From Search as Learning to Converse for Learning

SAL

AIED (pre LLM)

Transformer Shift

CHATBOTS and LLM

Converse
- Prompt Engineering
- RAG
- Specialized Conversational Agents
- Socratic posture

SAPIENZA
UNIVERSITÀ DI ROMA

Marco Temperini
Dept. Computer, Control, and Management Engineering

The complex balance

TEL is not about *adding* technology to education; it is about designing learning environments with technology, guided by learning theory, pedagogical design, and evidence about how people learn.

SUPPORT

Supported learning =
cut wasteful (*extraneous*) load
in the student's ever-expanding knowledge environment
(web, repositories, GenAI preTraining ...).

EFFORT

Keep the effort that builds learning — critical thinking, cognitive capability, robust competence.

- Then, sometimes, the balance is lost:
  "Yay — let's everybody use ChatGPT to study!"
  An obvious step — too enthusiastic, reckless. Then we started to see ... and it ***was*** stopped ...

... was?

Search as Learning

Step 1 - example: from retrieval to constructive cognition

Definition

Search as Learning is the perspective that conceptualizes, designs and evaluates information searching as a *learning process in its own right* — an iterative, reflective and integrative activity in which the searcher's **knowledge structures are progressively modified** through querying, evaluating, comparing, and synthesizing sources.

Search systems are therefore reframed: not retrieval tools for relevance and look-up, but environments that sustain understanding, analysis, evaluation and creation — the higher levels of Bloom's taxonomy.

Vakkari (2016) • Rieh, Collins-Thompson, Hansen & Lee (2016) • von Hoyer et al. (2022) • Urgo & Arguello (2025)

Cognitive

Learning = change in knowledge structures (assimilation, tuning, restructuring)

Iterative

Comprehensive search: multiple sessions of analysis, synthesis, evaluation

Pedagogical

Constructivism, Vygotsky's ZPD, scaffolding — borrowed from learning sciences

SAL in action — Marta and the 2008 crisis

Search engine only. No conversational agent. Learning happens through querying, reading, and reformulating.

Four pedagogical features

1 Knowledge structures change

Queries restructure, tune, and populate concept networks — not just retrieve facts.

2 Self-regulated learning

Preparation → performance → appraisal. The searcher monitors her own progress.

3 Scaffolding within the ZPD

Search interface, snippets, and source comparison are the scaffolds available here.

4 Retention and transfer

Knowledge that persists and can be applied to new problems — beyond the immediate answer.

