Win-win situation: Generative AI in Educational Constrained Environments

Paulo C. R. Pinho[1], Renan Z. da Silva[1], Tiago T. Primo[1], Diego Dermeval[2], Ig I. Bittencourt[2], Seiji Isotani[3].

[1]Universidade Federal de Pelotas (UFPel)

[2]Universidade Federal de Alagoas (Ufal)

[3]Universidade de São Paulo (USP)

pcrpinho@inf.ufpel.edu.br, renan.zafalon@inf.ufpel.edu.br

tiago.primo@inf.ufpel.edu.br, diegodermeval@gmail.com

ig.ibert@ic.ufal.br, sisotani@icmc.usp.br

Abstract. *The global digital divide remains a major barrier to equitable access to technology, and this scenario compromises the pedagogical use of technology and limits learning opportunities for students from disadvantaged communities. Inequality in access to technology is a global issue that directly affects educational and socioeconomic development, especially in countries like Brazil, where there is a significant disparity in access to the Internet and technological devices between different regions and social classes. Despite ongoing efforts, 6% of the public schools still lack broadband, and 3.3% remain without electricity, according to the Ministry of Communications. This gap is most severe in rural and undeserved communities, where limited infrastructure and resources further hinder quality education. Traditional educational technology models are inadequate in such contexts. To address this challenge, we propose Unplugged Artificial Intelligence in Education (AIED), a framework designed to deliver accessible educational technologies without relying on modern infrastructure, stable internet or advanced digital skills. Our goal is to bridge the digital divide and empower students and educators with innovative tools that promote inclusion and reduce educational inequalities. This study presents a model for the deployment of LLMs in offline environments to support literacy and learning in resource-constrained areas. The proposed solution is low-cost and adaptable, allowing schools without internet access to benefit from generative AI. By deploying LLMs locally, the system provides real-time language support and generates educational content without requiring an internet connection. Preliminary results demonstrate the feasibility of running optimized generative models locally, offering practical insights into AI-powered solutions for low-resource environments. By integrating AI agents with curated educational resources, this approach promotes equity and sustainability, ensuring that generative AI serves as a tool for inclusion and innovation in education.*

1. Introduction

The Covid-19 pandemic posed complex challenges across various sectors of society, with education being particularly affected. In the Brazilian context, debates regarding response

DOI: 10.5753/sbie.2025.12497

strategies intensified, as education professionals faced not only the need to contribute to controlling the spread of the virus but also the social repercussions resulting from its circulation. Among the preventive measures adopted, social distancing emerged as one of the most significant, directly impacting the educational process. In this scenario, schools, teachers, and students were abruptly compelled, without prior preparation, to adapt to a new pedagogical reality characterized by an emergency transition to remote and virtual learning.

Inequality in access to technology is a global issue that directly affects educational and socioeconomic development, especially in countries like Brazil, where there is a significant disparity in access to the Internet and technological devices between different regions and social classes. This scenario compromises the pedagogical use of technology and limits learning opportunities for students from disadvantaged communities. At the end of 2022, approximately 3,400 schools in Brazil (2.5%) did not have access to the electrical grid, 9,500 schools (6.8%) lacked internet access, and 46,100 schools (33.2%) did not have computer laboratories. These figures were provided by the National Telecommunications Agency (Anatel) through the School Connectivity Panel, available on the agency's portal [ANATEL 2024].

In this context, it became imperative to explore alternative approaches to teaching and learning in order to mitigate the educational losses resulting from an unexpectedly emerging pandemic. The National Council of Education (CNE) of Brazil reported data that reveal the significant challenges in implementing technology-mediated instruction [CNE 2020]. According to the findings, 83% of public-school students live in socially vulnerable households with a per capita income of up to one minimum wage. While 79% of these students have access to the internet, 46% rely exclusively on mobile phones, and two-thirds do not have a computer.

In Brazil, additionally, the high cost of devices and the concentration of digital content production in the hands of privileged groups intensify digital exclusion. Even with decreasing equipment prices, access remains restricted. In this context, developing innovative solutions becomes essential to democratize access to quality education. Global initiatives such as India's National Academic Digital Library [UNESCO 2023] demonstrate the transformative power of technology in enabling personalized and inclusive learning for students with disabilities. In response, the proposal of a low-cost educational framework based on Large Language Models (LLMs) emerges as a practical and accessible solution that can operate in low-connectivity environments and with limited budgets. This framework not only promotes digital inclusion but also contributes to improving education quality, reducing inequalities, and establishing a replicable model for future initiatives.

