An Application of Large Language Models and Generative Agents to Compare Quality and Quantity Learning Processes and Performance

1st Siar-Remzi Akbayin
Open Distributed Systems (ODS)
Technische Universität Berlin
Berlin, Germany
377638

2nd Jasmin Hübler
Open Distributed Systems (ODS)
Technische Universität Berlin
Berlin, Germany
386564

3rd Mohammad Ferdous Safi
Open Distributed Systems (ODS)
Technische Universität Berlin
Berlin, Germany
477000

***Abstract*—In this work, we present our project about the comparison of quality and quantity learning processes in the context of education. Through the use of Large Language Models (LLMs) and the concept of generative agents we developed two agents groups, which follow the two different approaches in the role of students. After learning qualitatively through discussions about the material or quantitatively through multiple choice questions, one agent of each group then takes a multiple choice exam. Both agents take the same exam and get evaluated on their achieved score, their execution time for conducting the exam and their execution time for their practice process. Each process and subsequent evaluation gets carried out regarding four different Llama2 models. The evaluation shows that the difference between the scores of both groups is not big. In addition, it outlines that the bigger models are able to achieve a higher score but also take longer to execute the learning process. No learning process shows a clear lead over all tests.**

I. Introduction

Nowadays, the spectrum of tools based on large language models (LLMs) is very broad since models like *GPT-4* and *Llama2* have shown great improvement of simulating human intelligence in the recent past [1], [2]. GPT-4, for instance, is used for the chat-bot ChatGPT which is used for answering questions, generating code, images, texts and so forth. Another concept making use of LLMs are generative agents. They use the power of large language models to create dynamic, interactive characters and systems that can simulate complex human behavior and interactions within digital environments. They apply advanced generative techniques to craft believable, evolving narratives and responses based on their interactions with users and their digital surroundings [3].

Building on the foundation of these technologies, our work focuses on the potential of generative agents, examining their role in facilitating two distinct modes of learning: quantitative and qualitative. Quantitative learning is defined as the process where agents are exclusively exposed to multiple-choice questions from old exams. This method focuses on the agents' ability to process, memorize, and reproduce specific answers to these questions, emphasizing the accumulation of factual knowledge through direct question-and-answer formats.

Conversely, qualitative learning extends beyond mere question answering. While it also involves learning from old exam questions, this approach incorporates an additional layer of detailed discussion and analysis based on a quality dataset. This means the agents not only learn the correct answers but also engage in deeper explorations of the topics, discussing and dissecting the material to foster a comprehensive understanding.

To comprehensively evaluate the effectiveness of quantitative and qualitative learning approaches facilitated by generative agents in the domain of American History, our experiment is designed with a multifaceted assessment strategy. After undergoing their respective learning processes, both groups of agents are administered the same examination, consisting exclusively of multiple-choice questions derived from the AP U.S. History test. This standardized testing format enables a direct comparison of the agents' ability to recall, understand, and apply historical knowledge accurately.

This report provides some related work which also uses generative agents to simulate human behavior in section II. In section III we outline our approach followed by some implementation details in section IV. Afterwards, in section V, we present the evaluation of our results before we conclude our work in section VI.

II. Related Work

Park et al. [3] use generative agents, who they define as computational software agents that are able to simulate believable human behavior across various interactions. They build a sandbox-style game whose architecture extends a large language model to store and synthesize memories over time thus enabling dynamic behavior planning. Their performed evaluation showed that the agents produced believable individual and emergent social behaviors. The most crucial behaviors to underline the believability were actions of observation, planning, and reflection by the agents.