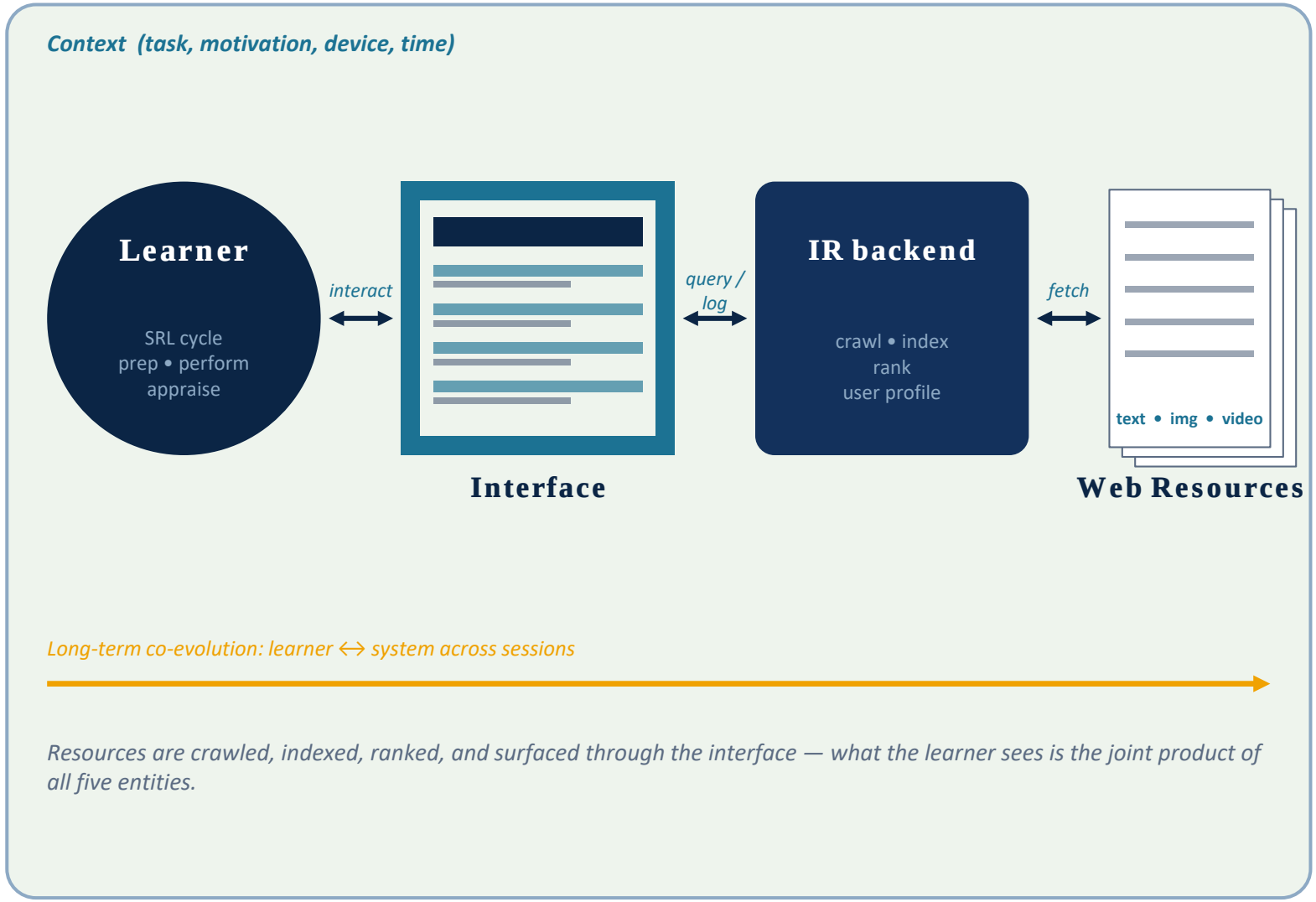
Marta — pure search workflow

Prepare	Activates prior knowledge (“subprime”), sets a goal: explain the causal chain in 4 paragraphs.
Query 1	Broad: “2008 financial crisis causes”. Vague relevance criteria — many sources of possible value. Restructuring phase.
Compare	Reads Wikipedia and Investopedia. Encounters “securitization” — a new concept she does not yet understand.
Reformulate	New query: “securitization mortgage-backed securities explained”. More specific terms; tuning phase begins.
Appraise	Drafts the essay. Identifies a remaining gap on systemic risk and runs one final, focused query.

Learning is in the concept network Marta restructured by reading and comparing sources.

The Spaceship Model of SAL

von Hoyer et al. (2022) — an interdisciplinary view of learner, interface, and IR backend



The five entities

A — Context
Task complexity, motivation, time, device, setting

B — Learner
Cognitive abilities, prior knowledge, SRL cycle

C — Interface
Query box, SERP, snippets — shapes what is seen

D — IR Backend
Crawler, index, ranking, user/learner profile

E — Resources
Web content of varied modality and quality

Psychology and IR are not adjacent: they co-design what learning can become.

Artificial Intelligence in Education 🥰

Step 2 (pre-LLM): personalization and adaptation, before the ... transformer shift

AI techniques to pursue the benefit Bloom attributed to one-to-one tutoring.

Tailoring content, pace, and feedback to each learner through **personalization** (fitting stable learner traits) and **adaptation** (reacting to the learner's evolving state during the interaction).

Intelligent Tutoring Systems

Guide a learner step-by-step through a structured domain, with diagnosis and feedback

Educational Recommender Systems

Select learning resources or activities from a larger pool to suit the learner

Common engine: the Student Model

Both families infer the learner's state and act on it — the shared core of AIED

At the centre of every AIED system sits a **Student Model** *— an inference of what the learner knows, can do, and misunderstands. Caveat: adaptation is not automatically helpful (expertise reversal effect -* Kalyuga et al. 2003*).*

Blloom (1984) • Anderson et al. (1995) • Brusilovsky (1996, 2001) • VanLehn (2011) • Manouselis et al. (2011)

How pre-LLM AIED systems work

Two questions every system must answer: what does the learner know? — and what should come next?

Modelling the learner

1. **Overlay & model tracing (rule-based)**
   Learner knowledge as a subset of expert rules; follow each solution step (Cognitive Tutors / ACT-R). (Anderson et al. 1995)

2. **Constraint-based modelling**
   Encode constraints any correct answer must satisfy; flag violations — lighter to author. (Ohlsson, 1994; Mitrovic et al., 2001)

3. **Bayesian Knowledge Tracing**
   Probabilistic, data-driven estimate of skill mastery as the learner answers over time. (Corbett & Anderson, 1995)

4. **Deep Knowledge Tracing**
   A neural network learns the skill model from large interaction logs — no hand-built skills. (Piech et al., 2015):

Deciding what comes next

Rules Vs. Data: Hand-authored expert rules, OR models learned from interaction data (machine learning).

Sequencing — Prerequisite-aware progression — knowledge spaces, adaptive hypermedia, mastery thresholds.

Coll. filtering — "Learners like you also studied X" — patterns across many learners, no content analysis.

Content-based — Match resource features to the learner profile.

Reinforcement L. — learn a teaching policy from interaction data. Lighter rule authoring, heavi in model design, reward specification, ... explainability.

What works — and at what cost

The recurring trade-off: interpretable and expensive (rule based), maybe brittle, vs scalable and opaque

Strengths reported in the literature

ITS	Learning gains close to human tutoring, well above conventional instruction (VanLehn, 2011).
Recommenders	Scale to large/open resource pools; support informal and lifelong learning.
BKT	Simple, transparent, drives mastery-based progression.
DKT	Higher predictive accuracy; no hand-engineered skill model needed.

