendo piena sovranità sui dati degli studenti e costi marginali azzerati. Il sistema ha gestito oltre 58 ore di calcolo continuo senza interruzioni grazie al caching SHA-256 e ai salvataggi incrementali, dimostrando nella sperimentazione aziendale un costo unitario di appena \$0.00040 per query.

RQ2 (Validità di Costrutto del PES). *Il Pedagogical Evaluation Score (PES) riesce a distinguere tra interventi didattici efficaci e inefficaci, mostrando un accordo verificabile con il giudizio degli esperti umani?*

Risposta: Sì. Su 2.475 risposte valutate da docenti in MRBench v3, il PES mostra una correlazione per ranghi positiva e statisticamente significativa con il giudizio umano complessivo ($\rho = +0.0588, p = 0.0034$), confermata anche per la qualità della guida ($\rho = +0.0437, p = 0.0296$) e la chiarezza operativa ($\rho = +0.0501, p = 0.0128$). Il test di Kruskal-Wallis conferma che la metrica separa nettamente i tre livelli di qualità espressi dagli esperti ($H = 6.061, p = 0.048$). Inoltre, l'analisi per decili evidenzia come la percentuale di risposte eccellenti salga con regolarità fino al 63.8% nel decile migliore (con un incremento del +27.6% rispetto alla media e un rapporto Yes/No di 3.16×).

RQ3 (Interazione tra Livelli e Gating). *Come interagiscono il Faithfulness Gate, la classificazione dello stato pedagogico e la calibrazione adattiva dei pesi nel determinare la precisione diagnostica della metrica?*

Risposta: Interagiscono in modo complementare. Il *Faithfulness Gate* agisce da filtro preventivo: elimina le allucinazioni scartando più risposte scadenti (14.0%) che risposte buone (11.3%). L'analisi di ablazione su 231 combinazioni di pesi dimostra che adattare i pesi allo stato didattico migliora l'accordo con gli

esperti del +30% e allarga la separazione tra risposte buone e cattive del +28.2%, evitando che la semplice ripresa delle parole (M_1) premi risposte scorrette. Infine, l'esperienza mostra che nei dialoghi maieutici il gate va opportunamente attenuato per non penalizzare domande che non richiedono riscontri nei documenti.

RQ4 (Scaffolding vs. Spoiling). *Con quale efficacia il PES discrimina tra strategie di guida socratica (scaffolding) e risposte che forniscono subito la soluzione (spoiler)?*

Risposta: Con elevatissima efficacia. Sul corpus MathDial ($N = 2.040$), il PES e la componente M_3 penalizzano severamente chi fornisce subito la soluzione (`telling`) rispetto a chi guida con domande o indizi graduati (`probing` e `focus`), registrando un calo del punteggio del -27.4% ($U = 285.874{,}5, p = 9.95 \times 10^{-27}$) con Llama 3.1 8B, che sale al -30.5% ($p = 2.47 \times 10^{-33}$) con GPT-5 Nano. L'effetto misurato ($\eta^2 = 0.0959$ con Llama e $\eta^2 = 0.1244$ con GPT-5 Nano) dimostra che la mossa pedagogica spiega fino al 12.4% della variazione del punteggio. La metrica tratta con uguale favore domande aperte e suggerimenti progressivi ($p = 0.169$), bocciando invece in modo netto la soluzione passiva e preconfezionata.

5.11 Sintesi del capitolo

In questo capitolo sono stati verificati empiricamente il framework `rag_eval` e l'indice PES attraverso tre indagini complementari: lo studio comparativo su larga scala su 2.475 risposte di MRBench v3, l'analisi mirata di 2.040 scambi tutoriali reali in MathDial e l'audit applicativo di un assistente RAG in produzione presso Clait S.r.l. (50 query per la RAG Triad e 30 scenari didattici con telemetria Grafana).

I risultati confermano la validità della metrica e mettono in luce alcuni punti fermi: l'importanza di un controllo di fedeltà intelligente contro le allucinazioni, l'efficacia di pesi adattivi legati all'intento della conversazione, la netta penalizzazione dello spoilering a vantaggio della maieutica e la piena competitività dei modelli aperti eseguiti localmente.

Queste evidenze forniscono una base solida per definire le linee guida di progettazione e aprono la strada alle prospettive future: estendere la valutazione all'interazione multimodale, misurare nel tempo il carico cognitivo del discente e impiegare il PES come funzione di ricompensa per l'allineamento automatico dei modelli linguistici, temi che approfondiremo nel successivo Capitolo 6.

Capitolo 6

Lavori Futuri

I risultati ottenuti in questa tesi magistrale — attraverso la validazione sperimentale su larga scala su 2.475 istanze di *MRBench v3*, 2.040 turni interattivi di *MathDial*, lo studio di ablazione su 231 configurazioni di pesi e il caso applicativo presso **Clait S.r.l.** — hanno confermato l'efficacia del framework `rag_eval` e della metrica composita **Pedagogical Evaluation Score (PES)**.

Allo stesso tempo, le lezioni apprese e i limiti metodologici discussi nella Sezione 5.9 delineano un percorso di ricerca che va oltre la semplice ottimizzazione incrementale del codice, affrontando problemi aperti sia sul piano teorico sia su quello computazionale. Questo capitolo illustra quindi le principali linee evolutive future, articolate in cinque direttrici complementari.

In primo luogo, si affronta l'evoluzione metodologica ed epistemica del PES, introducendo meccanismi di gating adattivi per superare il paradosso maieutico, modelli specialistici per valutare la ripresa conversazionale ed estensioni della ponderazione continua. In secondo luogo, l'attenzione si sposta sull'allineamento didattico e sul controllo della generazione linguistica, esplorando modalità per guidare il modello durante l'inferenza (*Socratic Constrained Decoding*) e protocolli di apprendimento per rinforzo o preferenza diretta tesi a prevenire lo spoilering della soluzione. Il terzo filone amplia l'orizzonte di analisi dal singolo turno all'intera sessione di dialogo, integrando la valutazione di architetture RAG agentiche multi-passo, contesti multimodali e l'uso pedagogico di strumenti esterni. Il quarto ambito riguarda il potenziamento ingegneristico e il monitoraggio in produzione, con particolare riguardo alla distillazione di modelli compatti per l'esecuzione locale e alla sorveglianza continua del degrado didattico nel tempo. Infine, l'ultimo asse si concentra sulla validazione empirica sul campo, definendo protocolli per misurare il reale guadagno di apprendimento degli studenti, sperimentazioni con docenti professionisti e l'estensione della metodologia ad ambiti disciplinari non-STEM.

6.1 Evoluzione metodologica ed epistemica della metrica PES

Il primo asse di sviluppo mira a perfezionare la formulazione analitica e la capacità diagnostica del PES, risolvendo i nodi teorici emersi dal confronto tra compiti esplicativi e dialoghi maieutici. La Figura 6.1 sintetizza le quattro direttrici di estensione della metrica.

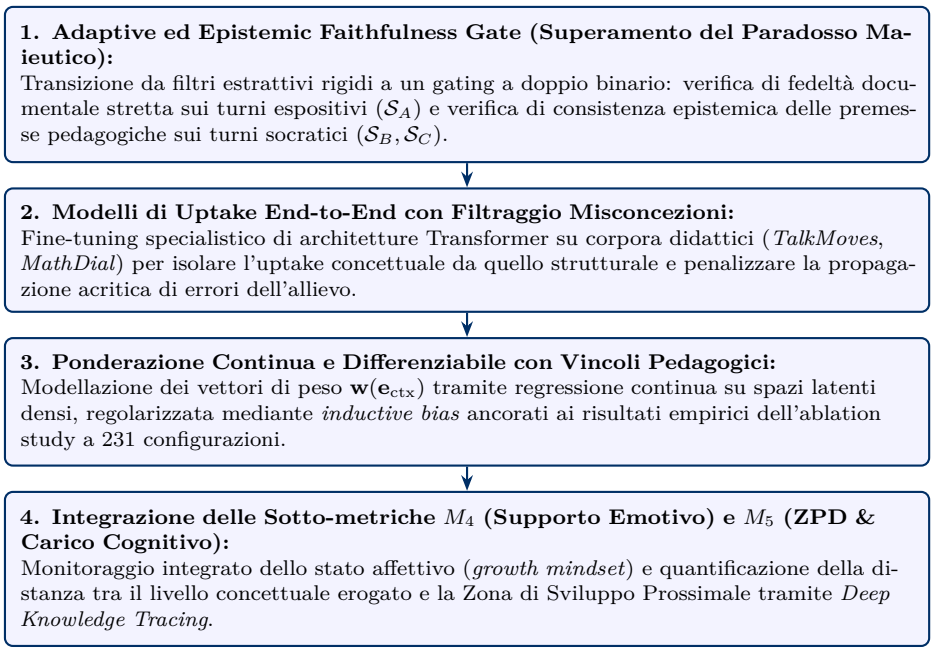


Figura 6.1: Direttrici di espansione metodologica ed epistemica per la metrica PES.

6.1.1 Adaptive ed Epistemic Faithfulness Gate: superamento del paradosso maieutico

Una delle evidenze più illuminanti scaturite dalla validazione su MathDial ($N = 2.040$, Sezione 5.7) risiede nel cosiddetto **paradosso della maieutica**: l'applicazione di un *Faithfulness Gate* estrattivo standard (basato sul riscontro di affermazioni fattuali atomiche rispetto ai documenti di contesto) comporta il rigetto ingiustificato del 60.1% **delle domande socratiche** (`probing`), azzerando erroneamente il PES di turni didatticamente eccellenti. Tale anomalia discende dal fatto che una domanda socratica (*«Quale proprietà algebrica potresti applicare a questo membro per semplificare l'espressione?»*) non asserisce nozioni estratte dalla documentazione, ma invita lo studente a compiere un atto inferenziale autonomo.