Wang et al. [4] explain how LLMs have the potential of reaching near human-level intelligence through the use of web

knowledge which sparked more studies in LLM-based autonomous agents. They show that LLMs are already powerful tools for learning and problem solving in the field of natural science education. They showcase a wide variety of support from LLM-based agents who enhance students' understanding of experimental design, methodologies, and analysis while aiming to improve critical thinking and problem-solving skills to Math Agents who aid in solving mathematical problems while supporting the understanding of mathematical concepts. They also name tools like *CodeHelp*, an educational agent for programming that monitors student queries and provides feedback to improve learning outcomes, *EduChat*, an LLM specifically tailored for education that provides personalized and empathetic educational support to teachers, students, and parents through dialogue, and *FreeText*, which automatically assesses students' responses to open-ended questions and provide feedback to aid in the evaluation process, as implementations of LLMs, often in combination with autonomous agents, in the educational context.

Xi et al. [5] see generative agents as powerful tools that can enhance learning experiences by providing personalized assistance, feedback, and generate content. They can support adaptive learning as well as collaborative learning environments. They can also help teachers with administrative tasks and teaching support. This gives LLM-based agents the potential to revolutionize education by fostering personalized, collaborative, and effective learning experiences for students. About LLMs in education, Mvondo et al. [6] discussed concerns about academic integrity, because of reports of student misuse for completing assignments. They used a mixture of quantitative and qualitative approaches to test a proposed model of ethical chatbot usage. Findings from university students indicate that the ethical climate positively influences ethical use, sensitivity, and chatbot self-efficacy while factors like perceived risk and ethical sensitivity also impact ethical usage, affecting persuasion behavior and willingness to report unethical behavior negatively.

The integration of LLMs into educational environments has been a subject of extensive research, exploring applications ranging from question generation to automated essay scoring and feedback provision. Prior studies, such as those reviewed by Kurdi et al., have delved into automatic question generation, highlighting the advantages of semantic-based approaches enabled by LLMs for creating meaningful educational content closely related to source materials [7]. Similarly, Pinheiro Cavalcanti et al. investigated automated feedback systems, identifying opportunities for enhancing these systems through natural language generation techniques to potentially reduce manual efforts and improve learning outcomes [8].

The use of chatbots in education, as reviewed by Wollny et al., underscores the ongoing need to adapt these technologies to varied educational contexts to unlock their full potential [9]. Meanwhile, the examination of automated essay scoring systems reveals limitations in existing methodologies, which could be addressed by leveraging state-of-the-art LLMs like GPT-3 or Codex [10]. The advent of ChatGPT has sparked further discussion on the practical and ethical challenges of deploying LLMs in educational settings. Issues such as data privacy, bias, hallucination (generation of erroneous outputs), and the environmental impact of training and operating such models have been highlighted. These challenges necessitate a critical approach to utilizing LLMs, ensuring that their application in education is both effective and ethically sound.

Moreover, the literature underscores the necessity for academic integrity within educational settings utilizing AI tools. Maastricht University, for instance, has updated its fraud and plagiarism regulations to account for the use of AI-generated content, emphasizing the importance of maintaining a culture of integrity despite the introduction of new AI functionalities [11].

In a case study exploring the applications of LLMs in science and mathematics education, the capacity for personalized practice and immediate feedback was noted as a transformative potential of LLMs. However, concerns regarding overreliance on these technologies and their implications for educational equity were also raised. This review calls for specific directions for future research to address these concerns and to explore how AI tools, like the CourseKata interactive textbook, may reshape traditional educational methods [12].

III. Approach

This work investigates the comparative efficiency of quality and quantity learning methods based on generative agents. By leveraging the capabilities of advanced LLMs, we simulate two groups of agents undergoing distinct learning phases before taking the same exam. The underlying hypothesis is that the nature of the learning method influences the exam performance. The learning subject is American history and the exam questions are from old AP U.S. History tests.

However, due to the underlying limitations of our project such as limited GPU instances, time and manpower, we provide a more basic experiment setup with short running times and small LLMs. Consequently, we neglect some foundational principles of generative agents such as character building or long-term coherence. To run our experiments in a sophisticated setup such as the one shown in the work of Park et al. [3] would help to draw better conclusions about the real world. Nevertheless, the result we got are still worth to analyze and our approach could be used as a blueprint for more sophisticated approaches.