Limitations & caveats

ITS	Authoring bottleneck — hundreds of expert hours per hour of teaching; domain-specific, brittle (Murray, 1999).
CF / content	Cold-start & sparsity (worse in small cohorts); content-based over-specialises and narrows exposure.
Reinf. Learning	Exploring on real learners raises ethic observations; what is reward for 'learning' is hard to specify; RL evidence is mixed (Doroudi et al., 2019).
BKT / DKT	BKT: no forgetting, identifiability problems (Beck & Chang, 2007). DKT: opaque, data-hungry. Some controversies, matched by tuned BKT (Khajah et al., 2016).

incoming transformer shift will reopen the discussion.

References — Pre-LLM AIED

Bloom, B.S. (1984). The 2 sigma problem. Educational Researcher, 13(6), 4–16.

Anderson, J.R., Corbett, A.T., Koedinger, K.R. & Pelletier, R. (1995). Cognitive tutors: Lessons learned. J. of the Learning Sciences, 4(2), 167–207.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*(4), 1036–1060.

Corbett, A.T. and Anderson, J.R. (1995) Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. User Modelling and User-Adapted Interaction, 4, 253-278. https://doi.org/10.1007/bf01099821

Mitrovic, Antonija. (2012). Fifteen years of constraint-based tutors: What we have achieved and where we are going. User Modeling and User-Adapted Interaction. 22. 39-72. 10.1007/s11257-011-9105-9..

Piech, C. et al. (2015). Deep knowledge tracing. NeurIPS 28, 505–513.

Khajah, M., Lindsey, R.V. & Mozer, M.C. (2016). How deep is knowledge tracing? EDM 2016.

Beck, J.E. & Chang, K.-m. (2007). Identifiability: A fundamental problem of Student Modeling. UM 2007, LNCS 4511, pp. 137–146.

VanLehn, K. (2011). The relative effectiveness of human tutoring, Intelligent Tutoring Systems, and other tutoring systems. Educational Psychologist, 46(4), 197–221.

Murray, T. (1999). Authoring Intelligent Tutoring Systems: An Analysis of the State of the Art. *International Journal of Artificial Intelligence in Education*, 10, 98–129.

Kalyuga, S., Ayres, P., Chandler, P. & Sweller, J. (2003). The expertise reversal effect. Educational Psychologist, 38(1), 23–31.

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. UMUAI, 6(2–3), 87–129.

Brusilovsky, P. (2001). Adaptive hypermedia. UMUAI, 11(1–2), 87–110.

Manouselis, N. et al. (2011). Recommender systems in Technology Enhanced Learning. In Recommender Systems Handbook (1st ed.), pp. 387–415. Springer.

Drachsler, H., Verbert, K., Santos, O. C., & Manouselis, N. (2015). Panorama of Recommender Systems to Support Learning. In F. Ricci, L. Rokach, & B. Shapira (Eds.), Recommender Systems Handbook (2 ed., pp. 421-451). Springer. https://doi.org/10.1007/978-1-4899-7637-6_12.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. Proc. ACM conference on Computer supported cooperative work, Pages 175 - 186, 1994

Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In Proc.19th int. conference on World wide web. ACM, pp. 661–670.

Doroudi, S., Aleven, V. & Brunskill, E. Where's the Reward?. Int J Artif Intell Educ 29, 568–620 (2019).

Generative (Search &) Explanation

NLP models helping to produce (generate) synthesized explanations

What changed: **self-attention** helps representing words (well ... tokens) by vectors weighting them "in parallel" – they get "context" in this way

BERT

encoder · understands

Reads a passage both ways at once; trained by predicting masked words. Brilliant at *comprehending* — classifying, scoring, matching. It judges meaning; it does not compose new text.

GPT (early)

decoder · generates

Predicts the next word, again and again, so it can *produce* new fluent text, compelling, convincing — a question, an explanation. GPT-1/2 proved this worked, but needed fine-tuning and human-feedback-based alignment.

The means were already here in 2018–19 — *available at the cost of more work, and less powerful than what would follow.*

Vaswani et al. (2017) Attention Is All You Need; Devlin et al. (2018) BERT; Radford et al. (2018, 2019) GPT-1 / GPT-2.

What a transformer is

Self-attention (Vaswani et al., 2017): instead of reading word-by-word, every token weighs every other token in the passage, in parallel.

Context

A word's meaning is computed from the whole sentence: *"bank"* in *river bank* vs *central bank* — exactly the disambiguation good feedback needs.

Transfer

Pre-train once on huge raw text, then *fine-tune* on a small task set. No more hand-engineered linguistic features — a small course dataset becomes workable.

The cost

But fine-tuning still demanded labelled data, GPU time, engineering. Powerful — yet narrower and more fragile than what came next.

Vaswani et al. (2017) Attention Is All You Need;

Already at work — the 2018–19 means

Understanding

BERT grades free-text answers

A BERT-based framework for *automatic short-answer grading* inside an ITS. Fine-tuned BERT encodes the student's answer and the systen judges by meaning, not keyword overlap. The model outperformed most state-of-the-art systems, achieving the best published (44% accuracy) result.
The machine doesn't just detect keywords — it grasps what the student meant, and classifies consequently.