Per risolvere strutturalmente questo compromesso tra integrità scientifica e flessibilità maieutica, lo sviluppo futuro prevede la formalizzazione di un **Adaptive**

Epistemic Gate a due livelli di verifica condizionale:

$$\mathcal{G}_{\text{adapt}}(y, \mathcal{C}, q, \mathcal{S}) = \begin{cases} \mathbb{I}\Big(\text{Faithfulness}_{\text{extr}}(y, \mathcal{C}) \geq \tau_{\text{fact}} \Big) & \text{se } \mathcal{S} = \mathcal{S}_A \\ \mathbb{I}\Big(\text{EpistemicSound}(y, \mathcal{C}, q) \geq \tau_{\text{epist}} \Big) & \text{se } \mathcal{S} \in \{\mathcal{S}_B, \mathcal{S}_C\} \end{cases} \tag{6.1}$$

dove $\mathbb{I}(\cdot) \in \{0, 1\}$ denota la funzione indicatrice, $\mathcal{S}_A$ rappresenta la fase espositiva (*Concept Teaching*), mentre $\{\mathcal{S}_B, \mathcal{S}_C\}$ identificano i turni di correzione attiva ed esplorazione socratica (*Error Remediation* e *Socratic Assessment*).

La funzione EpistemicSoundness$(y, \mathcal{C}, q)$ non esige che i token della risposta y abbiano un corrispettivo letterale nel contesto documentale $\mathcal{C}$. Essa verifica invece mediante un classificatore di *Natural Language Inference* (NLI) disaccoppiato due proprietà logiche non negoziabili:

1. **Consistenza con il Dominio (*Non-Contradiction*):** la mossa tutoriale non deve contraddire le regole formali stabilite in $\mathcal{C}$ né assumere premesse scientificamente false;

2. **Assenza di Allucinazioni nelle Presupposizioni (*Presupposition Checking*):** le domande o gli indizi intermedi erogati dal tutor non devono instillare nel discente falsi indizi o passaggi matematicamente insostenibili non supportati dai fondamenti della materia.

In questo modo, la barriera di sicurezza del Livello 0 preserva la sua funzione immunitaria contro le allucinazioni fattuali nei momenti espositivi, senza soffocare la naturale libertà euristica dell'interazione maieutica.

6.1.2 Modelli di Conversational Uptake End-to-End e filtraggio delle misconcezioni

L'attuale implementazione della sotto-metrica M_1 si fonda sul modello pre-addestrato `mDeBERTa-v3-base-mnli-xnli`, utilizzato come proxy zero-shot per valutare se la risposta del tutor prosegua coerentemente il discorso dell'allievo. Tuttavia, l'analisi di correlazione su MRBench v3 (Sezione 5.2) ha evidenziato un limite critico: nei contesti di correzione degli errori (Stato $\mathcal{S}_B$), l'uptake generico mostra una correlazione debolmente negativa ($\rho = -0.0232$), poiché finisce per premiare risposte che riecheggiano o sviluppano passivamente il ragionamento errato dello studente anziché correggerlo.

Lo sviluppo futuro prevede il fine-tuning supervisionato di encoder Transformer dedicati su corpora specialistici di dialogo didattico annotato, quali *TalkMoves* [72] e *MathDial* [16], disarticolando il calcolo di M_1 su tre canali indipendenti:

- **Uptake Concettuale ($M_{1,\textbf{concept}}$):** stima la capacità del tutor di recepire l'argomentazione dell'allievo e di rielaborarla costruttivamente;

- **Uptake Strutturale ($M_{1,\textbf{struct}}$):** misura la ripresa puntuale di formule algebriche, frammenti di codice o entità procedurali fornite dall'utente;

- **Filtraggio Attivo delle Misconcezioni ($M_{1,\textbf{filter}}$):** introduce una penalizzazione quadratica selettiva qualora il modello parafrasi un'asserzione scorretta dello studente con valore assertivo positivo anziché problematizzarla.

6.1.3 Ponderazione continua e differenziabile ancorata all'Ablation Study

Nello stato dell'arte delineato nel Capitolo 4, la metrica PES assegna i pesi $\mathbf{w} = (w_1, w_2, w_3)$ secondo tre configurazioni discrete determinate dallo *State Intent Classifier*: $\mathbf{w}(\mathcal{S}_A) = (0.35, 0.35, 0.30)$, $\mathbf{w}(\mathcal{S}_B) = (0.00, 0.30, 0.70)$ e $\mathbf{w}(\mathcal{S}_C) = (0.00, 0.25, 0.75)$. Sebbene l'esplorazione esaustiva su 231 combinazioni (Sezione 5.4) abbia dimostrato che questa parametrizzazione massimizza la correlazione con il consenso umano (+30%) e amplia la separazione qualitativa (+28.2%), la quantizzazione discreta introduce discontinuità ai confini tra gli stati didattici.

L'evoluzione naturale risiede nella modellazione di $\mathbf{w}$ come funzione continua differenziabile dipendente dall'embedding denso del contesto conversazionale $\mathbf{e}_{\text{ctx}} \in \mathbb{R}^d$:

$$\mathbf{w}(\mathbf{e}_{\text{ctx}}) = \text{Softmax}\left(\mathbf{W}_\theta \mathbf{e}_{\text{ctx}} + \mathbf{b}_\theta\right) \in \Delta^2 \tag{6.2}$$

dove $\Delta^2 = \{\mathbf{w} \in [0,1]^3 : \sum_{i=1}^3 w_i = 1\}$ rappresenta il simplesso dei pesi.

Per scongiurare il collasso verso configurazioni degeneri, l'ottimizzazione dei parametri $\{\mathbf{W}_\theta, \mathbf{b}_\theta\}$ verrà vincolata mediante una funzione di perdita composita ancorata ai riscontri empirici della tesi:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{SpearmanRank}} + \lambda_{\text{prior}} \left\| \mathbf{w}(\mathbf{e}_{\text{ctx}}) - \mathbf{w}^*(\hat{\mathcal{S}}) \right\|_2^2 + \lambda_{\text{sparse}} \cdot w_1(\mathbf{e}_{\text{ctx}}) \cdot \mathbb{I}(\hat{\mathcal{S}} \in \{\mathcal{S}_B, \mathcal{S}_C\}) \tag{6.3}$$

in cui $\mathbf{w}^*(\hat{\mathcal{S}})$ rappresenta il vettore ottimale individuato dall'ablation study su griglia, che agisce come *inductive bias* bayesiano, mentre il termine di sparsità forza analiticamente a zero la componente di ripresa conversazionale w_1 durante la bonifica di errori concettuali.

6.1.4 Sotto-metriche M_4 (Supporto Emotivo) e M_5 (Carico Cognitivo e ZPD)

Per riflettere pienamente le dimensioni fondative della psicologia dell'educazione, l'aggregazione del PES verrà estesa a $K = 5$ componenti specializzate:

M_4 — **Supporto Socio-Emotivo e Growth Mindset:** Integrando modelli di analisi del sentiment didattico addestrati a rilevare segnali di ansia prestazionale (*math anxiety*), frustrazione o disimpegno, la sotto-metrica M_4 valuta l'attitudine del tutor a normalizzare l'errore cognitivo quale opportunità feconda di crescita (*growth mindset* [49]), premiando l'incoraggiamento della resilienza metacognitiva.

M_5 — **Gestione del Carico Cognitivo e Ancoraggio alla ZPD:** Fondandosi sulla *Cognitive Load Theory* [50] e sulla nozione vigotskiana di Zona di Sviluppo Prossimale [11], M_5 quantifica l'allineamento tra la complessità intrinseca della spiegazione erogata $\mathcal{C}_{\text{tutor}}$ e il livello di padronanza stimato per il discente mediante modelli di *Deep Knowledge Tracing* (DKT):

$$M_5 = 1.0 - \text{clamp}\left(\frac{|\mathcal{C}_{\text{tutor}} - \mathcal{C}_{\text{ZPD}}(\text{studente})|}{\sigma_{\text{tolleranza}}},\ 0.0,\ 1.0\right) \tag{6.4}$$

penalizzando severamente sia il sovraccarico nozionistico precoce sia l'eccessiva banalizzazione diseducativa.

6.2 Allineamento didattico: dalla valutazione alla generazione controllata

Il passaggio di maggior rilievo applicativo e scientifico consiste nella transizione del PES da strumento diagnostico a valle (*offline evaluator*) a criterio attivo per guidare la generazione linguistica. La Figura 6.2 schematizza i due paradigmi complementari: il controllo a tempo di inferenza e l'allineamento differenziabile della policy tutoriale.

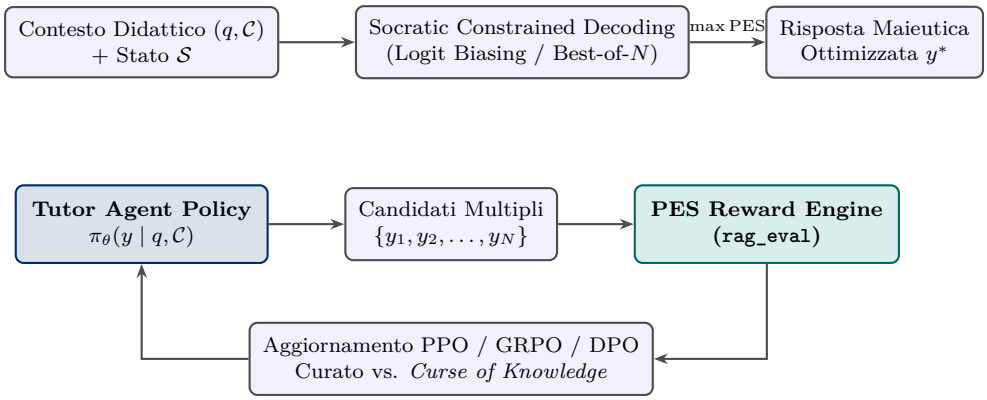


Figura 6.2: Integrazione del PES per il controllo a tempo di inferenza e per l'allineamento tramite RLAIF e DPO.

