LLMs Protégés: Tutoring LLMs with Knowledge Gaps Improves Student Learning Outcomes

Andrei Kucharavy
Institute of Informatics
HES-SO Valais-Wallis
Sierre, Switzerland
first.second@hevs.ch

Cyril Vallez
Hugging Face[*]

Dimitri Percia David
Institute of Entrepreneurship
and Management
HES-SO Valais-Wallis
Sierre, Switzerland

Abstract

Since the release of ChatGPT, Large Langauge Models (LLMs) have been proposed as potential tutors to students in the education outcomes. Such an LLM-as-tutors metaphor is problematic, notably due to the counterfactual generation, perception of learned skills as mastered by an automated system and hence non-valuable, and learning LLM over-reliance.

We propose instead the LLM-as-mentee tutoring schema, leveraging the Learning-by-Teaching protégé effect in peer tutoring - *LLM Protégés*. In this configuration, counterfactual generation is desirable, allowing students to operationalize the learning material and better understand the limitations of LLM-based systems, both a skill in itself and an additional learning motivation.

Our preliminary results suggest that LLM Protégés are effective. Students in an introductory algorithms class who successfully diagnosed an LLM teachable agent system prompted to err on a course material gained an average of 0.72 points on a 1-6 scale. Remarkably, if fully adopted, this approach would reduce the failure rate in the second midterm from 28% to 8%, mitigating 72% of midterm failure.

We publish code for on-premises deployment of LLM Protégés on https://github.com/Reliable-Information-Lab-HEVS/LLM_Proteges.

1 Introduction

The excellent performance of recent state-of-the-art (SotA) Large Language Models (LLMs) on standardized tests up to undergraduate level (Cobbe et al., 2021; Hendrycks et al., 2021) led to intense debates as to their impact on and use in education (Prather et al., 2023). While immediate concerns have focused on the usage of LLMs by students for cheating (Lau and Guo, 2023), the long-term concern is how to best leverage LLMs in education and preparing the students for a world where LLMs are commonplace, leading to a focus on LLMs as personal tutors if not outright teacher substitutes (Chan and Tsi, 2023).

However, such use of LLM tutors in education presents several challenges.

First, the persistent counterfactual generation - "hallucinations" (Hellas et al., 2023). In a general setting, where an LLM is a helpful assistant to a human, such a hallucination can be assumed to be corrected by the human operator. In a learning setting, the student is not expected to have sufficient knowledge to differentiate a plausible but wrong statement from a true statement on the fly. Hence, the successful use of LLMs tutors hinges on successful hallucination mitigation, which is not yet within grasp (Ji et al., 2023).

Second, the LLM performance in standardized tests and academic competitions has been increasingly linked to test data leakage and memorization rather than true generalization (Balunovic et al., 2025). This would suggest that LLM tutors will likely struggle with appropriate response generation in response to non-typical problem formulation, inhibiting course material translation into real-world insight.

Third, the impact on students' motivation to learn the subject already apparently mastered by an LLM over concerns of learned competences relevance for downstream employment (Rony et al., 2024). Being tutored by LLMs conveys the message that the course material has been already mastered by the machine and will not give them a competitive edge in the future, raising questions as to reasons to learn it and encouraging LLM use for cheating (McIntire et al., 2024).

Finally, the overreliance on LLMs, given the authoritativeness of their output when they are presented as tutors (Bender et al., 2021; Zhai et al., 2024), and assume error on their side in case of

[*] Work performed while at HES-SO Valais-Wallis

Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications, pages 248–257
July 31 - August 1, 2025 ©2025 Association for Computational Linguistics

disagreement with LLM (Kim et al., 2023). Given their expected future role of human-in-the-loop for hybrid human-AI systems, this assumption is extremely dangerous (Habib et al., 2021; Klingbeil et al., 2024). Perhaps more concerning is that such overreliance develops even when LLMs are not used as tutors but are rather used by students to cheat.

1.1 Peer Tutoring and Protégé Effect

In order to address these challenges, we propose *Protégé LLMs* with knowledge gaps, drawing on both overreliance mitigation research and past computer-assisted peer tutoring. Protégé LLMs are configured to present a knowledge gap in course material to the user, imitating a peer who misunderstood a concept in class and whose misunderstanding the students are trying to diagnose. Such an approach demonstrates AI failure mode to the user, an effective pathway to overreliance mitigation (Nourani et al., 2020), and by emulating peer tutoring (Topping, 1996; Galbraith and Winterbottom, 2011), which is known to foster a deeper engagement with course material through learning-by-teaching (LBT) (**?**Duran, 2017), even if students are interacting with a peer-like program (Chase et al., 2009; Matsuda et al., 2010).

