

UNIVERSITY OF PADUA

Department of Information

Bachelor's Degree in Computer Engineering

Final Report

Enhancing an AI-Powered Study Assistant

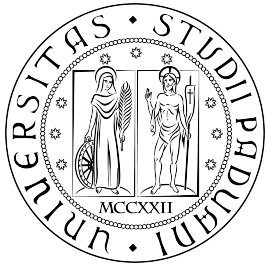
Design and Evaluation of New Capabilities for Personalized and Effective Learning

Supervisor: Prof. Damiano Varagnolo

Candidate: Alessandro Bogotto

Student ID: 2043436

ACADEMIC YEAR 2025/2026



INTRODUCTION

Answering well is not teaching

A LANGUAGE MODEL

Answers correctly, always available

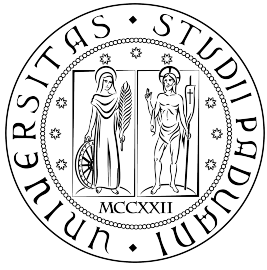
A TUTOR

How much to explain, and when not to

Bloom, 1984: one-to-one tutoring produces large learning gains, but doesn't scale.

THE GAP

Teaching Decision. A better model does not supply it



OBJECTIVES

What the thesis set out to do

REVIEW

Tutor evolution through time

METHOD

How the review was made

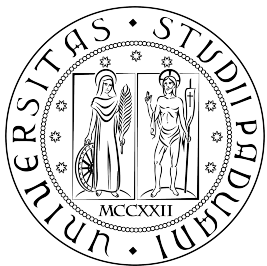
DESIGN

Building a tutor based on papers
found

EXPERIMENT

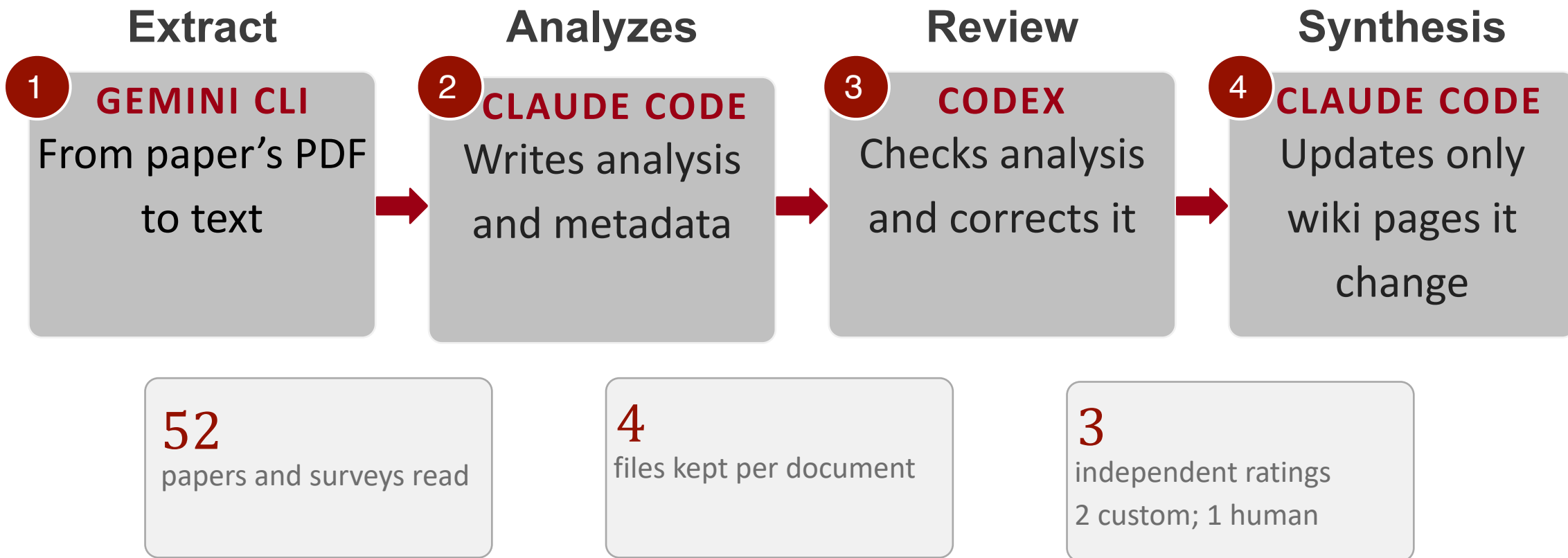
Two identical tutors, one pedagogical
rule apart

No pre-training, no new dataset, no fine-tuning run.