2. Theoretical foundation

2.1. Artificial Intelligence in Education (AIED)

Artificial Intelligence in Education (AIED) refers to the use of AI techniques to enhance and personalize the learning experience, adapting to the individual needs of students. Through adaptive systems, AIED offers personalized teaching, real-time feedback, and support for lesson planning and student monitoring. It enables data-driven pedagogical decisions, making the learning process more dynamic, student-centered, and accessible, especially in contexts with limited infrastructure [Wang et al. 2024]. AIED has transformative potential in areas such as content generation, feedback, and student engagement [Wang et al. 2024]. However, many of its benefits rely on cloud-based infrastructure, which limits accessibility in low-resource environments where connectivity and advanced hardware are scarce [Rodríguez and Cobo 2022]. To address this challenge, offline approaches such as AIED Unplugged [Isotani et al. 2023] have emerged. Introduced by [Isotani et al. 2023], the concept refers to the implementation of AI solutions in educational contexts with intermittent or no internet access, and where users possess minimal digital skills. AIED Unplugged promotes the use of lightweight AI on low-cost edge devices, enabling personalized and collaborative learning experiences in underserved regions. These solutions must be simple, functional offline, and accessible even to those with limited resources, ensuring educational equity and fostering social and economic development. Despite the promise of offline AIED, the feasibility of using LLMs in such contexts remains constrained due to their high computational demands [Yin et al. 2024]. Recent advances in model quantization and distillation have opened new pathways for deploying LLMs on mobile and low-power devices.

2.2. Large Language Models (LLMs)

LLMs are AI systems designed to understand and generate text on a massive scale. These models have been driven by significant advances in neural network architecture. In particular, the Transformer, proposed by [Vaswani et al. 2017], is a foundational architecture behind this progress. LLMs are trained on large textual datasets using supervised learning and transfer pre-training techniques. This approach, described by [Radford et al. 2018], enables the models to learn complex linguistic patterns and contextual relationships. Once trained, LLMs demonstrate the ability to generate coherent and relevant text in response to specific inputs, opening the door to a wide range of applications such as machine translation, assisted text generation, and other language-related tasks. In recent years, LLMs have made significant advances in the field of Natural Language Processing (NLP), achieving human-like text generation, question answering, and performing various other language tasks with high accuracy. The main applications of LLMs span a wide range of areas, reflecting the transformative impact of these technologies on how humans interact with computational systems. With their ability to understand and generate natural language, LLMs are employed in various tasks that require advanced linguistic intelligence. One of the most common applications is **text generation**, where LLMs are used to automatically produce summaries, articles, essays, and even fictional dialogues. They also play an important role in **machine translation**, providing more accurate and fluent translations between different languages. In addition, **virtual assistants**, such as chatbots and question-answering systems, make extensive use of LLMs to interpret commands and generate coherent responses in real time. Another relevant application is **sentiment analysis**, which enables the identification of whether a text expresses positive, negative, or neutral emotions—particularly useful in contexts such as social media analysis and opinion research. LLMs are also widely used in **automatic correction** systems, present in keyboards and text editors, where they provide word suggestions and correct grammatical and spelling errors.

Finally, LLMs are highly efficient in **text processing and generation**. During training, they learn to predict the next word in a text sequence. This ability is applied in text generation, allowing the model to continue provided sequences or initiate new texts based on a prompt. These examples demonstrate the versatility of LLMs and their potential as powerful tools for handling natural language processing tasks.

2.3. Open-source models

Open-source models are AI systems whose source code, training parameters, and sometimes the data used for their development are made publicly available. This allows anyone to access, study, modify, and distribute the model freely. The open-source philosophy promotes collaboration, transparency, and innovation, enabling developers and researchers worldwide to contribute to technological advancement. Key characteristics of open-source models include transparency, as all the code and, in some cases, the training process are public, allowing for a detailed understanding of how the model was built and how it functions. Accessibility is also crucial, since anyone can download and use the model for academic, commercial, or personal purposes, lowering barriers to AI adoption. Customization is another important aspect, enabling users to adapt models to their specific needs by adjusting parameters or retraining with new data. Additionally, open-source projects often have active and collaborative communities that contribute improvements, bug fixes, and new features, accelerating development and innovation. Finally, many open-source models are free of charge, allowing organizations with limited resources to access advanced technologies without additional costs.