Zhu, Wu & Zhang (2022) · Automatic Short-Answer Grading via BERT-Based Deep Neural Networks. 364-375IEEE Trans. Learning Technologies 15(3)

Generation

~~GPT-2~~ T5 asks questions

Another Transformer based model, an encoder-decoder actually, but used for generation only. It wrote short-answer questions on reading passages. In a controlled study (48 students), comprehension measurably improved — even when learners knew the questions were machine-made. The gain came from *provoking retrieval & self-testing*, not handing over answers.

Steuer, Filighera, Tregel & Miede (2022) . Educational Automatic Question Generation Improves Reading Comprehension in Non-native Speakers: A Learner-Centric Case Study· Frontiers in Artificial Intelligence 5

The takeaway: the means were here — and value came from generation that *provoked effort, not replaced it.*

References — Transformer shift

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems 30 (Nips 2017), 5998–6008. arXiv:1706.03762

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018/2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805 (2018); Proceedings of NAACL-HLT 2019, 4171–4186

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training. OpenAI technical report. — GPT-1: the decoder branch, generative pre-training.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI technical report. — GPT-2: scale and zero-/few-shot multitasking, the run-up to the LLM era.

Chatbots, before the generative turn

1966

ELIZA

Rogerian therapist

Pattern matching + keyword substitution. No memory, no understanding — yet users confided in it. Weizenbaum was alarmed: the *ELIZA effect* — people attribute intelligence to responsive machines.

1972

PARRY

paranoid patient

~500 heuristic rules + internal affect state (anger, fear, mistrust). In a Turing-like test, psychiatrists identified the computer **only 48% of the time** — indistinguishable from chance.

1995–

ALICE

open-domain chat

AIML: an XML pattern-matching language, community-maintained. Tens of thousands of patterns — 3× Loebner Prize winner. Scale, yet still *no understanding*: a knowledge base of templates.

Predictable & explainable by definition — every response traces to its rule.

Sensational at drawing people in — but brittle beyond their scripts.

Weizenbaum (1966); Colby et al. (1972); Heiser et al. (1980); Wallace (2009)

How they worked — and the trade-off

Finite-state automata	Pattern matching / rules	Internal-state models	Conversation schemata	FAQ retrieval
Fixed dialogue paths	*Input → template → response*	*Affect variables shape replies*	*Overall trajectory / topic steering*	*Answers from prepared documents*

Predictable & explainable

Every response traces to the rule that fired. No hidden representations, no hallucinations. Full control over the dialogue space — valuable in high-stakes contexts.

Brittle & coverage-limited

Every new topic = hand-authored rules. Scaling is linear in human effort. 500 rules gave PARRY a persona, not a mind. 40K patterns gave ALICE breadth, not understanding.

The ELIZA effect: users attribute understanding to responsive systems. A pedagogical hazard as much as a curiosity — *it will become far more urgent when the system can actually generate fluent, confident, wrong text.*

Abu Shawar & Atwell (2007) for the explainability argument; Jurafsky & Martin (2023) Speech and Language Processingch.15 for a technique taxonomy.

References — pit stop

Weizenbaum, J. (1966). ELIZA — A computer program for the study of natural language communication between man and machine. Communications of the ACM, 9(1), 36–45.

Colby, K. M., Hilf, F. D., Weber, S., & Kraemer, H. C. (1972). Turing-like indistinguishability tests for the validation of a computer simulation of paranoid processes. Artificial Intelligence, 3, 199–221.

Heiser, J. F., Colby, K. M., Faught, W. S., & Parkison, R. C. (1980). Can psychiatrists distinguish a computer simulation of paranoia from the real thing? The limitations of Turing-like tests as measures of the adequacy of simulation. Journal of Psychiatric Research, 15(3), 149–162. Cambridge Core

Wallace, R. S. (2009). The anatomy of A.L.I.C.E. In R. Epstein, G. Roberts, & G. Beber (Eds.), Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer (pp. 181–210). Springer.

The ChatGPT moment

November 2022

OpenAI wraps GPT-3.5 in a free, browser-based chat box. No API, no engineering skill, no download. Just type and get a reply.

1M
users in 5 days

100M
users in 2 months

The TEL challenge

Does it actually help students learn? Can it support and augment learning — or does it replace the effort that makes learning stick? Are the advantages clear or controversial? And if students adopt it faster than we can study it… who controls the balance?

A period of enthusiastic adoption followed. *Then the drawbacks appeared.*

The enthusiastic adoption

But: ChatGPT **reduces mental effort**. Among the studies measuring mental effort, only one explicitly banned ChatGPT from the post-test: **students reasoned and argued more weakly without the tool** (Stadler et al., 2024). Less mental effort – less reasoning

But: Shorter interventions show larger effects — The benefit fades (or students learn to stop overestimating it?). Long-term studies barely exist (well, the technology is too new in 2024).