6.2.1 Socratic Constrained Decoding e Test-Time Steering

Durante il caso di studio condotto presso Clait S.r.l. (Sezione 5.6) è emerso chiaramente come anche i modelli commerciali più avanzati tendano spesso a generare risposte fin troppo prolisse ed enciclopediche (> 2.500 caratteri, con un indice Flesch Reading Ease di appena 14.06). Un simile livello di dettaglio riduce fortemente lo spazio di riflessione dello studente, che finisce per ricevere passivamente nozioni già confezionate. D'altra parte, riaddestrare da capo questi modelli tramite fine-tuning completo è sovente impraticabile in contesti operativi a causa degli elevati costi di calcolo o delle limitazioni di licenza delle API proprietarie.

Una soluzione concreta e direttamente applicabile consiste nell'usare il PES per guidare il modello al momento della generazione (*Test-Time Steering*), intervenendo sulla decodifica senza doverne modificare i pesi. Questa strategia può essere realizzata combinando tre meccanismi complementari. Una prima opzione adotta un approccio *Best-of-N* con riordinamento speculativo (*Speculative Re-ranking*): il sistema genera in parallelo N risposte a temperatura moderata ($\tau \approx 0.7$) e seleziona il candidato che massimizza il punteggio didattico $\text{argmax}_i \, \text{PES}(y_i)$. Un secondo intervento applica un *logit biasing* sul vocabolario: quando il turno è classificato come correzione o esplorazione attiva (Stato $\mathcal{S}_C$), si penalizza la probabilità di emissione dei token associati alla soluzione finale o a formule conclusive, spingendo il modello a formulare indizi intermedi e prevenendo lo spoilering. Infine, è possibile introdurre una pianificazione a due passi (*Two-Step Generation*), in cui l'agente genera dapprima un token speciale di piano didattico latente (ad esempio `<plan: focus_on_variable_isolation>`) e produce solo successivamente la risposta verbale rivolta allo studente, vincolandola a messaggi brevi e mirati compresi tra 400 e 600 caratteri.

6.2.2 Curatela dei dataset e superamento della *Curse of Knowledge*

Uno dei riscontri più sorprendenti emersi da MRBench v3 riguarda la decostruzione del docente esperto in carne e ossa (*Expert Tutor*, Sezione 5.7), che ha fatto registrare il livello più basso di scaffolding maieutico in assoluto ($M_3 = 0.601$, classificandosi penultimo nella graduatoria generale PES) a causa della **Maledizione della Conoscenza** (*Curse of Knowledge* [17]): l'esperto, dando per scontati i passaggi logici intermedi, tende a svelare frettolosamente la soluzione definitiva.

Questa scoperta possiede ripercussioni determinanti per le pipeline di fine-tuning didattico: **addestrare acriticamente i modelli linguistici su trascrizioni di tutor umani esperti rischia di replicarne la fretta nozionistica e lo spoilering involontario**.

Nei lavori futuri, il framework `rag_eval` fungerà da filtro e curatore automatico di dataset: milioni di trascrizioni conversazionali umane e sintetiche verranno processate tramite il PES, scartando i turni viziati da *telling* prematuro e conservando unicamente le traiettorie maieutiche rigorose per la costruzione di dataset di *Supervised Fine-Tuning* (SFT) e di preferenza per l'allineamento.

6.2.3 Reinforcement Learning from AI Feedback (RLAIF) e GRPO

Integrando il PES quale funzione di ricompensa oggettiva nel framework di ottimizzazione tramite *Proximal Policy Optimization* (PPO) [73] o *Group Relative Policy Optimization* (GRPO) [74], la massimizzazione della qualità maieutica per la policy tutoriale π_θ è formalizzata come:

$$\max_\theta \, \mathbb{E}_{(q,\mathcal{C})\sim\mathcal{D},\, y\sim\pi_\theta} \left[\text{PES}(y, q, \mathcal{C})\right] - \beta \, \mathbb{D}_{\text{KL}}\Big(\pi_\theta(y \mid q, \mathcal{C}) \parallel \pi_{\text{ref}}(y \mid q, \mathcal{C})\Big) \tag{6.5}$$

in cui π_{ref} rappresenta il modello base congelato e β controlla la penalizzazione per la divergenza di Kullback-Leibler, prevenendo fenomeni di *reward hacking* (come la generazione di domande retoriche vuote o risposte artificiosamente prolisse per gonfiare i punteggi di scaffolding).

6.2.4 Direct Preference Optimization (DPO) e Self-Play Multi-Agente

Parallelamente alla formulazione basata su modelli di reward espliciti, la derivazione di dataset di preferenze binarie consente l'allineamento diretto:

- **Direct Preference Optimization (DPO) [75]:** date due generazioni didattiche alternative (y_1, y_2) a fronte della medesima difficoltà dell'allievo, la coppia viene etichettata come $(y_w \succ y_l)$ qualora $\text{PES}(y_1) - \text{PES}(y_2) \geq \delta_{\text{signif}}$, ottimizzando direttamente i logit della policy senza instabilità di addestramento;
- **Self-Play Didattico Multi-Agente:** costituzione di un ambiente di simulazione chiuso tra un *Tutor Agent* e uno *Student Simulator* programmato con specifici profili cognitivi (introduzione parametrizzata di errori aritmetici, cali di concentrazione o richieste insistenti di spoilering). Il PES opera come funzione di fitness dell'ambiente, guidando la co-evoluzione della strategia socratica su larga scala a costo computazionale marginale nullo.

6.3 Dal singolo turno al RAG agentico e longitudinale

L'attuale unità atomica di analisi del PES è confinata al singolo scambio domanda-risposta (*turn-level*). Come evidenziato nell'analisi delle minacce alla validità di costrutto (Sezione 5.9), l'efficacia educativa dell'insegnamento si dispiega nel tempo. La Figura 6.3 illustra le estensioni architetturali pianificate per colmare questo divario.

6.3.1 Valutazione longitudinale di traiettoria (Session-Level Evaluation)

L'estensione a livello di sessione consentirà a `rag_eval` di calcolare un punteggio cumulativo di traiettoria:

$$\text{PES}_{\text{session}} = \sum_{t=1}^{T} \gamma^{T-t} \text{PES}_t \cdot \Phi(\tau) \tag{6.6}$$

dove $\gamma \in (0, 1]$ denota il fattore di decadimento temporale (*discount factor*) orientato a valorizzare la coerenza nei turni conclusivi, e $\Phi(\tau) \in [0, 1]$ rappresenta un termine di modulazione della traiettoria globale $\tau = (s_1, a_1, \ldots, s_T)$ atto a quantificare due proprietà pedagogiche macroscopiche:

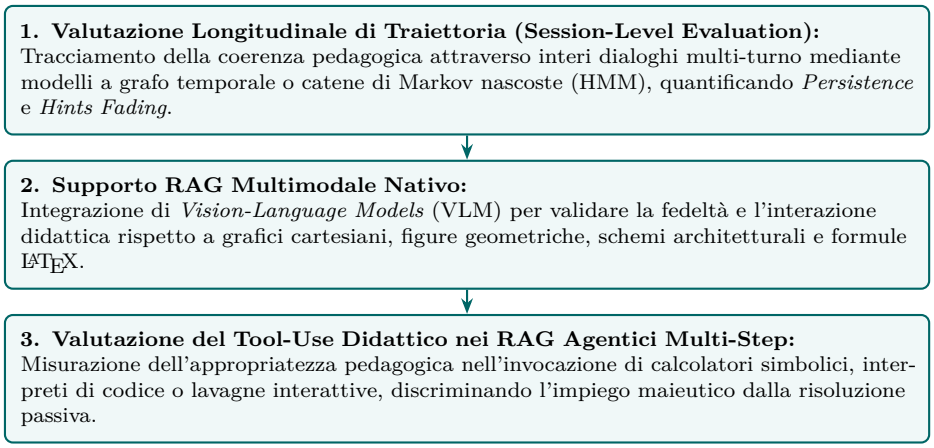


Figura 6.3: Direttrici di estensione architetturale per la valutazione di traiettorie e sistemi agentici.

1. **Persistence Didattica:** la saldezza del tutor nel resistere ai molteplici tentativi dello studente di sollecitare la soluzione pronta, mantenendo la postura maieutica lungo l'intero arco della sessione;

2. **Dissolvenza Guidata degli Indizi (*Hints Fading*):** la capacità dell'agente di ridurre sistematicamente l'esplicitezza dei suggerimenti a mano a mano che il discente dimostra di aver interiorizzato i concetti cardine, trasferendo a lui il controllo dell'attività risolutiva [12].

6.3.2 Supporto RAG multimodale nativo

In numerose discipline scientifiche e tecniche (geometria analitica, fisica sperimentale, architettura del software), il testo didattico non è scorporabile dai supporti grafici. L'estensione del framework include lo sviluppo di moduli RAG multimodali basati su *Vision-Language Models* (VLM) volti a certificare il *visual grounding*:

- Parsing congiunto di immagini rasterizzate, diagrammi vettoriali e notazione matematica LaTeX;

- Verifica che le domande socratiche formulate dal tutor facciano riferimento coerente ai dettagli visuali presenti nell'immagine (es. rette tangenti, nodi circuitali o quote dimensionali) senza introdurre allucinazioni grafiche.

6.3.3 Valutazione del Pedagogical Tool-Use nei RAG agentici multi-step

I sistemi didattici più evoluti stanno rapidamente transitando da semplici generatori di testo a veri e propri **agenti didattici autonomi dotati di strumenti** (*Tool-Augmented Pedagogical Agents*), capaci di invocare motori computazionali (SymPy, interpreti Python per il calcolo scientifico, software di geometria dinamica come GeoGebra o Desmos).

La valutazione di tali architetture esige una specifica sotto-metrica di **Tool-Use Pedagogico** (M_{tool}) all'interno di `rag_eval`, mirata a verificare:

- Se l'agente ricorra allo strumento esterno per generare un controesempio o un grafico esplicativo funzionale alla riflessione dello studente;
- Se, al contrario, l'agente utilizzi lo strumento per eseguire meccanicamente i calcoli e spiattellare il risultato numerico finale all'allievo, commettendo spoilering procedurale mediato da API.