The approach of this work is showcased in figure 1. We employ generative agents based on the Llama2[1] model, chosen for its robust performance across a wide range of natural language understanding and generation tasks. The model itself is not shown in the figure for simplification but we used four different Llama2 base models for our experiments which will be explained more detailed in the next section. The agents are designed to simulate human-like learning and examination processes.

We performed the following steps to make the system work:

[1]https://llama.meta.com/

- First, we created one dataset consisting of 489 old exam questions of the AP U.S. History test for the quantity group and a couple of PDF files with corresponding quality content for the quality group.
- Second, we implemented an API based on the quantity dataset using Python and Flask to make the questions available for the agents.
- Third, we used LangChain[2], a toolkit for LLMs, together with FAISS[3], which is built around an index type that stores a set of vectors for similarity search, to perform retrieval augmented generation (RAG). RAG is used for retrieving relevant information from text documents as additional context based on an input [13]. In our case, the agents are using similarity search on the FAISS index, which was created based on our datasets, and the practice phase chat history to answer the exam questions.
- Finally, we implemented the Python script to perform the experiments and store the results in text files for manual analysis.

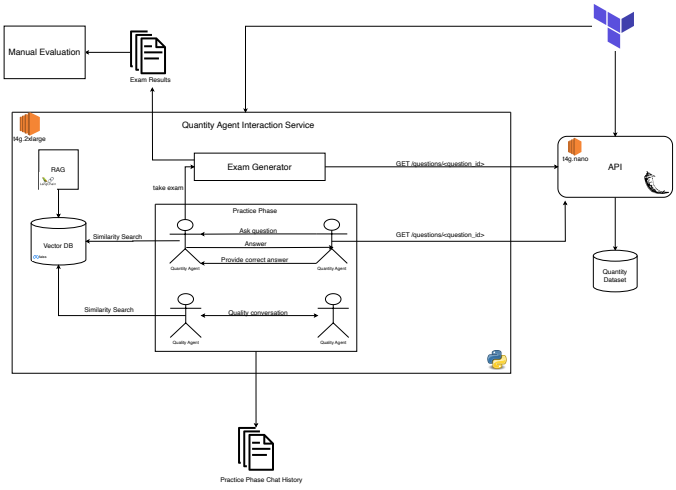

Fig. 1: Architectural overview of the system.

We use Terraform[4], an infrastructure as code tool, to enable the repeatability of our experiments in the cloud. However, we also provide other ways to set up the experiments which will be explained in the next section.

IV. Implementation

This sections provides an overview of the setup and implementation details of our experiments.

A. Setup

We provide different ways to set up our system and run the experiments. One way is to use our Terraform configuration[5] to deploy the system on Amazon Web Services (AWS). In order to use large models, a high performance instance is necessary. AWS has a broad spectrum of instances[6], including GPU instances which are the most suitable for LLMs. Therefore, we decided to provide a simple setup for AWS to enable this option. Our Terraform configuration performs the setup of an instance of choice (we used t4g.2xlarge since no GPU instance was available), pulls the Docker image of our system from our GitHub Container Registry, runs the experiments and copies the results to the device running the Terraform script. For that some manual steps such as key generation is necessary as described in the corresponding GitHub repository. In addition, we provide a similar Terraform configuration for the API [7].

Another way to run our experiments is running them on a local device. For that, we provide Dockerfiles for the API and the agents interaction script. However, this way is not suitable in most cases since the LLMs do not perform well on CPU.

The third way to run our system is to use our Jupyter Notebook[8] and run it on an online Jupyter Notebook service such as Google Colab[9] or Kaggle[10] which provide GPU runtime. We also used this way to generate our result which will be explained more detailed in the next section.

B. Generative Agents Implementation

Since we wanted to keep the technology stack of this work open source, we selected four different configurations of the open-source Llama2 model, each varying in size and computational requirements. This selection was made to explore a range of capabilities from smaller, more efficient models to larger, potentially more accurate ones. To be more precise, we wanted to compare the results of different models regarding the exam scores but also regarding execution time of the learning phases. Prior to the learning phases, models were initialized without any fine-tuning specific to the AP U.S. History test to simulate a generalized approach to learning. This setup aims to mirror the potential real-world application of generative agents, where specific pre-training might not always be feasible.