2.4. Tools and Devices

1) **LangChain:** Open-source framework designed to streamline the development of applications using LLMs, such as GPT-3 and GPT-4. Its core innovation lies in the concept of **chains**, modular workflows that integrate multiple AI functions, including natural language understanding, information retrieval, and text generation. This modular structure allows developers to flexibly combine and reuse components like databases, APIs, and LLMs, enabling more dynamic and context-aware applications. Available for both Python and JavaScript, LangChain supports a broad range of platforms and use cases, from chatbots and recommendation systems to code analysis and document summarization. Its architecture makes it easy to build, modify, and scale AI workflows, enhancing both collaboration and development speed. LangChain also simplifies model tuning and performance optimization within a centralized environment, making it a popular choice for developers and data scientists seeking efficient and adaptable AI solutions.

2) **LangFlow**: No-code tool designed to simplify the creation and management of workflows in LangChain. The platform enables the development of AI agents and complex processes without the need for programming, through an intuitive visual interface that allows users to create and manage workflows using drag-and-drop functionality. Additionally, LangFlow provides tools for parameter editing and offers a marketplace with pre-built models and examples. Its main advantages include accessibility, ease of use, and the fact that it is free and open-source, promoting community collaboration. The tool is particularly suitable for developing customer support agents, automating internal processes, and customizing content generation. Its core features include visual workflow design, parameter editing, and access to the model marketplace.

3) **Single-board Computers (SBCs)**: To promote a more inclusive and accessible educational environment, especially in contexts with limited resources, it is essential to

explore the use of low-cost devices. These devices offer practical and affordable solutions capable of meeting a variety of educational needs, from practicing digital skills to supporting creative projects and scientific experiments. Among the most effective tools for educational environments with tight budgets are SBCs. These modules integrate a processor, memory, input and output interfaces, and other essential components into a single board, providing a complete hardware and software solution. A widely used example is the Raspberry Pi, a compact and affordable device that plays a key role in educational applications. It enables the creation of low-cost computer labs where students can practice programming, logic, and other essential digital skills. Additionally, the Raspberry Pi can function as a local server to store and distribute educational materials such as books, videos, and exercises without relying on internet connectivity. Its versatility also makes it ideal for robotics and automation projects, offering a hands-on platform to learn about sensors, actuators, and robot programming. Furthermore, it is often used to create educational simulators for subjects like physics experiments or chemical process simulations, enriching the learning experience through accessible practical applications.

3. Methodological Procedures

The methodology was developed with the aim of creating an integrated solution for schools and communities with limited infrastructure or no internet access, and is divided into the following stages: (1) Creation of educational tools that operate effectively in environments with limited infrastructure, using devices such as the Raspberry Pi with optimized operating systems. The software will be designed to work offline, with local storage, and will feature interactive learning platforms and assessment systems, all tested to ensure robust performance, (2) Integration of artificial intelligence (AI) solutions will be applied to improve literacy and digital inclusion. Retrieval and Generation (RAG) techniques will be used, in addition to the adjustment of language models with specific educational data. Specialized AI agents, focusing on aspects such as grammar and clarity, will work collaboratively to provide accurate corrections and feedback, (3) Evaluation and impact measurement will involve the collection of quantitative data, such as essay scores and correction time, and qualitative data, such as feedback from students and teachers. Comparisons between automatic and manual corrections will be made to measure the accuracy and quality of feedback, allowing adjustments to the algorithms based on the results obtained and user feedback. Figure 1 presents the model application scenario:

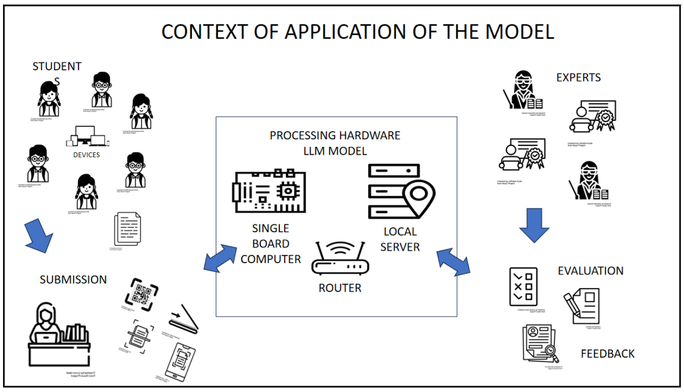

Figura 1. Model Application Scenario

The use cases for the framework solution cover two main contexts. In the first scenario, the technology will be applied offline on devices such as Raspberry Pi, cell phones and tablets, without the need for an internet connection, with the aim of evaluating the impact on students' writing skills and the adequacy of the feedback provided. In the second scenario, in environments with robust local infrastructure, the system will allow automatic corrections in real time, with the evaluation focused on comparing the effectiveness of automatic and manual corrections, adjusting the model based on user feedback. Both scenarios aim to ensure that the solution is effective in different contexts, providing quality educational support.

4. Framework AInclude

AInclude is a low-cost, LLM-based educational solution designed to support literacy and learning in areas with limited infrastructure, without relying on constant internet access. It includes the configuration of SBCs (such as Raspberry Pi and Orange Pi), local servers and APIs, ensuring accessibility in regions with limited connectivity, as well as the adaptation of open source models to meet educational needs, the Figure 2, details the framework:

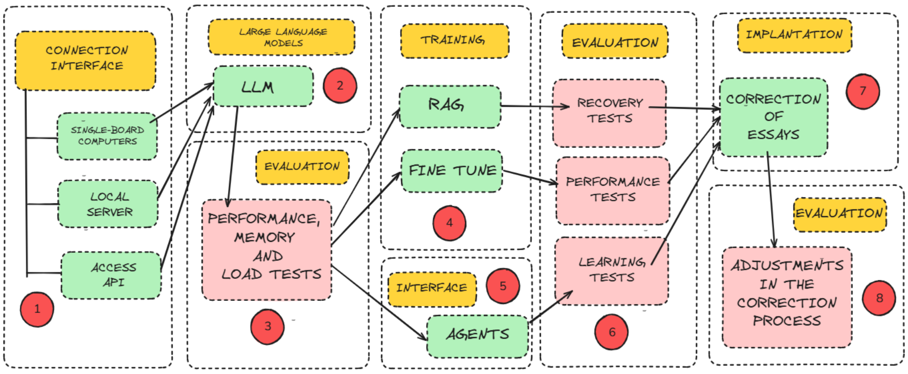

Figura 2. Framework Flow

The process is divided into 8 steps: (1) **Connection Interface**, which configures SBCs and servers to run LLMs with API access, allowing online and offline operations; (2) **LLMs**, where models such as TinyLLama and Mistral are adjusted for the limited hardware; (3) **Connection Interface and LLMs Assessment**, which tests the stability and performance of SBCs, servers and APIs; (4) **Training and Refinement**, using RAG and fine-tuning techniques to adapt the models to the educational context; (5) **Interfaces: AI Agents**, which uses specialized agents for text review, adopting a multi-agent approach; (6) **Training and Agents Evaluation**, to verify the accuracy of the responses and the performance of the agents; (7) **Implementation: Essay Correction**, where the models provide automated feedback on students' essays; and (8) **Implementation Evaluation**, which monitors and adjusts the correction process to ensure that the system is aligned with educational objectives. This integrated cycle aims to ensure access to artificial intelligence in an accessible and efficient way, especially in challenging educational contexts.

AInclude aims to promote digital inclusion and improve the quality of education in rural and remote areas, eliminating the need for continuous internet connection. The tool will be easy to use and accessible, designed to work offline, with online synchronization options, and incorporate natural language processing and computer vision technologies. The tool will perform a preliminary analysis of the essays, as a case study, based on standardized criteria, providing structured and detailed feedback. The evaluation process begins with the submission of the essays by the students, who generate preliminary automatic feedback, which is then reviewed by experts through a double-blind evaluation, ensuring impartiality and validating the automatic corrections. After the initial feedback, students will have the opportunity to revise their essays and resubmit them for a final evaluation by the experts. This cycle of feedback and review aims to improve the quality of the texts and confirm the effectiveness of the AI tool.