But: Higher-order thinking improvements measured by self-report (students say they think more critically). No study in the review uses objective measures — do students actually reason better, or just feel they do?

Deng et al. (2025) · 69 experimental studies, Computers & Education

The reckoning

Cognitive ease at a cost

91 students: ChatGPT vs Google. LLM users had lower cognitive load across all three facets (extraneous, intrinsic, germane) — but produced **lower-quality reasoning**. ChatGPT use → less germane load → lower-quality reasoning.

ChatGPT didn't make students reason badly — it made reasoning feel unnecessary, and that's what made them reason badly.

Stadler, Bannert & Sailer (2024) Computers in Human Behavior 160

ChatGPT as a cognitive crutch

n=120. Surprise test after **45 days**: AI-assisted students scored 57.5% vs 68.5% traditional).

Steeper forgetting curve. Effect persisted across all topics and prior AI experience.

“a generation of students who remember where to find information rather than the information itself.”

Barcaui (2025) Social Sciences & Humanities Open 12

Generative AI can harm learning

~1000 high-school students. GPT Base: **+48% on practice, −17% on exam**.

GPT Tutor (with safeguards): +127% on practice, no harm on exam.

Safeguards (teacher-designed hints) mitigated the damage.

Bastani et al. (2025) PNAS

The pattern: LLMs ease the task but erode the learning. *Unrestricted access offloads the cognitive effort that makes knowledge stick.*

The balance damaged

Adoption outpaces understanding

Students adopt LLMs **independently and massively**. The research community's understanding of the problems, and the design of solutions, moves slower. The result: millions of students using general-purpose LLMs without pedagogical control.

What's broken

- **Germane load reduced** — not just extraneous (Stadler)
- **Long-term retention harmed** — at 45 days (Barcaui)
- **Metacognitive blind spot** — fluency ≠ understanding
- **Without safeguards, −17%** — on exams (Bastani)

The complex balance of TEL is damaged. The intro promised a path of equilibrium between reducing wasteful load and preserving productive effort. Unrestricted LLM access has tipped the scale: it cut on *all* the load — including the effort that builds learning.

But Bastani showed that safeguards work. *The question is not whether to use LLMs — it's how.*

STEP 6 ·

Restoring the balance

1

Prompt Engineering as student literacy

Equip the student to interact with any LLM effectively and critically.

2

Specialized agent grounded by RAG

The teacher’s two fingerprints: a behavioural prompt + a curated knowledge base.

3

The Socratic posture

Guide through questions, not answers. Protect the effort that builds learning.

The answer to Step 5’s damaged balance: not a ban on LLMs, but a **framework.**

The student's role

Even with a well-designed agent, the quality of the interaction depends on the quality of the student's input.

What is Prompt Engineering?

The practice of crafting inputs to LLMs that guide the model toward more accurate, relevant, and useful outputs.

Better AI outputs
More effective use of any LLM, specialized or general-purpose.

Better thinking
Formulating a good prompt requires planning, self-monitoring, and evaluation — metacognitive skills that transfer beyond AI.

Qian (2025) · J. Educational Computing Research | Lee & Palmer (2025) · Int. J. Ed. Tech. in Higher Ed.

Two students, two literacies

without and with the use of Prompt Engineering principles and patterns.

The professor wants me to do a homework.
Ah, it's about a C program that should satisfy the following description: <text of the homework>.
Do the program.

I'm working on a C programming assignment for my first-year Programming Techniques course. We've covered arrays, pointers, and file I/O but haven't used dynamic memory allocation yet.
Here's the task: <text of the homework>.
Draw the solving algorithm and point out its parts step by step and explain why that approach fits this problem.
Write the solution as modular code: one function per logical task. Use comments in each function, with an explanation on top of it. Use snake_case for function names.
... sample: if the input is [example input], the output should be [expected output].
Are there cases the program doesn't handle?
If you find problems, fix them and tell me what you changed.AI.

Qian (2025) · J. Educational Computing Research | Lee & Palmer (2025) · Int. J. Ed. Tech. in Higher Ed.

Five principles — the ingredients

Give Direction	Specify Format	Provide Examples	Evaluate Quality	Divide Labour
Role, style, audience	*Structure the output*	*One-shot / few-shot*	*Self-critique & verify*	*Chain sub-steps*

Example — Divide Labour:

"Analyse this short story in three steps. Step 1: Identify the narrator's point of view. Step 2: List the three most important symbols. Step 3: Based on steps 1–2, write a thesis statement."

Each principle works with any LLM — specialized or general-purpose.

Five patterns — the recipes

PT1
Semantic prompts
Context-rich natural-language prompting

PT2
Error identification
Submit your work; model critiques

PT3
Context control
Set explicit boundaries on scope

PT4
Chain of Thought
Show reasoning step by step

PT5
Template-based
Reusable prompt with placeholders

Exempio — Chain of Thought:

"A train travels Rome–Milan at 120 km/h and returns at 80 km/h. What is the average speed? Think step by step: calculate each leg's time, then total time, then average speed. Show every step."