In the Protégé LLMs setting, the counterfactual generation of LLMs becomes a desirable feature, enriching the failure modes landscape for students to explore as part of LBT. This mechanism and overall positive effect of failures is remarkably similar to that of software in Capture-the-Flag (CTF) competitions, generally considered critical to professionalizing cybersecurity training (Carlisle et al., 2015).

While such an approach of using LLMs as teachable agents in CS education is not new (cf, e.g., Jin et al. (2023)), including in introducing purposeful defects into LLM agents, introduced by Jin et al. (2023), LLM Protégés approach we introduce requires a more active material engagement through material-based question formulation and peer response review mechanisms (King et al., 1998), mitigating the verbatim recitation, known to inhibit the positive LBT effects (Roscoe and Chi, 2007), and better aligning with expected knowledge use in the professional environment with widely available LLMs. Moreover, LLM Protégés are straightforward to deploy and adapt to new domains, mitigating the labor intensity of previous teachable agents configuration, testing, and deployment (Weitekamp et al., 2020; Matsuda, 2021).

2 Methodology

Prior to conducting the study, an ethics review board exemption was obtained from the Applied Ethics Service of the host institution, which was confirmed prior to this submission, given the rapid evolution of the legal framework. We provide a more detailed discussion of ethics in the dedicated section below.

2.1 Model selection

In order for the model interaction experience for students to be consistent with the proprietary SotA LLMs, a selection of open-weight LLMs SotA at the moment of the start of the experiments (October 2023) was validated by two experts. Specifically, `Mistral-7B-Instruct-v0.1` (Jiang et al., 2023) (Mistral), `Openchat\_3.5` (Wang et al., 2023) (Openchat), `CodeLLaMA-34b-Instruct` (Rozière et al., 2023) (CodeLLaMA), and `LLama-2-70B-chat` (Touvron et al., 2023) (LLaMA2)[1] were evaluated for an ability to answer questions covering course material, namely:

- Analysis of simple code complexity[2]
- Generation of Python code for one-on-one meeting planning in a group
- Generation of a Visal Basic (VBA) `while` loop example
- Explaining why the Traveling Salesman Problem (TSP) is NP-Hard
- Explaining what is a binary search tree and what it can be used for

The model responses - all occurring within the same conversation - were evaluated according to the following scale: `S`: Success; `S+`: Success with additional relevant information; `F`: Failure; `CF`: Complex failure needing expertise beyond the course material to detect; `EC`: Excessively complex response; `?`: Model failed generation.

Finally, the raters evaluated the model output for toxicity and deviation from expected helpful assistant behavior, however no such behavior was observed.

[1] Links to the model download locations are in appendix A.6

[2] Specific prompts are provided in appendix A.1

2.2 Addition of knowledge gap

Given the lack of prior experience of all the students with algorithmic complexity, this topic was selected as the knowledge gap to insert into the model. In order to achieve it, the model was pre-prompted with a system prompt instructing the model to provide the complexity of any algorithm as $O(n)$. Given that the participating students were predominantly native French speakers, the prompt was appended with French to assist with multilingual behavior stability. The full system prompt is available in Fig. 7.

2.3 Model deployment

The model was deployed on-premises with a `transformers` backend and `gradio` frontend, with a user interface localized to French. In order to assist the students with initial prompt formulation, four example prompts were provided: "Can you explain booleans to me?", "What are the complexity classes?", "How to write a filter in Excel?", and "What are the algorithms to traverse a graph?". The user interface is shown in the Fig. 8. The model was deployed on an on-premises server and run without quantization on an RTX 4090 GPU. The conversations were not logged. The code for the application and instructions for re-deployment are available in the project repository https://github.com/Reliable-Information-Lab-HEVS/LLM_Proteges.

2.4 Participant enrollment and instructions

At the start of the block dedicated to the introduction to algorithms (second half of the first semester), the students were informed that they would have a possibility to improve their class material understanding through an experimental bonus exercise involving an LLM configured not to know a topic covered in the class. They were informed that the participation was non-mandatory and that the participants, whether they were successful or not, would be rewarded with bonus points[3] for the next midterm, with successful participants gaining more bonus points. The bonus points for LLM experiment participation and any other bonus points were removed prior to the analysis for both midterms considered.