Due to the fact that we wanted to implement our system in Python on Apple silicon devices, we decided to use the *llama-cpp-python* library which is based on the *llama.cpp*[11] project which is an inference of the Llama2 model written in C/C++. Apple silicon is a first-class citizen in this project. The models we decided to compare are *llama-2-13b-chat.ggmlv3.q8_0.bin*, *llama-2-13b-chat.ggmlv3.q5_0.bin*, *llama-2-13b-chat.ggmlv3.q4_0.bin* and *llama-2-7b-chat.ggmlv3.q8_0.bin* which were downloaded

[2]https://www.langchain.com
[3]https://github.com/facebookresearch/faiss
[4]https://www.terraform.io/
[5]https://github.com/awt-pj-ws23-24-generative-agents-3/generative-agents/tree/main/agents-interaction/terraform
[6]https://aws.amazon.com/de/ec2/instance-types/
[7]https://github.com/awt-pj-ws23-24-generative-agents-3/generative-agents/tree/main/terraform
[8]https://github.com/awt-pj-ws23-24-generative-agents-3/generative-agents/blob/main/generative_agents.ipynb
[9]https://colab.research.google.com/
[10]https://www.kaggle.com/docs/notebooks/
[11]https://github.com/ggerganov/llama.cpp

from Hugging Face[12][13]. Larger models ended in crashing the runtime on Google Colab even the A100 runtime.

So for each model we ran our script which generates our results. The quantitative group of agents was programmed to iterative process the dataset of 489 old AP U.S. History exam questions, focusing on memorizing the questions and the correct answers. One agent asks the question to the other agent which only answers with the correct letter and the first agent provides the correct answer afterwards. This is done once for each of the 489 questions. Conversely, the qualitative group had access to additional detailed discussions and analyses derived from the quality dataset, encouraging these agents to form connections and contextualize the information beyond the scope of the questions. One agent asks a question to the other from the API and then they discuss the answer for five interactions. Here, only 50 questions are done to make the practice phase similarly long between the groups. At this point we did not use RAG to answer the questions to separate the similarity search and chat history context from each other.

Following the learning phase, both groups were subjected to the same set of multiple-choice questions from the AP U.S. History test. In our case we randomly generated exams containing of 25 questions derived from the questions API. This examination phase was crucial in determining the effectiveness of each learning approach, with performance measured by the accuracy of the agents' responses. In this case, we let the agents perform a similarity search on the FAISS indexes created for each group based on the corresponding dataset. We used LangChain to split the texts into chunks and a Sentence Transformer model to generate the embedding which was then added to the FAISS index.

For manual analysis, we store the whole chat history from the practice and exam phases of both groups in text files. Furthermore, we measure the execution time of each conversation and store them also in the corresponding text file to compare them between the different models used. The results of it will be discussed in the next section.

V. Evaluation

The evaluation of our results will focus on three aspects: the exam score, the exam execution time and the practice phase execution time. As shown in the following figures, we compared all four models based on those metrics.

The most important comparison to answer our initial research question is the exam score. However, as mentioned before, we had limited resources, therefore, we could only run the experiment once for each model. A better practice would be to repeat the experiments several times to ensure that the result is not an outlier. Nevertheless, as shown in figure 2 the results are interesting. While the quality group performs slightly better or as good as the quantity group for the larger models, the quantity group performs better for the smaller models. However, the difference between the groups is small. One reason for the change of performance might be that the bigger models are making better use of the chat history but this is only speculative. Furthermore, the overall performance is decreasing for the smaller models which is an expected outcome.

