



AdaptiveGPT: Towards Intelligent Adaptive Learning

Andréia dos Santos Sachete^{1,2} · Alba Valéria de Sant'anna de Freitas Loiola² · Raquel Salcedo Gomes²

Received: 1 October 2023 / Revised: 22 May 2024 / Accepted: 18 August 2024

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

Adaptive learning is an educational methodology that allows the personalization of learning according to the student's pedagogical path. In digital environments, the strategic use of technologies enhances adaptive learning initiatives, enabling a dynamic understanding of intricate contextual nuances and the ability to identify and recommend appropriate learning activities. Therefore, this work proposes developing and evaluating a prototype that uses a large language model to create adaptive educational activities in face-to-face and virtual environments automatically. The applied methodology involves the implementation of a large language model with advanced cognitive capabilities to generate learning activities that adapt to individual needs. A proof of concept was developed to evaluate the practicality and usability of this approach. The research results indicate that the approach is practical and adaptable to different educational contexts, reinforcing the synergy between adaptive learning, artificial intelligence, and learning environments. The proof of concept evaluation showed that the prototype is highly usable, validating the proposal as an innovative solution to the growing needs of modern education.

Keywords Adaptive learning · Application · Foundational model · Large language model

Alba Valéria de Sant'anna de Freitas Loiola and Raquel Salcedo Gomes are contributed equally to this work.

✉ Andréia dos Santos Sachete
andrea.sachete@ifarroupilha.edu.br

Alba Valéria de Sant'anna de Freitas Loiola
alba.portugues@gmail.com

Raquel Salcedo Gomes
raquel.salcedo@ufrgs.br

¹ Federal Institute Farroupilha, RS 377 - KM 27, Alegrete, RS, Brazil

² PGIE, Federal University of Rio Grande do Sul, Av. Paulo Gama, 110, Porto Alegre, RS, Brazil

1 Introduction

The acquisition of knowledge by students consists of an individualized learning curve [1], in which each has their pattern of progress, which generally differs from their classmates. However, classes are often taught homogeneously without considering the particularities of each learner.

The proposal of adaptive learning precisely aims to address these differences, as it involves the customization of teaching and learning processes and activities to align with each learner's developmental trajectory [2]. Information and Communication Technologies (ICTs) have been enabling the enhancement of this strategy, as computerbased environments allow the adaptation of activities and content via programming languages. Adaptive learning computer environments can store students' choices and compare them using conditionals, thus tailoring a pedagogical path more suitable for each learner's learning curve, adapting the journey according to their needs [3].

This means that adaptive learning environments that use digital technologies but do not use artificial intelligence suffer from two computational problems. The first one concerns spatial limitations, as such environments must initially maintain a large database with all activity options at different levels of difficulty so that the algorithm can navigate through these options. The second one concerns temporal limitations, as the larger the databases are, the slower the search process for new options by learning adaptation algorithms can be.

Although ICTs have aided in the dissemination of adaptive learning, and even with the advent of generative artificial intelligence [4], as well as large language models [5] where text generation capabilities are virtually unlimited, there are still few research studies that explore such technologies for this purpose. In terms of adaptive learning with the assistance of artificial intelligence, we can find papers: (i) review papers that aim to identify learning patterns [6]; (ii) case studies on the application of artificial intelligence in this context [7]; (iii) papers that assess the potential of existing applications and enable adaptive learning models [8, 9], and (iv) papers discussing the application of adaptive learning in foreign language teaching, analyzing which neural network models could be better utilized for this purpose [10, 11]. However, these studies do not provide applications or empirical evaluation implementations. One of the few proposals that implement artificial intelligence to learning does so in a gamified tutoring environment. It is the work by [12], which uses its neural network-integrated into a game.

Unlike the previous proposals, this work utilizes a generative artificial intelligence [4] that does not need to be implemented by the application developer but instead uses a pre-trained foundation model (GPT-3.5) for adaptive learning development. The idea behind generative artificial intelligence models is precisely the possibility of creating new and unique content, striving to simulate human creativity to the fullest extent possible. Therefore, adaptive learning can benefit from these models in producing new pedagogical approaches that can adapt to the individual trajectory of each student.