5. Preliminary Results (Testing Laboratory)

5.1. Phase 1: Results and Conclusions

Phase 1 of the Test Lab was essential to validate the feasibility of implementing LLMs in environments with limited infrastructure, as in scenario 1. During this phase, three main experiments were carried out, ranging from the initial hardware configuration to the installation and execution of local AI models.

LAB 1.0: Initial Configuration Process introduced the preparation of the Raspberry Pi 3 Model B, including writing the operating system to the 64 GB microSD card, configuring the network, and enabling SSH for remote access. The Raspberry Pi was connected to the power supply, and all configurations were successfully applied, allowing remote communication without the need for a physical monitor, keyboard, or mouse.

LAB 1.1: Installing a Local and Open Source LLM sought to install a lightweight model, considering the limitations of the Raspberry Pi 3. Initially, the TinyLlama model was tested with Llama.cpp, but the lack of RAM memory made it impossible to execute. Changes such as removing the operating system's graphical interface were also not enough. However, installing the Ollama platform allowed TinyLlama to be downloaded and successfully executed via the command prompt, although the 40-minute response time made the solution impractical for educational use. **LAB 1.2: Local Chatbot using Frameworks**, the focus was on testing the feasibility of a functional chatbot in the local environment. Ollama was installed on a compatible system and configured to run TinyLlama. The model was successfully loaded, allowing interactions via the command prompt without the need to connect to external servers, which confirmed the possibility of local use for testing and initial development.

Phase 1 of the Test Lab demonstrated that, despite hardware limitations, it is possible to run LLM models locally, albeit with performance constraints. Using the Raspberry Pi 3 was challenging due to its limited processing and memory capacity, but it pointed to future optimizations, such as using more powerful devices, such as the Raspberry Pi 5. Evaluation of the model on a laptop with an Intel Core i5-1135G7 processor and 8 GB of RAM showed a response time of 32 seconds, considered good for this configuration, but with room for improvement, especially compared to commercial models. The analysis indicates that system adjustments, model optimization, and processing improvements can result in faster response times and more efficient performance in offline environments.

5.2. Phase 2: Results and Conclusions

Phase 2 of the Test Lab evaluated the integration and operation of LLMs in an offline environment using LangFlow. This phase included installing the tool, configuring the workflow, and testing basic prompts to verify its viability in educational applications.

LAB 2.0: LangFlow Installation and Configuration, different approaches were explored to install the tool, which allows the construction of chatbots, agents and RAG flows in a visual way. The installation was performed through the pip package manager, simplifying the configuration process in VS Code. After installation, the execution was done with the command 'python -m langflow run, starting a local instance accessible via browser. The procedure demonstrated the practicality and flexibility of the tool for experiments with local AI models.

LAB 2.1: LangFlow Project, the development of an interactive flow for AI using the LangFlow visual interface was tested. The process involved creating a new project, defining the flow by adding modular blocks, configuring each block individually, and establishing logical connections between them. After adjustments and testing, the flow was validated in the Playground, allowing its functionality and adaptability for future educational implementations to be assessed.

LAB 2.2: Basic Prompt using LangFlow, a chat structure based on the TinyLlama model was built using Ollama. The flow was configured with the main blocks: 'Chat Input', which receives user input; 'LLM', responsible for processing with adjustable parameters, such as temperature and verbose; and 'Chat Output', where the response is displayed. The Playground allowed direct interactions, facilitating the analysis of the model's performance in a controlled environment and without the need for an internet connection.

Phase 2 demonstrated that LangFlow is an effective tool for structuring local AI workflows, facilitating the configuration and customization of models such as TinyLlama. The tests reinforced the feasibility of offline use for educational support, highlighting areas for improvement, such as performance optimization and workflow adjustments for greater usability on low-cost devices. With a focus on local functionality and interaction with the LLM, the LangFlow platform demonstrated itself to be efficient for experimentation and use of models in an offline environment. Its visual interface and modular structure provided a satisfactory evaluation, ensuring fluidity in interaction and immediate results. The tests confirmed that LangFlow offers a robust and efficient environment for the development and application of LLMs, meeting the needs of scenario 1 and highlighting its potential as a tool to support research.