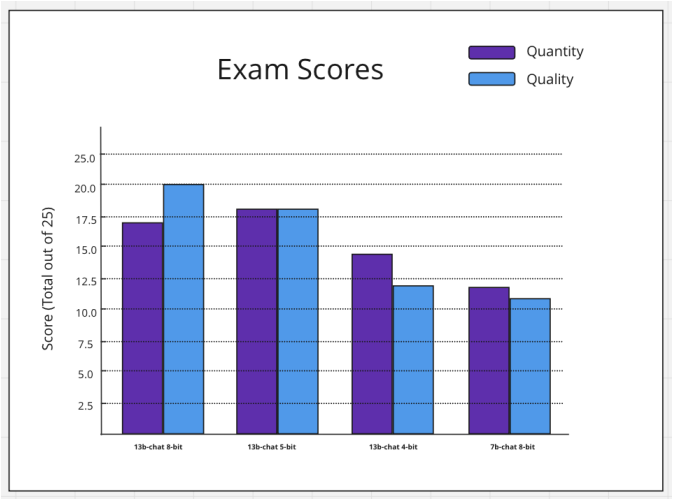

Fig. 2: Comparison of the exam score between the quality and quantity group and the different models.

Figure 4 shows the execution time in seconds for each exam taken in our experiment setup. The exams consist of 25 questions. The figure shows no clear trend for one specific group. However, the quantity exam taken by the 5-bit 13B-chat model took the longest execution time with almost 30 seconds. Nevertheless, due to the fact that the exams are very short, the significance of this comparison is not really high but we still wanted to present it briefly.

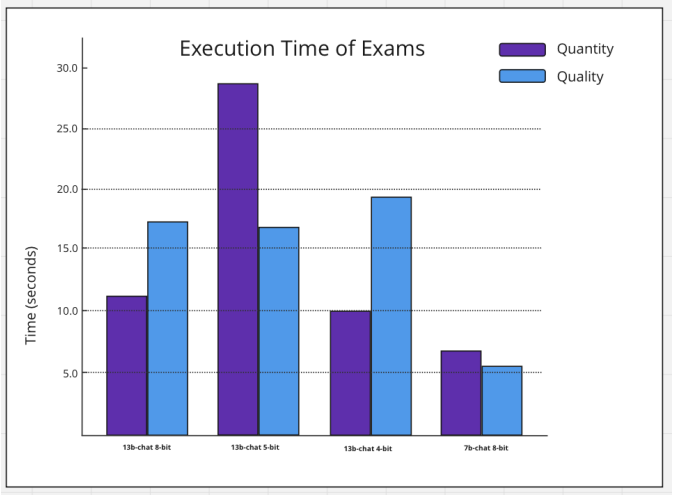

Fig. 3: Comparison of the exam execution time between the quality and quantity group and the different models.

The more significant result is the comparison of the execution time of the learning phases. There we have longer runs from five to 27 minutes. The quantity conversation took about

[12]https://huggingface.co/TheBloke/Llama-2-13B-chat-GGML
[13]https://huggingface.co/TheBloke/Llama-2-7B-Chat-GGML

nine minutes for the biggest model and about six minutes for the smallest model. The quality of the biggest three models are pretty even but the smallest took about five minutes less. The quality execution time is much longer than the quantity execution time since the similarity search takes longer for qualitative discussions and therefore we think it is still a fair competition especially due to the fact that the quantity group gets the answer for all questions once.

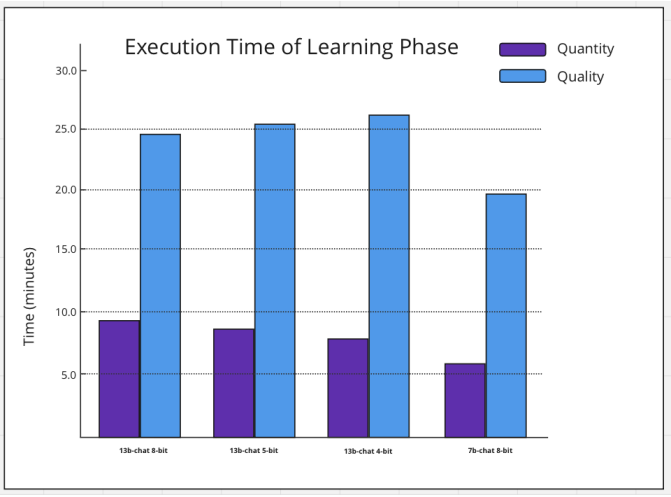

Fig. 4: Comparison of the practice phase execution time between the quality and quantity group and the different models.