Principles are the ingredients; patterns are the recipes *— reusable structures that shape the conversation.*

STEP 6 · PROMPT ENGINEERING AS LITERACY

"I'm working on a C programming assignment for my first-year Programming Techniques course. We've covered arrays, pointers, and file I/O but haven't used dynamic memory allocation yet."→ Context control (pattern): constrains the solution to the student's actual level — the LLM won't produce code with malloc/realloc the student couldn't defend.
Also give directions (principle): tells the LLM what's in bounds and what isn't.

"Draw the solving algorithm and point out its parts step by step and explain why that approach fits this problem."→ Chain of thought (pattern): "step by step" forces sequential reasoning before code.
Semantic prompt (pattern): "explain why that approach fits" asks for justification, not just output — the student will read a rationale, not just a solution.
Divide labour (principle): separates algorithm design from implementation — think first, code second.

"Write the solution as modular code: one function per logical task."→ Divide labour (principle)
Also give directions (principle): specific structural instruction.

"Use comments in each function, with an explanation on top of it. Use snake_case for function names."→ Specify format (principle): prescribes commenting style, naming convention, and code layout.

"... sample: if the input is [example input], the output should be [expected output]."→ Provide examples (principle) preventing misinterpretation.
Also template based (pattern):

"Are there cases the program doesn't handle?"→ Error identification (pattern): explicitly asks the LLM to find gaps and edge cases.
Evaluate quality (principle): prompts self-assessment before the student accepts the result.

"If you find problems, fix them and tell me what you changed."→ Evaluate quality (principle) ...gives the student a visible diff to learn from.
Error identification (pattern)

overall: template based — the prompt's own structure (context → algorithm → code → examples → review) implicitly templates the response order, so the LLM will reply in that sequence. The student has designed the shape of the answer before seeing it.

The teacher's two fingerprints

The behavioural prompt

Defines the agent's pedagogical posture, tone, workflow, and boundaries. Written in natural language — no code, no fine-tuning.

Part systemic (a base, instructional agent)
Part teacher-determined (course-specific instructions)

The knowledge base (RAG)

Course notes, textbooks, readings, transcripts. The teacher decides what goes in — and nothing else enters. Lecture notes, observations, horror stories

Grounds responses in verified content
Updated by uploading files — no retraining

A specialized agent ≠ a general chatbot. It knows what the teacher wants it to know, and behaves as the teacher instructs.

How RAG works

1 Index

Course materials are split into chunks and embedded into a vector database.

2 Retrieve

Student's question is matched against the database for the most relevant passages.

3 Generate

Retrieved passages + question are fed to the LLM, which generates a grounded response.

- **Reduced hallucination**
- **Teacher-controlled scope**
- **Verifiable sources**
- **No retraining needed**

Limits: residual hallucinations on topics with weak meterial; images; computational cost.

RAG in the classroom

Németh et al. (2025) · 4 courses, 2 universities, GPT-4o + RAG tutor

1.5%

of answers
factually incorrect

82%

correct AND grounded
in course material

16.5%

outside the provided
context

*"The AI tutor does not simply impart knowledge — it **mediates the relationship** between student, lecturer, and course material."*

Social Sciences & Humanities Open 12, 101751.

Guide through questions, not answers

The Socratic method: instead of providing a direct answer, pose a sequence of probing questions that lead the learner to discover the answer through their own reasoning.

Clarifying	*"What do you mean by…?"*
Probing assumptions	*"What are you taking for granted here?"*
Probing evidence	*"What evidence supports that?"*
Exploring implications	*"If that's true, what follows?"*
Questioning the question	*"Why is this question important?"*

Lineage: **AutoTutor** (Graesser, 1999–2016) showed Socratic-style ITS could match human tutors (for each topic lots of hand-authored dialogue).
Favero, Pérez-Ortiz, Käser, Oliver (2025). Enhancing Critical Thinking by Means of a Socratic Chatbot. Llama-2 (7B/13B) as a Socratic tutor

the evidence

S-ICA vs nS-ICA: same tasks, same interface, same time — the only difference was Socratic questioning vs direct answers, and the engine.

What improved

- Academic achievement (modest)
- Reflective thinking (strong)
- Quality of group research proposals
- Motivation, no

In their words

"The system prevented me from relying on ready-made answers."

Nearly every student in the S-ICA group made this observation — the exact opposite of the cognitive offloading from Step 5.

Xi, Zhang & Wang (2026) · 94 students · Computers & Education 241

The teacher's fingerprint

Systemic defaults (built into the platform)

- Never give direct answers
- Require justification
- Probing questions
- Connect to prior knowledge

Teacher-specific (only the teacher can provide)

Terminology enforcement

Use course vocabulary, not colloquial synonyms

Common misconceptions

Flag known confusions (also as RAG examples)

Assessment alignment

Match practice format to exam type

The prompt is part systemic, part teacher-determined. *The knowledge base is entirely the teacher's.*

STEP 6 · SOCRATIC AGENT IN ACTION

In practice: Marta's essay

Socratic posture · RAG-grounded in course readings

Marta I need to write about the causes of the 2008 financial crisis. Can you give me the main causes?

Agent Before I list anything — what do you already know about what triggered the crisis? Start with whatever comes to mind.