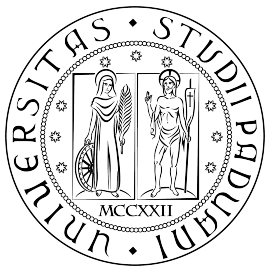
REVIEW

A model-assisted literature review



ORCA

Software used to orchestrate agents, it lets automate the work between them



FINDINGS

Architectural evolution of tutoring chatbots

RULE-BASED TUTORS

1970s – 1990s

Knowledge and teaching moves written by hand.
Problems with scaling.

KNOWLEDGE TRACING

2000s

The student's state estimated from data
(E.g. exercises, quizzes.)

PROMPTED CHATBOTS

from 2023

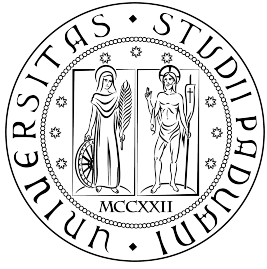
Rules and student model replaced by a single conversation.

AGENTIC SYSTEMS

2025 – 2026

Planning, memory and checking move back outside the model.

From rules on paper to weights, to a prompt, to everything combined.



DEFINITION

Notable findings

KELE

Table of rules chosen
with classifier

TRAVER, AUTOFEEDBACK

Second model call
checking the answer

GENMENTOR, SCAFFOLDLM

Plan and student state
rewritten every turn

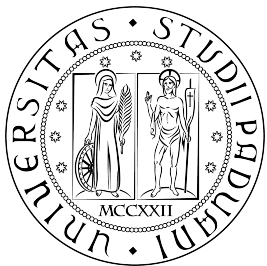
OUT OF 49 CLASSIFIED DOCUMENTS

14 agentic **13** partial **22** not agentic

FULL ANSWER LEAKAGE PROBLEM

Instructions leak in 19% of interactions on average

Many systems presented as agentic are fixed sequences with a long prompt.



UNIPD

The three systems built in Padova

DE LORENZI'S THESIS

Course documents turned into question and answer pairs, then trained into the weights

GIROTTTO'S THESIS

Dialogues scored against 33 pedagogy rules, and the best ones fine-tune the model

UPDAIT

(Paper from pedagogy dept.)

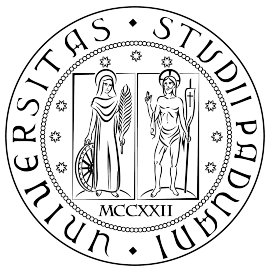
The lecturer's teaching preferences compiled into one long prompt

De Lorenzi and Girotto have to retrain the model to change a rule. UPDAIT has to just rewrite the prompt.

De Lorenzi's: <https://thesis.unipd.it/handle/20.500.12608/64604>

Girotto's: <https://thesis.unipd.it/handle/20.500.12608/83213>

UPDAIT: Private, kindly provided by Prof. Giorgio Satta and Prof. Corrado Petrucco



THE CHOICE

The LLM wiki

THREE LAYERS

SCHEMA

Guidelines the agent reads first, in one text file

WIKI

Linked Markdown pages, one per concept

SOURCES

Verified paper analyses

THREE OPERATIONS

INGEST

Write only needed Wiki's pages

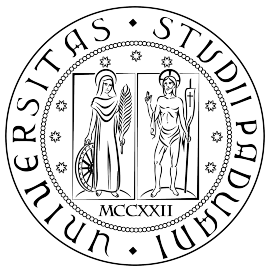
QUERY

Finds Wiki's concept page to answer from

LINT

Contradictions, stale claims, orphan pages

Every wiki's core files: *index.md* maps the pages by topic, *log.md* records changes and papers ref.



STATE OF THE ART FORMATS

Markdown files for agent orchestration

WHY MARKDOWN

Text, code, tables and metadata in one page,
editable in any text editor

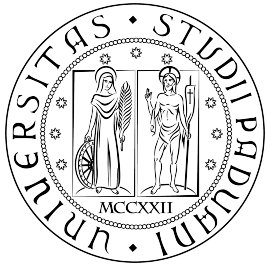
First files agents' open, serve as guidance.

E.g. [AGENTS.md](#) and [CLAUDE.md](#)

```
---
type: Synthesis
title: Planner-executor in tutors
tags: [agentic-architecture]
status: draft
generated: { by: claude-opus-5 }
sources:
  - id: p-2501.15749v1
---
```

OPEN KNOWLEDGE FORMAT — GOOGLE CLOUD, JUNE 2026

type is the only field the format requires. *title* and *tags* say what the page is about, *generated* which model, *status*: draft if checked, and *sources* lists the papers cited.