Overall no learning method has a clear lead but the bigger models have a higher score on the one hand and the smaller models have a faster execution time on the other hand. Therefore, the results are as expected, whereas we thought that the quality group would get a higher score than the quantity group. Furthermore, we expected a higher average score of the models in general.

VI. Conclusion

In conclusion, our study on the use of generative agents for educational purposes, particularly in American History, reveals that both quantitative and qualitative learning methods facilitated by large language models (LLMs) hold promise. The experiment showed slight differences in performance between these methods across various model sizes, with a trend towards qualitative learning being slightly more effective with larger models. However, the performance difference was not significant, indicating that both approaches can be valuable in educational settings. The study also highlighted the trade-offs between model size and efficiency, underscoring the importance of further research to optimize the use of generative agents in education. Despite the challenges of resource limitations and the need for sophisticated setups, the potential of generative agents to enhance learning experiences is clear. Future work should aim to refine these methods and explore their scalability and applicability in diverse educational contexts.

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.08774

[2] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.09288

[3] J. S. Park, J. C. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein, "Generative agents: Interactive simulacra of human behavior," in *In the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2304.03442

[4] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin *et al.*, "A survey on large language model based autonomous agents," *arXiv preprint*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2308.11432

[5] Z. Xi, W. Chen, X. Guo, W. He, Y. Ding, B. Hong, M. Zhang, J. Wang, S. Jin, E. Zhou *et al.*, "The rise and potential of large language model based agents: A survey," *arXiv*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2309.07864

[6] G. F. N. Mvondo, B. Niu1, and S. Eivazinezhad, "Generative conversational ai and academic integrity: A mixed method investigation to understand the ethical use of llm chatbots in higher education," *SSRN Electronic Journal*, 2023. [Online]. Available: http://dx.doi.org/10.2139/ssrn.4548263

[7] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, "A systematic review of automatic question generation for educational purposes," *International Journal of Artificial Intelligence in Education*, vol. 30, pp. 121–204, 2020. [Online]. Available: https://doi.org/10.1007/s40593-019-00186-y

[8] A. Pinheiro Cavalcanti, A. Barbosa, R. Carvalho, F. Freitas, Y.-S. Tsai, D. Gašević, and R. F. Mello, "Automatic feedback in online learning environments: A systematic literature review," *Computers and Education: Artificial Intelligence*, vol. 2, 2021. [Online]. Available: https://doi.org/10.1016/j.caeai.2021.100027

[9] S. Wollny, J. Schneider, D. Di Mitri, J. Weidlich, M. Rittberger, and H. Drachsler, "Are we there yet? - a systematic literature review on chatbots in education," *Frontiers in Artificial Intelligence*, vol. 4, 2021. [Online]. Available: https://doi.org/10.3389/frai.2021.654924

[10] L. Yan, L. Sha, L. Zhao, Y. Li, R. Martinez-Maldonado, G. Chen, X. Li, Y. Jin, and D. Gašević, "Practical and ethical challenges of large language models in education: A systematic scoping review," *British Journal of Educational Technology*, vol. 55, pp. 90–112, 2023. [Online]. Available: https://doi.org/10.1111/bjet.13370

[11] Large language models and education. Maastricht University. (accessed Feb. 27, 2024). [Online]. Available: https://www.maastrichtuniversity.nl/education/edlab/ai-education-maastricht-university/large-language-models-and-education

[12] J. D. Baierl, "Applications of large language models in education: Literature review and case study," Master's thesis, UCLA, 2023. [Online]. Available: https://escholarship.org/uc/item/6kf0r28s

[13] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020. [Online]. Available: https://doi.org/10.48550/arXiv.2005