Marta Something about banks giving mortgages to people who couldn't pay them back?

Agent Good start. Those are called subprime mortgages. But why would a bank deliberately lend to someone likely to default? What incentive structure could explain that?

• • •

Marta Maybe they sold the risk to someone else? Like, repackaged the loans?

Agent Exactly — that's securitization. Now, your course reading (Ch. 7) describes a specific mechanism that made this seem safe. Can you find it?

No answers given as an endpoint. Each agent turn probes, connects to course material, and pushes Marta to reason.

STEP 6 ·

The three pillars — together

1

Prompt Engineering as student literacy

The student is equipped to interact with any LLM effectively and critically — a new form of literacy.

2

Specialized agent grounded by RAG

The teacher's two fingerprints: behaviour via prompt, content via knowledge base. 1.5% error rate.

3

The Socratic posture

Guide through questions. Protect the effort. "The system prevented me from relying on ready-made answers."

The question is not whether to use LLMs in education — it's ***how to use them without losing the learning.***

Does PE training make a difference?

Addiucci, Temperini, Carfora & Dezi (2025) · High school, Rome · Analytical Geometry

Class 3α n = 20

PE-trained from start • uses ChatGPT from Phase 1

Class 3β n = 17

Waitlist control • PE-trained only in Phase 2

Pre-test	PE training	Exercise 1	PE training	Exercise 2	Post-test
in person no LLM	3α only (2 hours)	3α + LLM 3β no LLM	3β now trained	both classes + LLM	in person no LLM

Waitlist design: all students eventually receive PE training — ethical requirement of the school programme.

The trajectories

3α

Pre-test	Ex 1	Ex 2	Post-test
6.28	7.63	9.30	8.90
no LLM	*SIGNIFICANT*	*SIGNIFICANT*	*NO LLM - NOT SIG*

Steady climb
and non significant drop at
EX2→Post

3β

Pre-test	Ex 1	Ex 2	Post-test
7.79	5.79	9.13	8.99
no LLM	*SIG ↓*	*SIG*	*NO LLM - NOT SIG*

Drop then surge
and non significant drop at
EX2→Post

Without PE: −2.00
After PE: +3.34

What PE training changed

Performance improved

Both classes improved significantly from pre- to post-test. 3β's drop→surge pattern isolates the PE training effect.

Interaction became strategic

Divide Labour most used principle; Context Control, Chain of Thought most used patterns. Follow-up prompts: 86% of students.

Mild offloading persists

Small, non-significant drop when LLM is removed. PE mitigates but does not eliminate cognitive delegation.

PE training works

Addiucci, Temperini, Carfora & Dezi (2025)