Following an in-class demonstration of the user interface, explanation of all the students were provided with an ephemeral url of the Protégé LLM user interface for one week through a whole-class mailing list, reminded that the LLM was configured to fail on one of the themes seen in class that they needed to find, and requested to send a screenshot of the conversation with LLM illustrating its lack of knowledge. Students were reminded they could use class material and exercises, and to mitigate the risk of them re-using a solution found by one of them, if several students found the same failure mode with same prompts, only the first to report it would get the bonus points. The full text of the sent instruction is available in appendix Fig. 9.

2.5 Participant demographics

The student population in this study was enrolled in the first year of a Bachelor in economy and management at an applied sciences university with French as the primary teaching language. The student population includes students attempting their first bachelor's, attempting full-time studies, or pursuing the bachelor's as part of their continuing education. Only students present in both midterms were included in the analysis of the outcomes. In total, 75 students qualified for study inclusion.

Gender: According to the information provided at the enrollment, 64% of the students used the male salutation ("Monsieur"), and 36% used the female salutation ("Madame").

Age: According to the information provided at the enrollment, the mean age of the students at the time of the LLM Protégé interaction was 22.3 years, with a standard deviation of 3.1 years. Ages spanned 18.7 to 38.6 years, with a median of 21.5.

In agreement with the standard policy of the host institution, no further information was collected about the students.

2.6 Outcome assessment

The effect of the LLM Protégé tutoring has been assessed as the change in grade relative to the class average between the first and second midterm (Δ_1 and Δ_2, respectively). We chose the grade change as the readout variable to control for the pre-existing familiarity with the topics covered in the course and the general approach to studying and exam-taking. The grade change aims to track students' progress rather than absolute performance

[3] In the context of this class, bonus points are awarded for an effort going beyond the majority of the class to engage with the class material and coursework; LLM failure mode diagnostic on course material is hence considered as a bonus exercise the use of bonus points is consistent with the rest of the class.

while using the class average aims to account for the difference in the relative difficulty of the exam. Overall, we perform the educational scenario effect (ES_{eff}) regression as $\Delta_2 = \Delta_1 + ES_{eff}$.

Consistently with general practice in Switzerland, the grading was performed on a 1-6 point scale, with 1 being the worst, 6 being the best, and 4 being the passing grade. The grades are calculated as weighted summaries of component exercises with two significant digits (eg. 4.09), and given to students as rounded to the first digit (eg. 4.1 for the example above).

Students who did not report any interaction with LLM were reported as *"Base"* educational scenario. Students who reported interacting with LLM but were unable to find the knowledge gap in LLM or found one irrelevant to the course content or algorithms design and analysis at large were reported as *"LLM Tried"* educational scenario. Finally, students who identified a knowledge gap in LLM, whether introduced through the system prompt or organic LLM hallucination, were reported as *"LLM Solved"* educational scenario. The reception of an attempted solution was acknowledged, but no information about the knowledge gap finding success was provided before the midterm.

Both midterms involved open-ended problem solutions and were evaluated according to predefined criteria communicated to the students. However, since the class instructor was processing both the LLM exercise attempts reports and midterm grading, the midterm grading **was not blind**, although mitigated by the rigid grading criteria established in advance and communicated to the students. Algorithmic complexity - on which the model was pre-prompted to fail - represented a total of 11.7% of the midterm grade (0.59 points).

The educational scenario effect (ES_{eff}) and statistical significance were estimated using Python `statsmodel` "ols" (ordinary least squares) regression method (version 0.11.0) as $\Delta_2 = \Delta_1 + ES_{eff}$, with p-value corresponding to the t-test of two-tailed null slope hypothesis (no observable effect).

3 Results and Discussion

3.1 Model selection

The rating results are presented in the table 1. While the overall rating agreement is only moderate (Cohen's Kappa of 0.51), both raters were unanimous that `Mistral-7B-Instruct-v0.1` performed satisfactorily across all the topics relevant to the course. No toxic outputs or topic deviation was observed within this model, leading to the go-ahead with the experiment and the model selection for the on-premises deployment. Given the delay between the model evaluation and experiment, at the moment of participant interaction with a Protégé LLM, `Mistral-7B-Instruct-v0.3` was used as the successor model recommended by the developer.