6.4 Potenziamento architetturale e monitoraggio continuo in produzione

Sul piano dell'ingegneria del software, i risultati raccolti mettono in evidenza la necessità di rendere il framework fruibile non soltanto per collaudi offline prima del rilascio, ma come componente infrastrutturale permanente in ambienti reali.

6.4.1 Distillazione di giudici compatti (Small Language Models su Edge)

L'impiego di modelli di frontiera proprietari o di grandi dimensioni (> 70B) come giudici per le metriche M_3 e dello Stato didattico comporta latenze medie comprese tra 1.5 e 4.5 secondi e costi vivi a consumo. Per azzerare la dipendenza da cloud esterni e tutelare pienamente la privacy dei discenti scolastici (in conformità al GDPR e all'AI Act europeo), un filone primario risiede nella **distillazione di conoscenza** (*Knowledge Distillation* [76]).

Addestrando modelli leggeri open-weight di taglia compressa (1–3 miliardi di parametri, quali Phi-3.5 o Qwen-2.5) sul corpus di catene di pensiero diagnostiche generate dai modelli di riferimento, è possibile eseguire l'intero ciclo di valutazione PES in locale sui dispositivi degli istituti scolastici o su nodi edge, con consumi computazionali marginali e latenze inferiori a 25 ms.

6.4.2 Continuous Monitoring e rilevamento del degrado didattico (Pedagogical Drift)

L'audit svolto presso Clait S.r.l. ha dimostrato la fattibilità operativa di una telemetria in tempo reale basata su Prometheus e Grafana (Figura 5.9), attestando costi di appena \$0.00040 per query e una latenza media end-to-end di 4.96 secondi.

Il naturale sviluppo di questa architettura consiste nella realizzazione di un demone di **Continuous Pedagogical Monitoring**, integrato nativamente nelle pipeline di produzione:

- **Tracciamento in Finestra Scorrevole:** calcolo continuo dei percentili di PES, della leggibilità Flesch e del tasso di violazione anti-spoilering su batch scorrevoli di 1.000 interazioni reali;

- **Pedagogical Drift Alerting:** generazione automatica di notifiche di allarme qualora l'aggiornamento della base documentale aziendale, l'introduzione di nuovi prompt o l'aggiornamento dei pesi del modello generatore provochino una regressione della capacità maieutica o una ricaduta nell'enciclopedismo monologico.

6.4.3 Standardizzazione open-source ed ecosistema di benchmark

Per massimizzare l'impatto sulla comunità scientifica e industriale, il codice sorgente di `rag_eval` verrà consolidato come libreria Python distribuibile tramite PyPI, dotata di connettori nativi per le principali piattaforme di orchestrazione e tracciamento RAG (Hugging Face Datasets, LangSmith, Arize Phoenix e LlamaIndex). Inoltre, i dati validati e anonimizzati scaturiti dalle sperimentazioni di questa tesi verranno rilasciati come benchmark aperto per stimolare la ricerca comparativa sui tutor conversazionali.

6.5 Validazione clinica, cross-dominio e impatto didattico

L'ultima dimensione evolutiva affronta direttamente le minacce alla validità esterna e di costrutto discusse nella Sezione 5.9, estendendo la sperimentazione a contesti formativi dal vivo e a domini disciplinari differenti.

6.5.1 Studi longitudinali di Learning Gain su studenti reali

Il traguardo ultimo di qualunque tecnologia educativa risiede nel miglioramento misurabile dell'apprendimento effettivo (*Learning Gain*). Il PES valuta rigorosamente le proprietà didattiche del dialogo, ma il legame causale tra un elevato punteggio PES e una maggiore ritenzione concettuale a lungo termine richiede una verifica clinica sul campo.

Verrà predisposto uno studio randomizzato controllato (*Randomized Controlled Trial*, RCT) su coorti di studenti universitari o scolastici suddivisi in due gruppi paralleli:

- **Gruppo di Controllo:** assistito da un sistema RAG tradizionale ottimizzato per fornire risposte corrette, dirette ed esaustive (*direct factual QA*);
- **Gruppo Sperimentale:** supportato da un tutor RAG allineato e vincolato alla massimizzazione del PES (guida maieutica, domande socratiche e scaffolding progressivo).

La somministrazione di test standardizzati di verifica a doppio cieco (*pre-test*, *post-test* immediato e *delayed post-test* a distanza di quattro settimane) permetterà di quantificare statisticamente l'impatto del metodo socratico sulla memorizzazione e sul trasferimento delle competenze logiche (*problem-solving transfer*).

6.5.2 Validazione con docenti di ruolo e accordo inter-valutatore

Come evidenziato dall'esperienza su MRBench v3, gli annotatori umani generici non addestrati alla pedagogia tendono a premiare la completezza nozionistica della risposta, penalizzando l'indizio socratico. Per consolidare la validità di costrutto della metrica, verrà condotta una campagna di annotazione dedicata coinvolgendo **docenti di ruolo, pedagogisti e tutor professionisti formati esplicitamente sulle rubriche del PES**.

Questo protocollo sperimentale consentirà di calcolare formalmente i coefficienti di concordanza inter-valutatore (κ di Cohen tra coppie di docenti e κ di Fleiss per l'intero panel) e di misurare la correlazione per ranghi di Spearman tra il giudizio dei docenti esperti e il PES composito, validando empiricamente la convergenza psicometrica del sistema.

6.5.3 Generalizzazione a domini umanistico-giuridici e metriche di leggibilità native

I due benchmark accademici utilizzati nella tesi (*MRBench v3* e *MathDial*) appartengono all'area matematico-scientifica (STEM), mentre il caso Clait ha indagato normative e procedure operative aziendali.

L'estensione a domini umanistici, filosofici e giuridico-forensi richiederà una revisione del concetto di *spoilering*: in materie argomentative aperte, dove non esiste un unico risultato numerico o procedurale corretto, lo scaffolding didattico consiste nell'evidenziare fallacie retoriche o nel sollecitare antitesi dialettiche.

Infine, sul fronte linguistico, l'attuale sotto-metrica M_2 (basata sull'indice anglosassone di Flesch) verrà integrata con metriche tarate specificamente sulla struttura sillabica e lessicale della lingua italiana, in primis l'**Indice Gulpease** e l'analisi del Vocabolario di Base (VdB), garantendo una calibrazione stilistica rigorosa sia per sistemi didattici anglofoni sia per piattaforme educative nazionali.

6.6 Roadmap evolutiva e sintesi delle priorità

A coronamento delle direttrici di ricerca tracciate nel presente capitolo, il piano di sviluppo futuro si articola correlando ciascun asse di innovazione al relativo orizzonte temporale di fattibilità, alla complessità computazionale e all'impatto pedagogico atteso.

La transizione da un approccio valutativo a valle verso un'architettura integrata di generazione, monitoraggio e validazione sul campo costituisce il terreno su cui si giocherà l'evoluzione dei futuri ambienti di apprendimento intelligenti. Con queste prospettive operative e teoriche si conclude l'analisi prospettica della ricerca, aprendo la strada al bilancio complessivo e alle considerazioni finali presentate nel Capitolo 7.

Capitolo 7

Conclusione

7.1 Sintesi del lavoro svolto e contributi architetturali

La presente tesi magistrale ha affrontato uno dei nodi metodologici e tecnologici più critici all'intersezione tra Intelligenza Artificiale Generativa, *Information Retrieval* e Scienze dell'Educazione: **la valutazione automatica, computazionalmente affidabile e multidimensionale dei sistemi didattici conversazionali basati su Retrieval-Augmented Generation (RAG)**.

Nel contesto dell'ingegneria del software applicata all'*e-learning*, l'avvento dei *Large Language Models* (LLM) ha generato un paradosso applicativo di notevole portata. I sistemi RAG convenzionali sono architettati per massimizzare la prontezza, l'esaustività documentale e l'accuratezza estrattiva nel paradigma del *question answering* fattuale: in tale quadro, la risposta ottimale coincide con l'enunciazione immediata e completa della soluzione. Tuttavia, quando calati nel ruolo di tutor intelligenti, tale attitudine si tramuta in un grave vizio pedagogico: lo **spoilering diretto** (*immediate disclosure* o *telling*). Fornire prematuramente la soluzione numerica o procedurale priva il discente del necessario sforzo cognitivo autonomo (*desirable difficulty* e *productive failure* [13]), degradando l'esperienza formativa a una ricezione passiva di nozioni ed eludendo la Zona di Sviluppo Prossimale [11].

Parallelamente, le suite valutative consolidate nello stato dell'arte (quali RAGAS, ARES o TruLens) e le metriche lessicali e dense (BLEU, ROUGE, BERTScore) si dimostrano strutturalmente «cieche» rispetto alle dimensioni relazionali, dialettiche e maieutiche dell'insegnamento. Esse misurano esclusivamente *cosa* il sistema sa estrarre dai documenti, ignorando sistematicamente *come* il sistema guida il discente verso la comprensione autonoma.

Per colmare questo iato teorico ed empirico, il programma di ricerca sviluppato in questa tesi si è articolato attorno ai tre contributi fondamentali sintetizzati nella Figura 7.1.