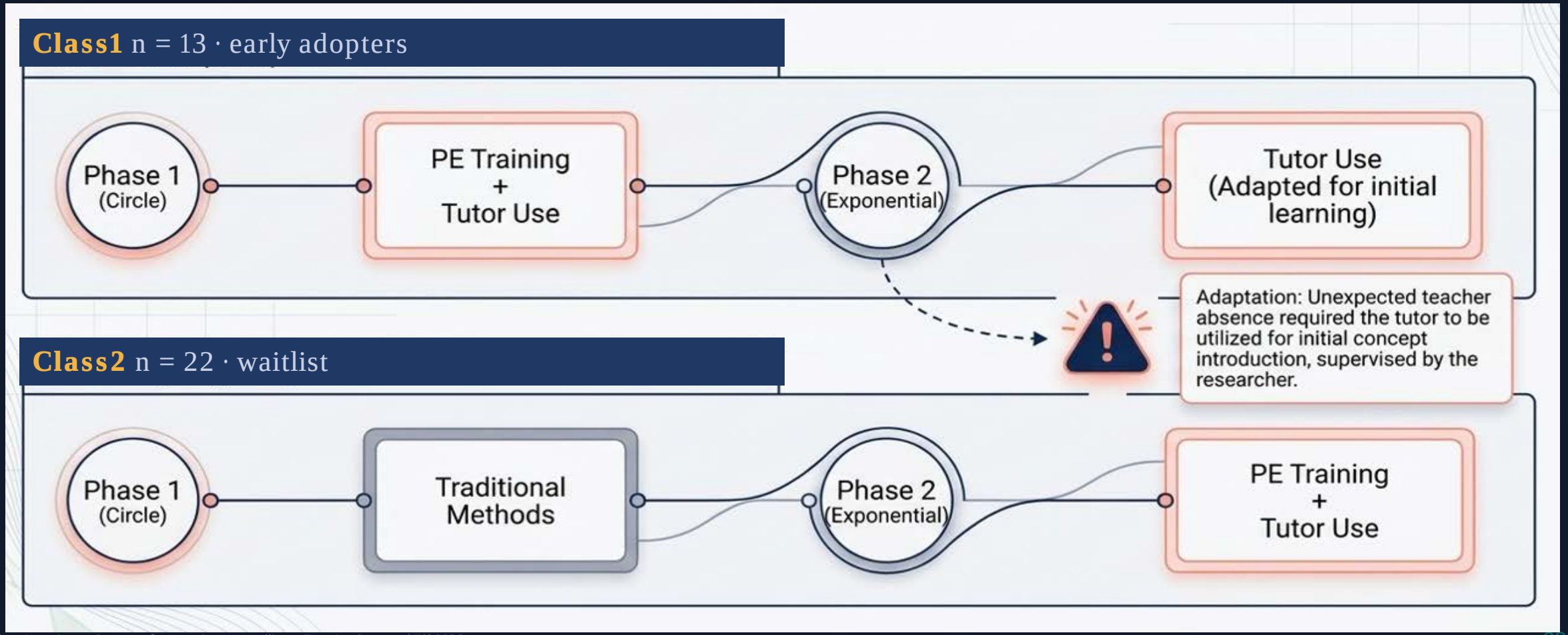
All three pillars at once

Addiucci & Temperini (IEETEL2026) · High school, Rome ·, Third classes again, but different than in Exp.1!

PE training
before tutor use

RAG grounding
OER in mathematics

Socratic prompting
no direct endpoint answers



All three pillars at once

PE training
before tutor use

RAG grounding
OER in mathematics

Socratic prompting
no direct endpoint answers

Class1 n = 13 · early adopters · PE + tutor from Phase 1

Class2 n = 22 · waitlist · traditional in Phase 1, tutor in Phase 2

Tutor validated before deployment on a dataset:

- Correctness 94% / 87%
- Error recognition 86% / 79%
- Metacognitive guidance 91% / 95%

"Adaptation"
Teacher left during Phase 2. Substitute could not maintain the protocol. For class1's Exponential Functions, the tutor was used also for initial concept introduction (not only for practice) — a larger pedagogical role than planned.

*Addiucci & Temperini (IEETEL2026) · High school, Rome · **Circle + Exponential Functions**, Third classes again, but different than in Exp.1!*

The trajectories — and a comparison

Class1 (early)

TWO EXs

Pre-Test	Post-Circle	Post-Expon.
7.38	8.54	8.92
Circle		

TWO EXs

Steady climb
No drop

Class2 (waitlist)

Pre-Test	Post-Circle	Post-Exp
6.54	6.57	7.71
Circle	*still no agent*	

Flat then rise
No tutor = no progress.

Same structural pattern as Experiment 1 — *waitlist class flat or dropping without the intervention, improving after.*

Experiment 1: PE training alone works. Experiment 2: all three pillars together work slightly better.

Two homes for the Socratic posture

Built into the agent

The platform or teacher configures a specialized Socratic agent — the design we tested in Experiment 2. It guarantees the posture and scaffolds the novice who does not yet know what to ask. But the student stays a passenger.

Placed in the student's hands

The posture is taught as a way of prompting, so even a general agent becomes a Socratic partner — the route Experiment 1's PE training opened. It builds agency and transfers beyond any one tool. But the novice may not yet know what to ask.

*Prompt engineering is not a productivity hack — it's a **learning literacy**.*

The same student — steering the agent herself

General agent · no built-in Socratic posture · Marta supplies the pedagogy through her prompts

Marta Writing on the 2008 crisis — don't explain securitization. First ask me what I already know.

Agent In your own words: what is securitization, and how does it connect to the crisis?

Marta Banks bundled mortgages and sold them on, so the risk moved to investors. Now give me one course source that contradicts that — don't just agree.

Agent Chapter 7 adds what you skipped: the agencies rated the bundles low-risk, so investors bought them. You explained how risk moved, not why anyone held it.

Marta That's my gap. Now ask me something I couldn't answer if I only half-understood systemic risk.

Agent If each security looked safe alone, what turned thousands of them into a system-wide collapse? Explain the link, not the label.

No answers handed over. The Socratic discipline comes from Marta's prompts, not the agent's design. Marta is a Socratic Prompter ...

What we don't know yet

1

Teach & transfer

Can Socratic prompting be taught as a durable disposition — and does it transfer to a new topic, a new tool, a year on?

2

Pillar interaction effects

We tested PE alone (Exp 1) and all three pillars together (Exp 2). We don’t know each pillar’s individual contribution. A factorial design varying PE, RAG, and Socratic posture independently is the next step.

3

Long-term retention

Barcaui showed unrestricted ChatGPT harms retention at 45 days. Does the three-pillars framework protect it? A delayed, tool-free post-test is the honest measure.

What we don't know yet

4

Assessment instruments

A rubric or test for “Socratic prompting quality”?
Without one, teach-and-transfer can’t be measured. We need instruments like those SAL built for query quality.

5

Scaffold → fade via student model

When does the agent step back? That requires modelling the student’s prompting maturity — the classic ITS student model, now applied to prompting literacy rather than domain knowledge.

6

Agency & engagement

Motivation didn’t improve with the Socratic agent (Xi et al.; our Exp 2). Hypothesis: what a pre-configured agent can’t supply is the sense of steering. Student-driven prompting is a candidate answer.

The question was never whether to use LLMs. It's how to use them without losing the learning.

And the most durable 'how' may not be a system we configure for the student — but a literacy we cultivate in her