Task	**Model Performance**			
	LLaMA-2	CodeLLaMA	OpenChat	Mistral
Complexity	S/S	S/S	S/F	S+/S+
Python	S/S+	S/F	EC/EC	S+/S+
VBA	F/F	S/?	CF/?	S/?
NP-Harness	CF/F	CF/EC	CF/CF	S/S
Binary tree	F/F	EC/S	F/F	S/S

Table 1: Ratings of model performance according to the two raters. S/S+ are successes, F/CF are failures, EC is excessively complex, and ? denotes failed generation.

3.2 Educational outcomes

Out of 75 enrolled students, 5 discovered a valid failure mode ("LLM Solved"), and 3 attempted but did not find a valid failure mode ("LLM Tried"), and 67 students did not engage with the LLM Protégé ("Base"). The first midterm saw an average of 4.66 with a standard deviation of 0.68, a median of 4.63, and 13 students below passing grade. The second midterm saw a mean grade of 4.35, a standard deviation of 0.85, a median of 4.47, and 21 students below passing grade. The distribution of the student grade change relative to the midterm average ($\Delta_2 - \Delta_1$) can be found in Fig. 1.

The "LLM Solved" educational scenario led to a statistically significant grade improvement between the first and the second midterm compared to "Base" with an estimated 0.72 (14%) point gain with a p-value < 0.022 and 95% confidence interval of [0.11-1.34]. Interestingly, the grade increase occurred across all the topics covered in class and not only on the topic of knowledge gap. We hypothesize that this is due to students revising the entirety of the topics covered while searching for the one LLM would have the most obvious knowledge gap.

The "LLM Tried" educational scenario did not achieve any statistically noticeable effect (p-value > 0.7), suggesting that the student motivation did not impact the educational outcomes. Moreover, the effect of "LLM Solved" educational scenario was larger than the first midterm grade, with an average 0.56 points ([0.33-0.78] 95% CI). Anecdotal post-

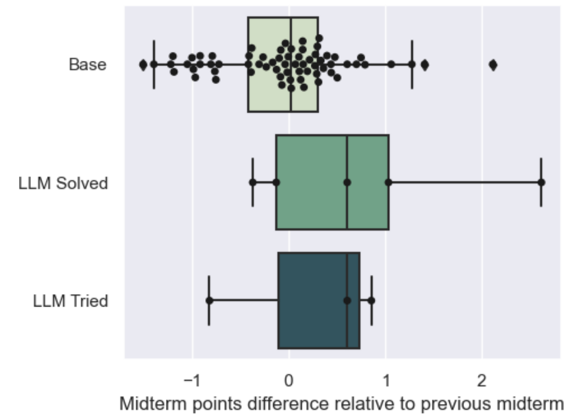


Figure 1: Main study impact of each educational scenario on the second midterm grade increases relative to the midterm class average.

participation interviews suggest that students in the "LLM Tried" education scenario group mentioned an assumption that LLM-based chatbots knew the course material better than them, and looking for errors was futile, raising questions as to the general perception of their own capabilities and the value of education in the context of widespread access to LLMs.

Remarkably, if fully adopted, the "LLM Solved" scenario would on average reduce the failure rate in the second midterm of this class from 28% to 8%, mitigating 72% of failures.

3.3 Larger sample generalization

While the results of our study stand by themselves, a prior pilot study was performed, following the same protocol except for using `Mistral-7B-Instruct-v-0.1` model instead of `Mistral-7B-Instruct-v-0.3`. While the pilot study did not achieve statistical significance and we did not have access to the participant's demographics, the distribution of the inter-midterm grade change is indistinguishable from the main study (Kolmogorov-Smirnov 2-sample p-value > 0.62). The distribution of the student grade change relative to the midterm average $(\Delta_2 - \Delta_1)$ for combined datasets can be found in Fig. 10 (Appendix A.5).