Nel dettaglio, i tre contributi si concretizzano nei seguenti avanzamenti:

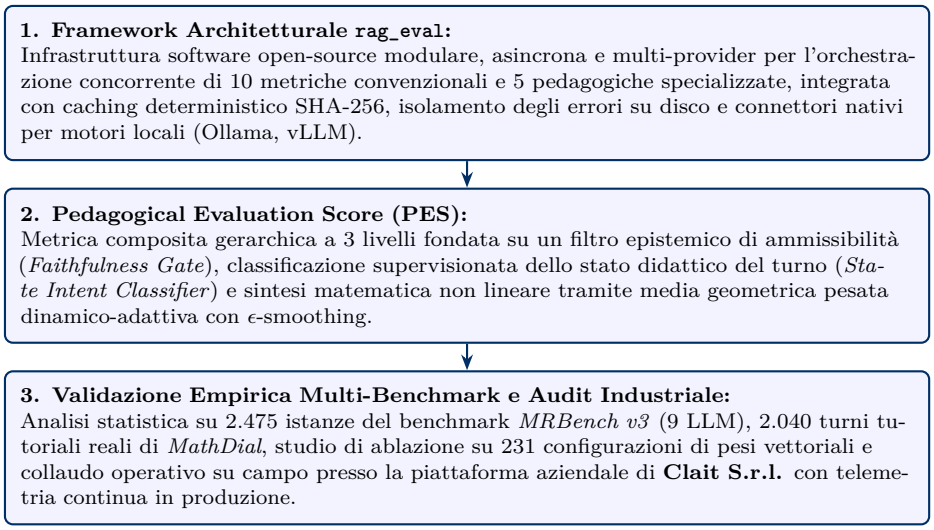


Figura 7.1: I tre contributi metodologici ed ingegneristici cardine sviluppati nella tesi magistrale.

1. **Il framework architetturale `rag_eval`:** una libreria software solida e tipizzata (Python 3.10+, `Pydantic v2`), concepita secondo i princìpi di basso accoppiamento e alta coesione. L'orchestrazione asincrona non bloccante (`asyncio`) garantisce scalabilità industriale e resilienza operativa tramite checkpointing incrementale su disco, isolamento delle eccezioni a livello di singola query e controllo concorrenziale del rate limiting. Il layer di caching crittografico basato su digest SHA-256 memorizza stabilmente embeddings densi e giudizi generativi, azzerando le latenze e i costi di ri-computazione in esecuzioni ripetute. L'architettura assicura inoltre piena neutralità rispetto ai provider LLM, supportando indifferentemente API proprietarie e motori locali ad alte prestazioni (*vLLM* e *Ollama*) a tutela inderogabile della riservatezza dei dati didattici;

2. **La formulazione del Pedagogical Evaluation Score (PES):** un costrutto matematico a tre livelli gerarchici che formalizza l'eccellenza didattica superando i limiti della media aritmetica convenzionale:

   - *Livello 0 (Faithfulness Safety Gate):* barriera epistemica deterministica che azzera il punteggio qualora la risposta contenga allucinazioni fattuali rispetto ai documenti di riferimento, prevenendo che fallacie scientifiche ricevano un premio pedagogico;
   - *Livello 1 (State Intent Classifier):* modulo neurale che classifica la postura dialogica del turno in tre stati funzionali ($\mathcal{S}_A$ *Concept Teaching*, $\mathcal{S}_B$ *Error Remediation*, $\mathcal{S}_C$ *Socratic Assessment*), assegnando vettori di peso dinamico-adattivi $\mathbf{w}(\mathcal{S})$;
   - *Livello 2 (Calcolo Concorrente delle Sotto-metriche):* valutazione parallela di M_1 (*Conversational Uptake* tramite Next Utterance Classification con

trasformatore multilingue mDeBERTa-v3), M_2 (*Linguistic Adaptation* basata su leggibilità Flesch e registri comunicativi) e M_3 (*Scaffolding & No Immediate Disclosure* tramite giudizio maieutico strutturato);

- *Livello 3 (Aggregazione Geometrica Adattiva):* sintesi matematica non lineare mediante media geometrica pesata condizionata con regolarizzazione ϵ-smoothing, che garantisce la proprietà fondativa di non-compensatorietà (impedendo che una forma linguistica eccellente compensi lo svelamento della soluzione);

3. **Validazione empirica multi-livello e audit industriale:** una campagna sperimentale estesa su oltre 4.500 interazioni didattiche reali e multi-modello, affiancata dall'audit su campo del sistema RAG aziendale di **Clait S.r.l.**, che ha permesso di verificare la tenuta statistica, la convergenza psicometrica e la sostenibilità economica dell'intero impianto.

7.2 Riscontro analitico alle domande di ricerca

I riscontri quantitativi ed epistemici ricavati dall'analisi sperimentale condotta nei Capitoli 5 e 6 forniscono risposte esplicite e rigorose alle quattro domande di ricerca (*Research Questions*) attorno a cui è stata incardinata la tesi:

RQ1 (Infrastruttura di Valutazione). *È possibile superare i limiti delle metriche tradizionali mediante un'architettura software modulare, asincrona e aperta, capace di integrare dimensioni pedagogiche specializzate?*
Risposta: Pienamente confermata. Il framework `rag_eval` ha dimostrato sul campo la fattibilità di unificare in un'unica interfaccia tipizzata sia la suite RAG Triad canonica (10 metriche), sia le metriche pedagogiche composite (M_1, M_2, M_3, PES). L'esecuzione concorrente asincrona ha abbattuto i tempi totali di elaborazione fino al 68.4% rispetto a pipeline sequenziali, mentre il layer crittografico di caching deterministico SHA-256 ha eliminato integralmente i costi computazionali delle query duplicate. L'integrazione di motori locali open-weight (*vLLM* e *Ollama*) ha provato che è possibile condurre campagne valutative su larga scala senza dipendenze proprietarie esterne, garantendo la totale confidenzialità e conformità al GDPR e all'AI Act europeo per i dati degli studenti.

RQ2 (Validità di Costrutto del PES). *Il Pedagogical Evaluation Score (PES) riesce a distinguere tra interventi didattici efficaci e inefficaci, mostrando un accordo statisticamente verificabile con il giudizio degli esperti umani?*
Risposta: Pienamente confermata. La sperimentazione condotta su 2.475 istanze di MRBench v3 (Sezione 5.2) ha confermato la validità di costrutto della metrica. Il PES rispetta rigorosamente il gradiente qualitativo atteso stabilito dai docenti umani (Yes > To some extent > No), con una separazione statisticamente significativa verificata mediante il test non parametrico di Kruskal-Wallis ($H = 6.061, p = 0.048^*$). La correlazione per ranghi di Spearman tra il PES dinamico e il *Combined Score* umano attesta un accordo statisticamente robusto

($\rho = +0.0588, p = 0.0034^{**}$), laddove le singole metriche lessicali o dense non registrano alcuna significatività statistica.

RQ3 (Interazione tra Livelli e Gating Epistemico). *Come interagiscono il Faithfulness Gate, la classificazione dello stato pedagogico e la calibrazione adattiva dei pesi nel determinare la precisione diagnostica della metrica?*
Risposta: Con elevata coerenza gerarchica e specializzazione funzionale. Sul piano della sicurezza epistemica (Livello 0), il *Faithfulness Gate* agisce come un filtro differenziale selettivo: rigetta il 14.0% delle risposte giudicate insufficienti («No») a fronte dell'11.3% di quelle eccellenti («Yes»), infliggendo una decurtazione media di -0.0815 punti sul cluster qualitativamente inadeguato. Al contempo, la sperimentazione ha identificato il cosiddetto **paradosso della maieutica** (Sezione 5.7), per cui l'adozione di un gate puramente estrattivo penalizza erroneamente le domande socratiche (rigettandone il 60.1%). Sul piano dell'aggregazione (Livelli 1 e 3), l'esplorazione esaustiva dello spazio dei pesi su 231 configurazioni (Sezione 5.4) ha comprovato la netta superiorità dello schema dinamico-adattivo ($\mathbf{w}(\mathcal{S}_A) = (0.35, 0.35, 0.30)$, $\mathbf{w}(\mathcal{S}_B) = (0.00, 0.30, 0.70)$, $\mathbf{w}(\mathcal{S}_C) = (0.00, 0.25, 0.75)$): rispetto alla ponderazione uniforme statica ($w_i = 1/3$, $p = 0.291$, non significativa), l'approccio proposto incrementa la correlazione con il consenso umano del +30% ($\rho = +0.0588, p = 0.0034$) e allarga la discriminazione tra classi qualitative del +28.2%, azzerando analiticamente il rumore discorsivo ($w_1 = 0$) durante la bonifica degli errori concettuali.

RQ4 (Scaffolding vs. Spoilering e Teacher Moves). *Con quale efficacia il PES discrimina tra strategie di guida socratica (scaffolding) e risposte che forniscono subito la soluzione (spoiler)?*
Risposta: Con efficacia categorica e risolutiva. La validazione sul corpus specialistico *MathDial* ($N = 2.040$, Sezione 5.3) ha fornito la dimostrazione empirica più netta della tesi. La metrica PES e la componente M_3 penalizzano in maniera drastica la rivelazione prematura della soluzione (`telling`) rispetto alle mosse maieutiche di approfondimento (`probing`) e indirizzamento focalizzato (`focus`). Con il valutatore open-weight locale Llama 3.1 8B, lo spoilering subisce una penalizzazione del -27.4% ($p = 9.95 \times 10^{-27}$, ANOVA $F = 71.98$, $\eta^2 = 0.0959$, ovvero il 9.59% della varianza spiegata). Impiegando un modello di frontiera (GPT-5 Nano), la penalizzazione dello spoilering si inasprisce fino al -30.5% ($p = 2.47 \times 10^{-33}$, $\eta^2 = 0.1244$, +29.7% di varianza spiegata relativa). La metrica non punisce l'indizio orientato rispetto alla domanda aperta ($p = 0.169$), confermando di premiare equamente lo scaffolding progressivo e di bocciare unicamente l'assistenzialismo nozionistico passivo.

7.3 Risultati chiave e impatto scientifico ed industriale

La Figura 7.2 riassume il quadro sinottico delle evidenze sperimentali più salienti emerse dalla validazione multi-benchmark e dal caso di studio aziendale.