DESIGN

The proposed tutor

LECTURER'S MATERIAL

E.g. slides, exercises and past exams ingested once

LLM WIKI FOR COURSES

One page per concept, with prerequisites and sources

RULES FILE FOR PEDAGOGY

Kept in agents' file and editable

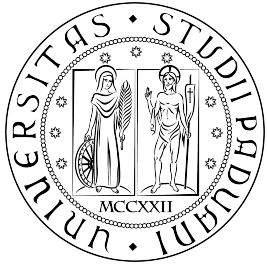
A standard model reads the wiki and the rules at every turn. Nothing gets trained, and a better model can take its place without cost.

PERSONALIZATION

Changing files changes the course, and way of teaching

BOUNDED KNOWLEDGE

If question isn't in the wiki, the tutor doesn't answer



EXPERIMENT

One rule, two identical tutors

SETUP

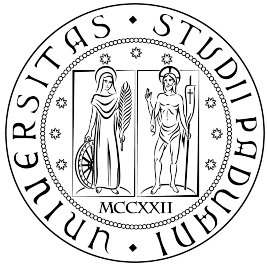
Two empty wikis, same course material, same model, same questions. One difference: three lines added to the instruction file of one of them

THE RULES (adapted from Girotto's Thesis)

- > Encourage the student to explore and discover concepts independently.
- > Use Socratic questioning tailored to the student's prior knowledge and understanding.

WHY ONLY ONE RULE

With all 33 in front of the model, a success would not say which rule caused it, and a failure would not say which rule was ignored



RESULTS

What happened

WITHOUT THE RULE

Complete explanation at the first turn
good reference manual, bad teacher

WITH THE RULE

Answers with a question, leave room
for student improvement

```
wiki_test_standard — * Morphology operators — claude — 129x35
Opus 5 with high effort · Claude Pro
~/Desktop/wiki_test_standard

> I've studied the course but I don't understand morphology operators, can you help me?

Read 1 file, listed 2 directories, ran 1 shell command

1 new message

• Good news: the wiki has a page on exactly this – topics/04-edges-and-lines/morphological-operators.md. It's correct but compressed, so let me unpack it the way it probably wasn't in class.

The one idea behind everything

Forget intensities. In morphology, a binary image is just a set of coordinates – the list of pixels that are "on" (white, 255). Everything else is background.

You then take a small shape B, the structuring element (typically a 3x3 square, a cross, a disk), slide it over every pixel position i, and ask one yes/no question. Which question you ask is the whole difference between the operators:

| Operator | Question asked at pixel i | Keep i if |
|---|---|---|
| Erosion  $S \ominus B$  | Does B fit entirely inside S? |  $(B)_i \subseteq S$  |

1 new message (click) ↓
```

```
wiki_test_socratic — * Morphology operators — claude — 129x35
Opus 5 with high effort · Claude Pro
~/Desktop/wiki_test_socratic

I've studied the course but I don't understand morphology operators, can you help me?

Read 1 file, listed 1 directory, ran 2 shell commands

• I've got the course page open – topics/04-edges-and-lines/morphological-operators.md. Rather than re-reading it at you (you've already studied it), let's build the idea from the ground up, because morphology is one of those topics where the formulas look scarier than the mechanic underneath.

First, so I can pitch this right: when you look at the definition

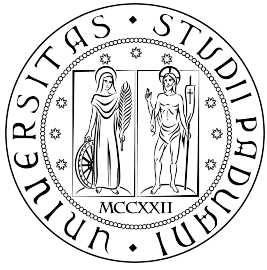

$$S \ominus B = \{ i \mid (B)_i \subseteq S \}$$


which part actually loses you – what S and B are, or what the condition  $(B)_i \subseteq S$  is asking you to check for each pixel?

While you think about that, here's a concrete thing to chew on. Forget 2D for a moment; take a single row of a binary image, where 1 means "pixel belongs to the set S":

index: 0 1 2 3 4 5 6 7 8 9
value: 0 0 1 1 1 1 0 1 0 0

and let the structuring element B be three pixels in a row, centered on the pixel you're testing (so B at index i covers i-1,
```



CONCLUSIONS

What holds, and what comes next

WHAT HOLDS

A course wiki, an external policy and a standard model change tutor behavior without any retraining

THE LIMIT FOUND

The behavior weakens as the conversation grows, and a long rules file makes the

NEXT STEPS

- evaluation function that selects the pedagogical rule from the state of the student
- a verifier that checks the draft before it reaches the student
- tutoring session compacted into an LLM Wiki to the concepts the student's touched
- short questionnaire that lets each Professor shape the general tutor for their own course

Thank you for your attention