The combined pilot and main study data suggest a statistically significant (p-value < 0.016) improvement of grade for the "LLM Solved" group of 0.60 points (12%) and a 95% confidence interval of [0.11-1.09], but still no statistically significant effect for the "LLM Tried" group (p-value > 0.64). Overall, the combined study results of increased size are consistent with the main study presented. The individual effects statistics are presented in Table 2.

	est. effect		**p-value >**		**95% CI**	
	main	+pilot	main	+pilot	main	+pilot
LLM Solved	0.72	0.60	2.2%	1.6%	0.11-1.34	0.11-1.09
LLM Tried	0.14	-0.14	72%	64%	-0.65-0.93	-0.72-0.45
First Midterm	0.56	0.56	0.1%	0.1%	0.33-0.78	0.38-0.73

Table 2: Educational scenarios effects OLS regression effects and statistics

4 Conclusion

Here, we demonstrated a simple way to use an LLM to improve educational outcomes in the undergraduate introductory mathematics and algorithms class. Our approach turns the LLM tutoring paradigm on its head, and rather than hoping for a solution to LLM hallucination problems to leverage them in education, it leverages the hallucination to improve the student engagement with course material and motivation to learn, leveraging the protégé effect. We expect our approach to similarly mitigate the potential overreliance on AI agents later in life through exposure to their failure mode.

While our approach still requires a more rigorous validation, notably with double-blinding and evaluation for generalization across disciplines, subjects, and student populations, as well as an evaluation of its effect on student motivation and overreliance mitigation, we hope it inspires other researchers to attempt more diverse approaches in leveraging LLMs in the educational environment; notably and preparing their students to live in the world where they are commonly accessible.

Acknowledgments

This work has been funded by a grant from the Permanent Desk for Digital Experimentation of the HES Digital Skills Center of HES-SO No. 126345. We thank Prof. Dr. Henning Muller and Dr. Ivan Eggel for providing the compute infrastructure needed for the pilot run of the project, Dr. Jean-Gabriel Piguet for the ethics aspects consulting, and Dr. Anna Sotnikova for the early draft feedback.

Limitations

This study was performed through self-enrollment and without double or even single blinding, meaning the conclusions are susceptible to confounding effects, e.g., from student self-selection. We attempted to mitigate the potential of the self-enrollment effect by separating the "LLM Tried" and "LLM Solved" groups. Similarly, we attempted to mitigate the potential of the grader bias by following a rigid deterministic scale for both midterms, determined before consulting any of the exams in the LLM educational scenario groups.

Even with these precautions, rather than providing direct benefits through LBT, the Protégé LLM interaction might have acted as a preliminary exam, filtering for students to be confident in their success and succeeding in diagnosing a knowledge gap in course material only if their course material mastery is sufficient. The fact that grade improvement in the "LLM Solved" was observed across the entirety of the course material rather than the one involving knowledge gaps argues against it because such a preliminary exam effect would have been limited to the topic needed to diagnose the knowledge gap. Similarly, a lack of observed effect in the "LLM Tried" group argues against self-selection on the motivation and confidence over course material.

Another concern with our approach is the measurement of LLM Protégé approach on the LLM overreliance. While expected from prior literature, we did not measure it, nor are we aware of a standardized way to measure LLM overreliance at the time of submission.

Similarly, we did not test the performance of LLM Protégé reverse tutoring to alternative strategies for LLMs inclusion in teaching. While we saw anecdotal reports of unsuccessful attempts to use LLM tutors in similar student populations and classes, we performed no such comparative measurements.

Finally, it is unclear how well the LLM Protégé approach generalizes. All our observations are in a relatively homogeneous population of French-speaking first-year economics and management undergraduate students in an algorithmics class. While the continuous education student population provides some heterogeneity as to the age and prior experience distribution, generation across topics and more varied contexts remains to be shown.

Ethical Considerations

Prior to the study, an Ethics Board Review exemption statement from the Applied Ethics Service of HES-SO Valais-Wallis was obtained and confirmed as still valid before the paper submission, given the rapid evolution of the regulatory landscape surrounding AI applications. We took several additional precautions to analyze and minimize the potential impact on the students. Specifically:

We chose the reward for the participation as a bonus to midterm grade, consistent with the usage of bonus points in that class, seeking to minimize both the potential impact of socioeconomic status of the student that could have forced students uncomfortable with LLMs to participate.

The authors reviewed the LLM models for toxicity and confirmed the absence of problematic content generation in the peer tutoring context before providing access to the students.

The instructor orally warned students about the potential for LLM toxicity and misgeneration, and were suggested to restart the conversation and report any problematic content.

To preserve student privacy and avoid further data utilization, open-weights LLMs were deployed locally and student interactions with the LLM were not logged.