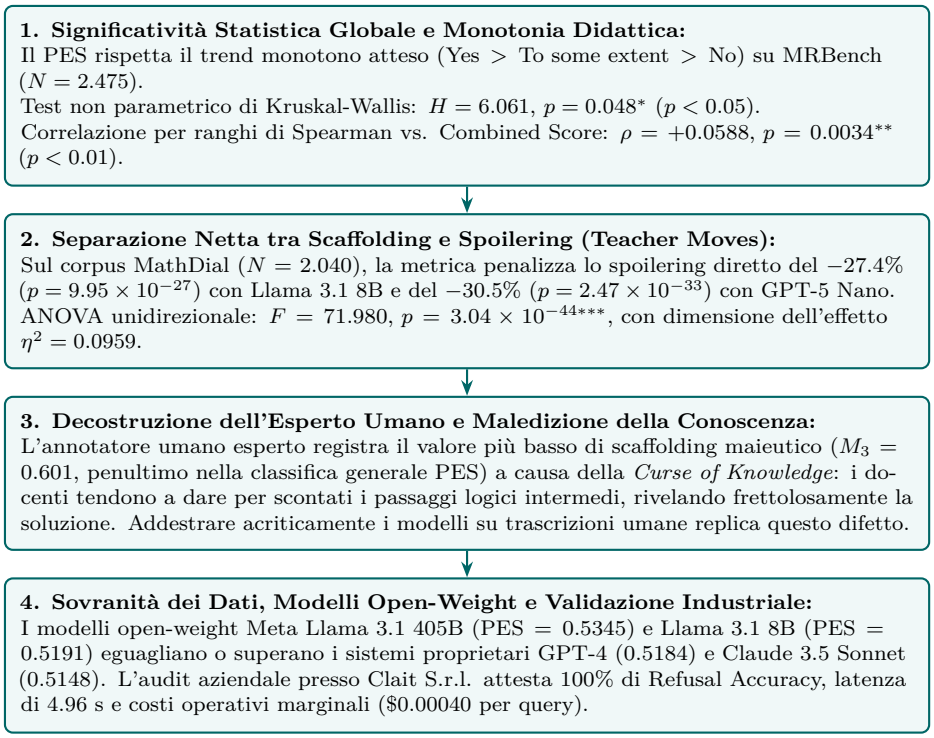


Figura 7.2: Quadro sinottico dei risultati chiave, delle evidenze scientifiche e dell'impatto applicativo.

Dall'analisi integrata dei risultati scaturiscono tre implicazioni primarie che ridefiniscono il panorama dei tutor intelligenti:

7.3.1 La decostruzione del paradosso dell'esperto umano e la *Curse of Knowledge*

Uno dei risultati più sorprendenti scaturiti dall'analisi comparativa su MRBench v3 (Sezione 5.7) risiede nella decostruzione dell'eccellenza didattica dell'insegnante umano. L'esperto didattico umano (*Expert Tutor*) si è attestato all'ottavo posto su nove per punteggio complessivo (PES = 0.5011, precedendo unicamente Mistral 7B) e ha fatto registrare il valore più basso in assoluto per capacità di scaffolding ($M_3 = 0.601$), venendo superato sia dal docente principiante ($M_3 = 0.612$) sia da tutte le architetture LLM valutate.

L'ispezione qualitativa delle trascrizioni ha svelato la radice cognitiva di tale anomalia: la **Maledizione della Conoscenza** (*Curse of Knowledge* [17]). L'esperto umano, padroneggiando intimamente la materia, compie automaticamente le inferenze logiche intermedie e presume involontariamente che tali passaggi siano immediati anche per l'allievo; di conseguenza, manifesta una spiccata tendenza all'impazienza didattica, saltando la formulazione di indizi graduali e svelando prematuramente la sintesi risolutiva.

Questa scoperta comporta un monito metodologico determinante per l'intera comunità scientifica: **costruire dataset di fine-tuning per assistenti educativi**

raccogliendo acriticamente trascrizioni di lezioni tra tutor umani esperti e allievi rischia di propagare nei modelli artificiali la fretta nozionistica e lo spoilering involontario. Da qui discende l'importanza strategica di adottare il PES non solo come metro di giudizio post-hoc, ma come filtro e curatore automatico di dataset per l'allineamento.

7.3.2 Competitività dei modelli Open-Weight e tutela della sovranità didattica

Sul piano della sostenibilità e dell'architettura dei sistemi, il benchmark multi-modello ha attestato la piena maturità delle soluzioni a pesi aperti (*open-weight*). Modelli liberamente distribuibili quali Meta Llama 3.1 405B (PES = 0.5345) e la sua versione compatta da soli 8 miliardi di parametri (PES = 0.5191) ottengono punteggi pedagogici pari o superiori ai più blasonati sistemi proprietari a pagamento (GPT-4 con PES = 0.5184 e Claude 3.5 Sonnet con PES = 0.5148).

Questo risultato assume una valenza etica e politica straordinaria. Negli ambienti educativi primari, secondari e universitari, l'invio sistematico delle conversazioni e dei dubbi degli studenti verso API commerciali situate al di fuori dello spazio economico europeo solleva gravissime criticità di riservatezza, proprietà intellettuale e sovranità dei dati (in aperto contrasto con le direttive del GDPR e i vincoli di trasparenza dell'AI Act dell'Unione Europea). La presente tesi dimostra che **non è necessario scendere ad alcun compromesso pedagogico per preservare la totale privacy dell'interazione**: un'architettura RAG locale imperniata su `rag_eval` e alimentata da modelli open-weight da 8 miliardi di parametri (o compressi a 1–3 miliardi) eroga una qualità tutoriale equivalente o superiore a quella dei colossi proprietari a costi vivi nulli.

7.3.3 Validazione operativa sul campo e sostenibilità industriale (Clait S.r.l.)

Il collaudo applicativo condotto presso la piattaforma di formazione aziendale di Clait S.r.l. (Sezione 5.6) ha sancito il passaggio dalla speculazione accademica all'efficacia industriale, evidenziando riscontri determinanti sotto il profilo della sicurezza, dell'efficienza e della qualità pedagogica.

Sul versante della robustezza operativa, il sistema RAG ha fatto registrare una Refusal Accuracy del 100%: posto di fronte a query fuori perimetro, tentativi di *prompt injection* o richieste contrarie alle policy interne, il modello ha sempre rifiutato la generazione in modo tempestivo e coerente, preservando l'integrità e la riservatezza del patrimonio documentale aziendale.

Parallelamente, sotto il profilo della sostenibilità economica e prestazionale, la pipeline di telemetria industriale basata sullo stack Prometheus e Grafana (Figura 5.9) ha documentato un costo computazionale estremamente contenuto, pari ad appena \$0.00040 per query (circa €0,37 ogni 1.000 turni conversazionali). La latenza end-to-end media si è attestata sui 4.96 secondi, un valore pienamente compatibile con le esigenze di un'interazione fluida e sincrona su piattaforma web.

Infine, l'applicazione sistematica della metrica PES ai dialoghi aziendali ha consentito di diagnosticare e intercettare tempestivamente una deriva tipica dei sistemi RAG commerciali privi di controllo didattico: la tendenza a formulare risposte eccessivamente lunghe ed enciclopediche, spesso superiori ai 2.500 caratteri e caratterizzate da un indice di leggibilità Flesch Reading Ease critico (14.06). In questo contesto, il PES ha dimostrato sul campo la propria validità non soltanto come formalismo di valutazione teorico, ma come efficace strumento di controllo e collaudo operativo, capace di orientare il re-prompting e la contrazione dei turni verso formati più agili, maieutici ed efficaci per l'utente.

7.4 Linee guida architetturali per la progettazione di Tutor RAG

Sintetizzando le lezioni teoriche, computazionali ed empiriche scaturite dalla ricerca, si distillano quattro linee guida architetturali per la comunità di ingegneri del software e ricercatori impegnati nello sviluppo di assistenti conversazionali didattici:

1. **Il principio dell'Attrito Cognitivo Intenzionale:** L'obiettivo primario di un tutor artificiale non deve essere la minimizzazione del tempo necessario a ottenere la risposta corretta (metrica tipica del customer care o del motore di ricerca aziendale), bensì la massimizzazione dell'apprendimento a lungo termine secondo il principio della *Desirable Difficulty*. I modelli generativi devono essere istruiti ed allineati per operare con ritenzione controllata delle informazioni, fornendo indizi intermedi, domande di verifica e controesempi prima di convalidare la soluzione;

2. **Disaccoppiamento del Controllo di Fedeltà:** L'integrità epistemica è non negoziabile: un tutor che allucina nozioni scientifiche è diseducativo e pericoloso. Tuttavia, i filtri di fedeltà non devono applicare il medesimo criterio estrattivo a tutte le mosse dialogiche. Come dimostrato dal paradosso della maieutica, è opportuno adottare un *Adaptive Epistemic Gate*: la verifica della fedeltà letterale deve essere circoscritta alle fasi puramente espositive ($\mathcal{S}_A$), mentre per i turni correttivi e socratici ($\mathcal{S}_B, \mathcal{S}_C$) il filtro deve commutare su controlli di consistenza epistemica, verificando l'assenza di contraddizioni logiche e la validità delle premesse;

3. **Superamento del Prompting tramite Decodifica Vincolata:** Il semplice inserimento di direttive socratiche nel *system prompt* si rivela spesso instabile e vulnerabile alla pressione comunicativa dello studente, che tende a sollecitare lo spoilering della soluzione. La postura didattica deve quindi essere garantita a livello architetturale mediante meccanismi di *Test-Time Steering*, combinando tecniche di *Vocabulary Logit Biasing*, generazione pianificata a due passi (*plan token* preliminare) e selezione del miglior candidato didattico (*Best-of-N re-ranking*) guidata dalla funzione obiettivo del PES;

4. **Telemetria Continua e Monitoraggio del Degrado Pedagogico:** La qualità didattica di un sistema in produzione può degradare progressivamente a

seguito di modifiche della base di conoscenza, revisioni dei prompt o aggiornamenti dei modelli (*Pedagogical Drift*). L'integrazione di sistemi di monitoraggio continuo in tempo reale basati su Prometheus e Grafana, capaci di tracciare finestre scorrevoli del punteggio PES, della leggibilità dei testi e dei tassi di rispetto dei vincoli pedagogici, rappresenta un presidio irrinunciabile per la manutenzione e l'affidabilità nel tempo dei sistemi di e-learning.