5.3. Considerations

Several important challenges and considerations have emerged for implementing AI technologies in educational environments with low-cost devices, highlighting the need to adapt solutions to specific usage conditions. The experiments in LAB 1.0 and LAB 1.1, with the Raspberry Pi 3, highlighted advances in the use of affordable hardware, but also revealed significant limitations in processing and memory, especially when running the TinyLlama model. This highlighted the need for optimization of the operating system and the consideration of more robust devices, such as the Raspberry Pi 4 or 5. In LAB 1.1 and LAB 1.2, the implementation of TinyLlama via Ollama was successful, but the 40-minute response time proved unfeasible for educational use. With the migration to a notebook with greater capacity, response times decreased, but there is still room for improvement. These results reinforce the importance of balancing performance and feasibility, taking into account the limitations of the available hardware. In LAB 2.0, the implementation of LangFlow has proven to be effective in structuring personalized AI workflows, although the learning curve and the need for technical skills may make it difficult for educators without prior experience. In addition, it is necessary to test its feasibility on low-cost devices, considering local processing and the possibility of an offline environment with a dedicated server. Continued testing in the Labs will be essential to optimize the performance of the models in offline scenarios and improve the integration of the tools, ensuring effective and accessible solutions for environments with limited infrastructure.

6. Future Steps

Based on the investigations carried out, several future directions emerge for enhancing AI solutions in the educational context. Continued testing and exploration of new possibilities are crucial to advancing the development of more effective and adaptable technologies.

1) **Advances in Hardware and Software**: The need for devices with better processing power and memory to support advanced AI models becomes evident. Additional testing should focus on optimizing low-cost hardware and adapting operating systems to ensure that technological solutions can operate efficiently in resource-limited environments.

2) **Improving the Efficiency of LLMs**: The effectiveness of an LLM implementation should be improved through fine-tuning and the integration of techniques such as RAG. The goal is to increase the accuracy and relevance of the feedback generated by open-source models using the essay database available on the UOL[1] website, reducing response times and increasing the usefulness of essay-correction solutions.

3) **Development of Specialized and Multi-Agent Systems**: The creation and improvement of specialized agents, such as those focused on spelling, grammar, and coherence, have the potential to transform the essay correction process. Research should explore how collaboration between different agents can lead to a more detailed and efficient correction system, as well as evaluate the benefits and feasibility of multi-agent systems.

4) **Implementation and Scalability**: Exploring deployment possibilities for LangFlow projects and the use of APIs is essential to ensure the scalability and accessibility of the developed solutions. Future studies should consider how these technologies can be implemented on a large scale and integrated into various educational contexts, ensuring that AI solutions are adaptable and sustainable.

These future directions provide a clear path for the development and application of AI technologies in the educational field, with a focus on the effectiveness, accessibility, and scalability of the proposed solutions.

[1]https://educacao.uol.com.br/bancoderedacoes/

Referências

ANATEL (2024). Painel conectividade nas escolas. Disponível em: `https://informacoes.anatel.gov.br/paineis/infraestrutura/conectividade-nas-escolas`. Acesso em: 20 jun. 2024.

CNE (2020). Parecer cne/cp nº 11/2020, orientações educacionais para a realização de aulas e atividades pedagógicas presenciais e não presenciais no contexto da pandemia. Disponível em: `https://www.gov.br/mec/pt-br/cne/parecer-cp-2020`. Acesso em: 04 ago. 2025.

Isotani, S., Bittencourt, I. I., Challco, G. C., Dermeval, D., and Mello, R. F. (2023). Aied unplugged: Leapfrogging the digital divide to reach the underserved. In *International Conference on Artificial Intelligence in Education*, pages 772–779. Springer.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training. *OpenAI*.

Rodríguez, M. R. B. and Cobo, C. (2022). Covid-19 and education in the global south: Emergency remote learning solutions with long-term implications.

UNESCO (2023). Global education monitoring report summary, 2023: technology in education: a tool on whose terms? Disponível em: `https://unesdoc.unesco.org/ark:/48223/pf0000386147`. Acesso em: 25 ago. 2025.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, B., Liu, J., Karimnazarov, J., and Thompson, N. (2024). Task supportive and personalized human-large language model interaction: A user study. In *Proceedings of the 2024 Conference on Human Information Interaction and Retrieval*, pages 370–375.

Yin, W., Xu, M., Li, Y., and Liu, X. (2024). Llm as a system service on mobile devices. *arXiv preprint arXiv:2403.11805*.