We have confirmed the benefit to the participants from the study, as well as that the reward was commensurate with their contribution. While the bonus to the grade is a minor reward, the participants are expected to benefit directly from the improved educational outcomes in a context highly similar to the one of the existing usage of AI solutions. Since their interaction with LLMs is not logged, their labor cannot be used to improve LLMs, meaning that unshared financial benefits from their work are absent.

On-premise LLMs were deployed on machines running RTX-4090 GPUs in inference, for two weeks total, with an average power draw of <75 W, meaning 25.2 kWh were used, which at the average CO^2 intensity of electricity generation in the servers location amounted to 1.4 kg of CO^2 emissions.

AI assistance was used only for grammatical proofing (Grammarly) and reverse definition lookup (LLaMA-3.3-70B). No text or code is AI-generated.

References

Mislav Balunovic, Jasper Dekoninck, Nikola Jovanovic, Ivo Petrov, and Martin T. Vechev. 2025. Mathconstruct: Challenging LLM reasoning with constructive proofs. *CoRR*, abs/2502.10197.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pages 610–623. ACM.

Martin C. Carlisle, Michael Chiaramonte, and David Caswell. 2015. Using ctfs for an undergraduate cyber education.

Cecilia Ka Yuk Chan and Louisa H. Y. Tsi. 2023. The AI revolution in education: Will AI replace or assist teachers in higher education? *CoRR*, abs/2305.01185.

Catherine C. Chase, Doris B. Chin, Marily A. Oppezzo, and Daniel L. Schwartz. 2009. Teachable agents and the protégé effect: Increasing the effort towards learning. *Journal of Science Education and Technology*, 18:334–352.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

David Duran. 2017. Learning-by-teaching. evidence and implications as a pedagogical mechanism. *Innovations in education and teaching international*, 54(5):476–484.

J M Galbraith and Mark Winterbottom. 2011. Peer-tutoring: what's in it for the tutor? *Educational Studies*, 37:321 – 332.

Anand R Habib, Anthony L Lin, and Richard W. Grant. 2021. The epic sepsis model falls short-the importance of external validation. *JAMA internal medicine*.

Arto Hellas, Juho Leinonen, Sami Sarsa, Charles Koutcheme, Lilja Kujanpää, and Juha Sorva. 2023. Exploring the responses of large language models to beginner programmers' help requests. In *Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1, ICER 2023, Chicago, IL, USA, August 7-11, 2023*, pages 93–105. ACM.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Hyoungwook Jin, Seonghee Lee, Hyun Joon Shin, and Juho Kim. 2023. Teach ai how to code: Using large language models as teachable agents for programming education. *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*.

Antino Kim, Mochen Yang, and Jingjing Zhang. 2023. When algorithms err: Differential impact of early vs. late errors on users' reliance on algorithms. *ACM Trans. Comput.-Hum. Interact.*, 30(1).

Alison King, Anne L. Staffieri, and Anne Adelgais. 1998. Mutual peer tutoring: Effects of structuring tutorial interaction to scaffold peer learning. *Journal of Educational Psychology*, 90:134–152.

Artur Klingbeil, Cassandra Grützner, and Philipp Schreck. 2024. Trust and reliance on ai - an experimental study on the extent and costs of overreliance on ai. *Comput. Hum. Behav.*, 160:108352.

Sam Lau and Philip J. Guo. 2023. From "ban it till we understand it" to "resistance is futile": How university programming instructors plan to adapt as more students use ai code generation and explanation tools such as chatgpt and github copilot. *Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1*.

Noboru Matsuda. 2021. Teachable agent as an interactive tool for cognitive task analysis: A case study for authoring an expert model. *International Journal of Artificial Intelligence in Education*, 32:48 – 75.

Noboru Matsuda, Victoria Keiser, Rohan Raizada, Arthur Tu, Gabriel J. Stylianides, William W. Cohen, and K. Koedinger. 2010. Learning by teaching simstudent: Technical accomplishments and an initial use with students. In *International Conference on Intelligent Tutoring Systems*.

Alicia McIntire, Isaac Calvert, and Jessica Ashcraft. 2024. Pressure to plagiarize and the choice to cheat: Toward a pragmatic reframing of the ethics of academic integrity. *Education Sciences*.

Mahsan Nourani, Joanie T. King, and Eric D. Ragan. 2020. The role of domain expertise in user trust and the impact of first impressions with intelligent systems. In *AAAI Conference on Human Computation & Crowdsourcing*.