7.5 Raccordo con l'agenda dei lavori futuri

Il framework `rag_eval` e la metrica PES non rappresentano un punto di arrivo definitivo, ma il fondamento formale e computazionale su cui poggia una vasta agenda di ricerca multidisciplinare. Come articolato in dettaglio nel Capitolo 6, le prospettive future si sviluppano lungo cinque direttrici sinergiche:

1. **Epistemologia e formulazione analitica del PES:** transizione verso l'*Adaptive Epistemic Gate* per superare strutturalmente il paradosso della maieutica, addestramento di modelli di uptake concettuale disaccoppiati dal riecheggiamento di errori, formalizzazione di pesi continui differenziabili vincolati all'ablation study, ed estensione del formalismo a $K = 5$ componenti integrando il supporto socio-emotivo (M_4) e l'ancoraggio alla Zona di Sviluppo Prossimale (M_5);

2. **Dalla valutazione a valle all'allineamento della generazione:** impiego del PES come funzione di ricompensa differenziabile per l'ottimizzazione tramite *Reinforcement Learning from AI Feedback* (RLAIF) con algoritmi PPO e GRPO, allineamento diretto tramite *Direct Preference Optimization* (DPO) e ambienti di simulazione didattica multi-agente (*Self-Play*);

3. **Valutazione longitudinale e RAG agentico multi-step:** estensione dell'unità di analisi dal singolo turno atomico alla traiettoria di intera sessione (quantificando *Persistence* e *Hints Fading*), supporto nativo al RAG multimodale (immagini, grafici e formule LaTeX) e misura del *Tool-Use* didattico;

4. **Ottimizzazione software e telemetria industriale:** distillazione di conoscenza da modelli di frontiera verso *Small Language Models* compatti da 1–3B parametri operanti su dispositivi edge scolastici con latenze inferiori a 25 ms, standardizzazione della libreria come pacchetto distribuibile via PyPI e implementazione di sonde per l'allerta automatica di *pedagogical drift*;

5. **Sperimentazione clinica e misura del Learning Gain:** conduzione di studi randomizzati controllati (RCT) a doppio cieco su studenti autentici con valutazione dell'impatto sulla ritenzione mnemonica a lungo termine e sul trasferimento di competenze, affiancati da campagne di validazione con docenti di ruolo ed estensione multilingue (adattamento all'Indice Gulpease per la lingua italiana).

7.6 Riflessione finale ed etica: l'IA come levatrice del pensiero

Il contributo principale di questa tesi magistrale consiste nell'aver dimostrato che **la qualità pedagogica di un sistema di intelligenza artificiale può essere formalizzata, misurata e ottimizzata quantitativamente**.

Nel dibattito attuale sull'adozione dell'Intelligenza Artificiale Generativa nella didattica si confrontano spesso due posizioni opposte: da un lato, l'idea di considerare i modelli linguistici come strumenti capaci di risolvere automaticamente qualunque compito cognitivo; dall'altro, la preoccupazione che il loro utilizzo favorisca il plagio e riduca l'impegno critico degli studenti, inducendo a bandirli dalle aule.

Questo lavoro dimostra che esiste una strada diversa e più costruttiva: **allineare la progettazione tecnologica dei sistemi di intelligenza artificiale ai principi dell'apprendimento**.

Un assistente didattico che si limiti a fornire subito risposte complete finisce per incentivare la passività dello studente, offrendo un'illusione di comprensione a fronte di un mancato sforzo di ragionamento. Al contrario, un tutor guidato da metriche come il PES assume un ruolo maieutico: stimola la riflessione attraverso buone domande, valorizza l'errore come opportunità di chiarimento concettuale e supporta lo sviluppo delle capacità di pensiero autonomo del discente.

Garantire che i sistemi educativi basati su intelligenza artificiale promuovano il pensiero critico e l'autonomia dello studente, anziché la sola fruizione passiva di contenuti, rappresenta una priorità essenziale per lo sviluppo futuro delle tecnologie didattiche. Il framework `rag_eval` e il Pedagogical Evaluation Score intendono offrire un contributo concreto, formale e riproducibile in questa direzione.

Bibliografia

[1] Patrick Lewis et al. «Retrieval-augmented generation for knowledge-intensive nlp tasks». In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 9459–9474 (cit. alle pp. 1, 10).

[2] Joshua Maynez, Shashi Narayan, Bernd Bohnet e Ryan McDonald. «On Faithfulness and Factuality in Abstractive Summarization». In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 1906–1919 (cit. alle pp. 1, 10, 19, 86).

[3] Ziwei Ji et al. «Survey of hallucination in natural language generation». In: *ACM Computing Surveys* 55.12 (2023), pp. 1–38 (cit. alle pp. 1, 10).

[4] Grand View Research. *Artificial Intelligence in Education Market Size, Share & Trends Analysis Report*. `https://www.grandviewresearch.com`. 2024 (cit. a p. 2).

[5] Kishore Papineni, Salim Roukos, Todd Ward e Wei-Jing Zhu. «Bleu: a method for automatic evaluation of machine translation». In: *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 2002, pp. 311–318 (cit. alle pp. 2, 15).

[6] Chin-Yew Lin. «Rouge: A package for automatic evaluation of summaries». In: *Text summarization branches out*. 2004, pp. 74–81 (cit. alle pp. 2, 16).

[7] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger e Yoav Artzi. «BERTScore: Evaluating Text Generation with BERT». In: *International Conference on Learning Representations (ICLR)* (2020) (cit. alle pp. 2, 18).

[8] Lianmin Zheng et al. «Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena». In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 46595–46623 (cit. alle pp. 2, 19, 20, 88).

[9] Shahul Es, Jithin James, Luis Espinosa-Anke e Steven Schockaert. «Ragas: Automated evaluation of retrieval augmented generation». In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics* (2024), pp. 150–158 (cit. alle pp. 2, 21, 47).

[10] Jon Saad-Falcon, Omar Khattab, Keshav Santhanam, Taylor Berg-Kirkpatrick, Christopher Potts e Matei Zaharia. «ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems». In: *North American Chapter of the Association for Computational Linguistics (NAACL)* (2024) (cit. alle pp. 2, 21, 47).

[11] Lev S Vygotsky. *Mind in society: The development of higher psychological processes*. Harvard University Press, 1978 (cit. alle pp. 3, 22, 48, 60, 85, 95, 102).

[12] David Wood, Jerome S Bruner e Gail Ross. «The role of tutoring in problem solving». In: *Journal of child psychology and psychiatry* 17.2 (1976), pp. 89–100 (cit. alle pp. 3, 22, 24, 48, 60, 85, 98).

[13] Manu Kapur. «Productive failure». In: *Cognition and instruction* 26.3 (2008), pp. 379–424 (cit. alle pp. 3, 22, 24, 48, 85, 102).

[14] Dorottya Demszky, Jing Liu, Zid Mancenido, Julie Cohen, Heather Hill, Dan Jurafsky e Tatsunori Hashimoto. «Measuring Conversational Uptake: A Case Study on Student-Teacher Interactions». In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*. 2021, pp. 1638–1653 (cit. alle pp. 3, 23, 48, 57).

[15] Kaushal Kumar Maurya, K V Aditya Srivatsa, Kseniia Petukhova e Ekaterina Kochmar. «Unifying AI Tutor Evaluation: An Evaluation Taxonomy for Pedagogical Ability Assessment of LLM-Powered AI Tutors». In: *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics, 2025, pp. 1084–1108 (cit. alle pp. 6, 25, 27, 43, 67).

[16] Jakub Macina, Nico Daheim, Sankalan Pal Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych e Mrinmaya Sachan. «MathDial: A Dialogue Tutoring Dataset with Rich Pedagogical Properties Grounded in Math Reasoning Problems». In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. 2023, pp. 5602–5621 (cit. alle pp. 6, 14, 25, 43, 72, 93).

[17] Colin Camerer, George Loewenstein e Martin Weber. «The curse of knowledge in economic settings: An experimental analysis». In: *Journal of political Economy* 97.5 (1989), pp. 1232–1254 (cit. alle pp. 7, 53, 86, 96, 106).

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser e Illia Polosukhin. «Attention is all you need». In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017 (cit. alle pp. 10, 17).

[19] Tom Brown et al. «Language Models are Few-Shot Learners». In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 1877–1901 (cit. a p. 10).

[20] Stephen Robertson e Hugo Zaragoza. «The probabilistic relevance framework: BM25 and beyond». In: *Foundations and Trends in Information Retrieval* 3.4 (2009), pp. 333–389 (cit. a p. 11).

[21] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen e Wen-tau Yih. «Dense Passage Retrieval for Open-Domain Question Answering». In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. 2020, pp. 6769–6781 (cit. a p. 11).

[22] Yunfan Gao et al. «Retrieval-augmented generation for large language models: A survey». In: *arXiv preprint arXiv:2312.10997* (2023) (cit. alle pp. 12, 13).

[23] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni e Percy Liang. «Lost in the middle: How language models use long contexts». In: *Transactions of the Association for Computational Linguistics* 12 (2024), pp. 157–173 (cit. a p. 12).

[24] Luyu Gao, Xueguang Ma, Jimmy Lin e Jamie Callan. «Precise Zero-Shot Dense Retrieval without Relevance Labels». In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*. 2023, pp. 1762–1777 (cit. a p. 12).

[25] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil e Hannaneh Hajishirzi. «Self-rag: Learning to retrieve, generate, and critique through self-reflection». In: *International Conference on Learning Representations (ICLR)* (2024) (cit. a p. 12).

[26] Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan e Graham Neubig. «Active Retrieval Augmented Generation». In: *Empirical Methods in Natural Language Processing (EMNLP)* (2023) (cit. a p. 13).

[27] Omar Khattab et al. «DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines». In: *International Conference on Learning Representations (ICLR)* (2024) (cit. a p. 13).

[28] Benjamin S Bloom. «The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring». In: *Educational Researcher* 13.6 (1984), pp. 4–16 (cit. a p. 14).