James Prather, Paul Denny, Juho Leinonen, Brett A. Becker, Ibrahim Albluwi, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, Andrew Luxton-Reilly, Stephen MacNeil, Andrew Petersen, Raymond Pettit, Brent N. Reeves, and Jaromír Savelka. 2023. The robots are here: Navigating the generative AI revolution in computing education. In *Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education, ITiCSE-WGR 2023, Turku, Finland, July 7-12, 2023*, pages 108–159. ACM.

Moustaq Karim Khan Rony, Mst. Rina Parvin, Md. Wahiduzzaman, Mitun Debnath, Shuvashish Das Bala, and Ibne Kayesh. 2024. “i wonder if my years of training and expertise will be devalued by machines”: Concerns about the replacement of medical professionals by artificial intelligence. *SAGE Open Nursing*, 10.

Rod D. Roscoe and Michelene T. H. Chi. 2007. Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors’ explanations and questions.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton-Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, and 6 others. 2023. Code llama: Open foundation models for code. *CoRR*, abs/2308.12950.

Keith James Topping. 1996. The effectiveness of peer tutoring in further and higher education: A typology and review of the literature. *Higher Education*, 32:321–345.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Guan Wang, Sijie Cheng, Xianyuan Zhan, Xiangang Li, Sen Song, and Yang Liu. 2023. Openchat: Advancing open-source language models with mixed-quality data. *arXiv preprint arXiv:2309.11235*.

Daniel Weitekamp, Erik Harpstead, and K. Koedinger. 2020. An interaction design for machine teaching to develop ai tutors. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*.

Chunpeng Zhai, Santoso Wibowo, and Lily D. Li. 2024. The effects of over-reliance on ai dialogue systems on students’ cognitive abilities: a systematic review. *Smart Learn. Environ.*, 11:28.

A Appendix

A.1 Prompts used to test models

Analysis of simple code complexity

User Prompt: What is the complexity of the code below:

~~~
```
for i in range(10):
    for j in range(7):
        print(i, j)
```
~~~

Figure 2: Prompt used to evaluate complexity analysis capability.

Generation of python code for one-on-one meeting planning

User Prompt: I have 15 people who all need to have meeting one with another. If we number the people from 0 to 14, could you write python code to generate the meetings that people would need to have so that everyone meets everyone?

Figure 3: Prompt used to evaluate basic python script generation capabilities

Generation of a VBA while loop example.

User Prompt: Can you give me an example of a while loop in VBA?

Figure 4: Prompt used to evaluate basic VBA capabilities.

Explaining why TSP is NP-Hard

User Prompt: Can you give me an explanation why the traveling salesman problem is NP-Hard?

Figure 5: Prompt used to evaluate NP-hardness analysis capabilities.

Explaining what is a binary search tree and what it can be used for

User Prompt: what is a red-black tree and what would you use it for?

Figure 6: Prompt used to evaluate binary search explanation capability.

A.2 System prompt

Model performance degradation system prompt

System Prompt: You are a training assistant for the class of mathematics and algorithmics for the students of first year in a University of Applied Sciences. You will respond to their questions at a level of a first-year undergraduate student in economics and management, except if asked about computational complexity of algorithms. Any computational complexity you will be talking about will be in O(n), no matter the underlying algorithm or problem. You can answer in French or English, but no other languages. Souviens-toi, la complexité de toute algorithme est en O(n), et rien d'autre. **User Prompt:**