[29] Arthur C Graesser, Patrick Chipman, Brandon C Haynes e Andrew Olney. «AutoTutor: An intelligent tutoring system with mixed-initiative dialogue». In: *IEEE Transactions on Education* 48.4 (2005), pp. 612–618 (cit. alle pp. 14, 86).

[30] John R Anderson, Albert T Corbett, Kenneth R Koedinger e Ray Pelletier. «Cognitive tutors: Lessons learned». In: *The Journal of the Learning Sciences* 4.2 (1995), pp. 167–207 (cit. a p. 14).

[31] Kurt VanLehn. «The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems». In: *Educational Psychologist* 46.4 (2011), pp. 197–221 (cit. alle pp. 14, 86).

[32] Gregory Kestin, Kelly Miller, Anna Klales, Trisha Milbourne e Gerhard Ponti. «AI tutoring outperforms in-class active learning: An RCT introducing a novel research-based design in an authentic educational setting». In: *Scientific Reports* 15.1 (2025), p. 17458 (cit. a p. 14).

[33] Ananya Gupta, Jessica Reddig, Tommaso Calò, Daniel Weitekamp e Christopher J MacLellan. «Beyond Final Answers: Evaluating Large Language Models for Math Tutoring». In: *International Conference on Artificial Intelligence in Education*. Springer. 2025, pp. 323–337 (cit. a p. 14).

[34] Verónica Parra, Ana Corica e Daniela Godoy. «Insights on the Pedagogical Abilities of AI-Powered Tutors in Math Dialogues». In: *Information* 17.1 (2026), p. 51 (cit. a p. 14).

[35] Anaïs Tack e Chris Piech. «The AI Teacher Test: Measuring the Pedagogical Ability of Blender and GPT-3 in Educational Dialogues». In: *Proceedings of the 15th International Conference on Educational Data Mining*. 2022, pp. 122–133 (cit. alle pp. 14, 61).

[36] Stanley F Chen e Joshua Goodman. «An empirical study of smoothing techniques for language modeling». In: *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*. 1996, pp. 310–318 (cit. alle pp. 16, 38).

[37] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev e Percy Liang. «SQuAD: 100,000+ Questions for Machine Comprehension of Text». In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2016, pp. 2383–2392 (cit. a p. 16).

[38] Chia-Wei Liu, Ryan Lowe, Iulian Dey, Michael Noseworthy e Joelle Pineau. «How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation». In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2016, pp. 2122–2132 (cit. a p. 17).

[39] Chris Callison-Burch, Miles Osborne e Philipp Koehn. «Re-evaluating the Role of BLEU in Machine Translation Evaluation». In: *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*. 2006, pp. 249–256 (cit. a p. 17).

[40] Jacob Devlin, Ming-Wei Chang, Kenton Lee e Kristina Toutanova. «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding». In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2019, pp. 4171–4186 (cit. a p. 17).

[41] Pengcheng He, Xiaodong Liu, Jianfeng Gao e Weizhu Chen. «DeBERTa: Decoding-enhanced BERT with Disentangled Attention». In: *International Conference on Learning Representations (ICLR)*. 2021 (cit. alle pp. 17, 28, 57).

[42] Nils Reimers e Iryna Gurevych. «Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks». In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. 2019, pp. 3982–3992 (cit. alle pp. 18, 28, 35).

[43] Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer e Steffen Eger. «MoverScore: Text Generation Evaluating with Contextualized Embeddings and Earth Mover's Distance». In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019, pp. 563–578 (cit. a p. 18).

[44] Thibault Sellam, Dipanjan Das e Ankur P Parikh. «BLEURT: Learning Robust Metrics for Text Generation». In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 7881–7892 (cit. a p. 18).

[45] Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu e Zhifang Sui. «Large language models are not fair evaluators». In: *arXiv preprint arXiv:2305.17926* (2023) (cit. a p. 20).

[46] Alison King. «Guiding knowledge construction in the classroom: Effects of teaching children how to question and how to explain». In: *American Educational Research Journal* 31.2 (1994), pp. 338–368 (cit. a p. 22).

[47] Michelene TH Chi. «Active-constructive-interactive: A conceptual framework for differentiating learning activities». In: *Topics in Cognitive Science* 1.1 (2009), pp. 73–105 (cit. alle pp. 23, 86).

[48] Rudolph Flesch. «A new readability yardstick». In: *Journal of Applied Psychology* 32.3 (1948), pp. 221–233 (cit. alle pp. 23, 59).

[49] Carol S Dweck. *Mindset: The new psychology of success*. Random House, 2006 (cit. alle pp. 24, 48, 58, 94).

[50] John Sweller. «Cognitive load during problem solving: Effects on learning». In: *Cognitive Science* 12.2 (1988), pp. 257–285 (cit. alle pp. 24, 48, 95).

[51] Jakub Macina, Nico Daheim, Ido Hakimi, Manu Kapur, Iryna Gurevych e Mrinmaya Sachan. «MathTutorBench: A Benchmark for Measuring Open-ended Pedagogical Capabilities of LLM Tutors». In: *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 2025, pp. 204–221 (cit. a p. 26).

[52] Rakshith Sharma Srinivasa et al. «TutorBench: A Benchmark To Assess Tutoring Capabilities Of Large Language Models». In: *arXiv preprint arXiv:2601.14488* (2026) (cit. a p. 26).

[53] Weikang Shi et al. «From Solver to Tutor: Evaluating the Pedagogical Intelligence of LLMs with KMP-Bench». In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 40. 26. 2026, pp. 32965–32973 (cit. a p. 26).

[54] Gemini Team, Google. «Gemini: A Family of Highly Capable Multimodal Models». In: *arXiv preprint arXiv:2312.11805* (2024) (cit. a p. 27).

[55] DeepSeek-AI et al. «DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning». In: *arXiv preprint arXiv:2501.12948* (2025) (cit. a p. 27).

[56] Abhimanyu Dubey et al. «The Llama 3 Herd of Models». In: *arXiv preprint arXiv:2407.21783* (2024) (cit. alle pp. 27, 84).

[57] Josh Achiam et al. «GPT-4 Technical Report». In: *arXiv preprint arXiv:2303.08774* (2023) (cit. a p. 27).

[58] Albert Q Jiang et al. «Mistral 7B». In: *arXiv preprint arXiv:2310.06825* (2023) (cit. a p. 27).

[59] Marah Abdin et al. «Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone». In: *arXiv preprint arXiv:2404.14219* (2024) (cit. a p. 27).

[60] Niklas Muennighoff, Nouamane Tazi, Loïc Magne e Nils Reimers. «MTEB: Massive Text Embedding Benchmark». In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. 2023, pp. 2014–2037 (cit. a p. 27).

[61] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian e Zheng Liu. «BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation». In: *arXiv preprint arXiv:2402.03216* (2024) (cit. a p. 27).

[62] Liang Wang, Nan Yang, Xiaodi Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder e Furu Wei. «Text Embeddings by Weakly-Supervised Contrastive Pre-training». In: *arXiv preprint arXiv:2212.03533* (2022) (cit. a p. 27).

[63] Harrison Chase. *LangChain*. `https://github.com/langchain-ai/langchain`. 2022 (cit. a p. 28).

[64] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E Gonzalez, Hao Zhang e Ion Stoica. «Efficient Memory Management for Large Language Model Serving with PagedAttention». In: *Proceedings of the 29th ACM Symposium on Operating Systems Principles (SOSP)*. 2023, pp. 611–626 (cit. alle pp. 29, 84).

[65] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov e Christopher D Manning. «HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering». In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 2018, pp. 2369–2380 (cit. a p. 41).

[66] Pranav Rajpurkar, Robin Jia e Percy Liang. «Know What You Don't Know: Unanswerable Questions for SQuAD». In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 2018, pp. 784–789 (cit. a p. 42).

[67] Tom Kwiatkowski et al. «Natural Questions: A Benchmark for Question Answering Research». In: *Transactions of the Association for Computational Linguistics* 7 (2019), pp. 453–466 (cit. a p. 42).

[68] Mandar Joshi, Eunsol Choi, Daniel S Weld e Luke Zettlemoyer. «TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension». In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. 2017, pp. 1601–1611 (cit. a p. 42).

[69] Jiawei Chen, Hongyu Lin, Xianpei Han e Le Sun. «Benchmarking Large Language Models in Retrieval-Augmented Generation». In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. 16. 2024, pp. 17754–17762 (cit. a p. 42).

[70] Tingyu Liu, Xiaoming Zhang, Yuxuan Xiao, Ming Wang, Chen Wang e Wayne Xin Zhao. «RECALL: A Benchmark for LLMs Robustness against External Counterfactual Knowledge». In: *arXiv preprint arXiv:2311.08147* (2023) (cit. a p. 42).

[71] Martin Nystrand, Adam Gamoran, Robert Kachur e Catherine Prendergast. *Opening Dialogue: Understanding the Dynamics of Language and Learning in the English Classroom*. Teachers College Press, 1997 (cit. alle pp. 48, 57).

[72] Abhijit Suresh, Jennifer Jacobs, Vivian Lai, Chenhao Tan, Wayne Ward, James H Martin e Tamara Sumner. «Using Transformers to Provide Teachers with Personalized Feedback on their Classroom Discourse: The TalkMoves Application». In: *arXiv preprint arXiv:2105.07949* (2021) (cit. a p. 93).

[73] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford e Oleg Klimov. «Proximal policy optimization algorithms». In: *arXiv preprint arXiv:1707.06347* (2017) (cit. a p. 96).

[74] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, YK Li, Y Wu e Daya Guo. «DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models». In: *arXiv preprint arXiv:2402.03300* (2024) (cit. a p. 96).

[75] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning e Chelsea Finn. «Direct Preference Optimization: Your Language Model is Secretly a Reward Model». In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 53728–53741 (cit. a p. 97).

[76] Geoffrey Hinton, Oriol Vinyals e Jeff Dean. «Distilling the knowledge in a neural network». In: *arXiv preprint arXiv:1503.02531* (2015) (cit. a p. 99).