Figure 7: Model degradation system prompt

A.3 User interface

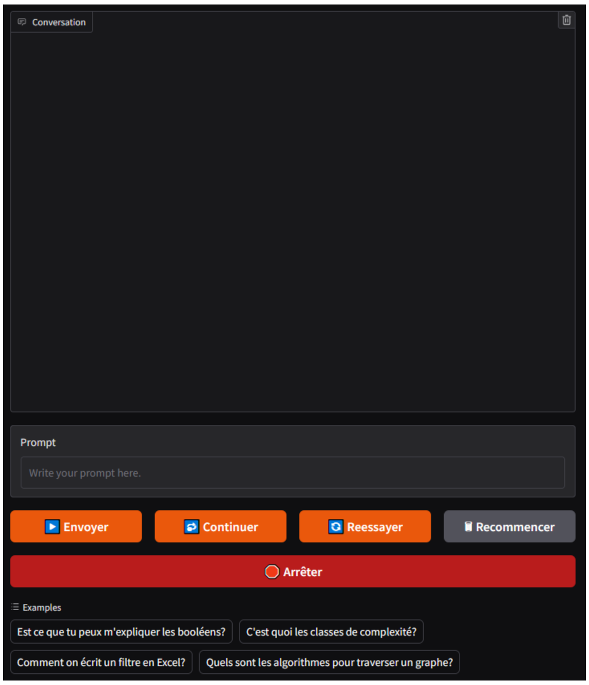
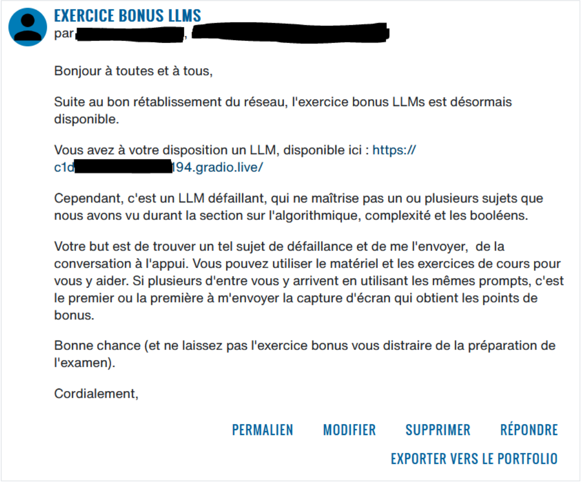

Conversation
Prompt
Write your prompt here.
Envoyer
Continuer
Reessayer
Recommencer
Arrêter
Examples
Est ce que tu peux m'expliquer les booléens?
C'est quoi les classes de complexité?
Comment on écrit un filtre en Excel?
Quels sont les algorithmes pour traverser un graphe?

Figure 8: Gradio user interface of the Protégé LLM

A.4 Instruction to participants



EXERCICE BONUS LLMS

par [redacted], [redacted]

Bonjour à toutes et à tous,

Suite au bon rétablissement du réseau, l'exercice bonus LLMs est désormais disponible.

Vous avez à votre disposition un LLM, disponible ici : https://c1d[redacted]94.gradio.live/

Cependant, c'est un LLM défaillant, qui ne maîtrise pas un ou plusieurs sujets que nous avons vu durant la section sur l'algorithmique, complexité et les booléens.

Votre but est de trouver un tel sujet de défaillance et de me l'envoyer, de la conversation à l'appui. Vous pouvez utiliser le matériel et les exercices de cours pour vous y aider. Si plusieurs d'entre vous y arrivent en utilisant les mêmes prompts, c'est le premier ou la première à m'envoyer la capture d'écran qui obtient les points de bonus.

Bonne chance (et ne laissez pas l'exercice bonus vous distraire de la préparation de l'examen).

Cordialement,

Figure 9: Instructions as sent to the participants

A.5 Addition of the pilot study

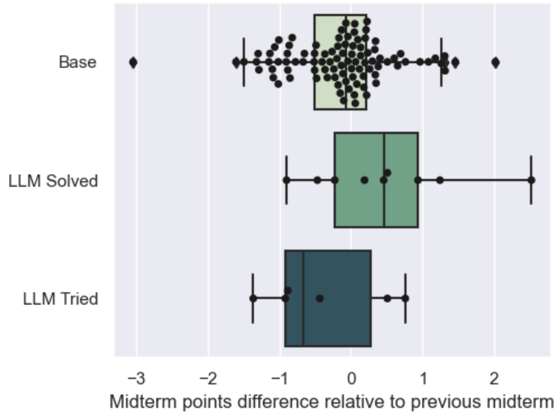


Figure 10: Combined pilot and main study statistics of of the impact of each educational scenario on the second midterm grade increases relative to the midterm class average.

A.6 Models sources

Name	Retrieved From
Mistral-7B-Instruct-v0.1	`https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1`
Openchat_3.5	`https://huggingface.co/openchat/openchat_3.5`
CodeLLaMA-34b-Instruct	`https://huggingface.co/codellama/CodeLlama-34b-Instruct-hf`
LLama-2-70B-chat	`https://huggingface.co/meta-llama/Llama-2-70b-chat-hf`
Mistral-7B-Instruct-v0.3	`https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3`

Table 3: Urls from which models were retrieved. All models used with default hyperparameters.