In this context, this work proposes developing and evaluating a prototype that integrates artificial intelligence and adaptive learning in the form of a reflective (dis- cursive) application. Initially, it comprehends the classroom context and subsequently generates the activities offered to the students dynamically and at varying difficulty levels. Furthermore, it shall be able to understand and evaluate the written responses to each proposed activity and adjust levels and quantities of activity in real-time based on the quality of the answers provided by the students. In this way, the pro- posed application should, based on

a pre-understood context, generate questions and evaluate responses in such a manner as to guide the adaptation of different levels of difficulty for the questions. Finally, the application should present to the teacher the individualized trajectory followed by the student.

To achieve this, we have developed a proof of concept that utilizes the application programming interface (API) of OpenAI [13]—the company behind ChatGPT. Teachers from various fields evaluated the usability of this approach, and the results demonstrated its potential for use in both traditional classrooms and virtual learning environments.

Therefore, this article presents three main contributions:

- The first contribution is a discussion on the resumption of adaptive learning supported by large language models.
- The second contribution is the development of a prototype (proof of concept) using languages and models accessible to teachers at the most diverse levels and education systems.
- The third contribution is an in-depth evaluation of the prototype's usability, accompanied by a discussion on its limitations and future work. This provides a valuable opportunity for developers of educational applications to implement and enhance the tool in their own contexts, as it is freely available.

Thus, the paper is organized according the following structure: in Section 2, fundamental concepts about adaptive learning are presented and described; in Section 3, the architectures of generative artificial intelligence and large language models are discussed; in Section 4, the proposal of this work is presented; in Section 5, the proof of concept implemented and evaluated in this work is proposed, and its evaluations are discussed in Section 6; the final remarks are presented in Section 7, and suggestions for future work are discussed in Section 8.

2 Adaptive learning

Adaptive learning has been explored by various researchers in different contexts, both in traditional instruction and in assisted instruction [14], as it is a teaching method that utilizes technology to personalize the learning process according to the individual needs of the student. Unlike traditional approaches that follow a fixed curriculum and uniform pace for all students, adaptive learning aims to adapt the curriculum to each individual's abilities, interests, preferences, and needs, enabling a personalized and efficient educational experience [15].

Adaptive learning has emerged as one of the most promising pedagogical approaches in the contemporary technological educational landscape. With roots in customization and the ability to meet individual student needs, this methodology reflects a shift towards more student-centered teaching, promoting optimized learning processes [16]. However, adaptive learning is not a new idea. Since the early days of education, teachers intuitively adjusted their approaches based on student responses and needs. What has changed in the present era is the scale and precision with which this adaptation can occur, thanks to technological advances.

At its core, adaptive learning refers to the ability to modify the educational experience in real time to suit a student's individual needs and skills. This adaptation can vary in terms of content, resources, activities, and assessments. Technology plays a crucial role in facilitating adaptive learning on a large scale [17]. Modern online learning platforms use sophisticated algorithms to analyze students' real-time performance. These systems can identify patterns, such as topics that students find more challenging, and adapt the content to address these areas of difficulty. These platforms provide immediate feedback, allowing students to quickly understand and correct their mistakes. Additionally, adjusting the pace of instruction to the student's level ensures that each student is constantly challenged, but not overwhelmed.

The importance of adaptive learning lies in the fact that each learner possesses unique learning styles, rates of knowledge assimilation, rhythm and different levels of prior knowledge. Therefore, with this approach, the learner can progress at their own pace, advancing more quickly in areas where they have more excellent proficiency and receiving additional support in areas where they need more assistance [18]. Some essential features are worth highlighting:

- **Customization:** In large classes, it is challenging for teachers to meet individual needs. Adaptive learning technology overcomes this barrier, allowing for more individual-focused education.
- **Efficiency:** Learning time is optimized by focusing on areas where students are struggling and advancing in areas where they are already proficient.
- **Engagement:** Students feel more engaged when the learning material is relevant and appropriate to their level.
- **Item Immediate Feedback:** Students can understand their areas of improvement before a formal assessment.

Respecting the student's learning trajectory and curve can promote motivation, engagement, and self-confidence. In this way, they are more likely to feel involved in the learning process, as they are challenged according to their skill level and receive immediate feedback on their performance. This approach helps prevent students from feeling discouraged, lost, or bored with a generic and inflexible curriculum [19].

Despite its many benefits, adaptive learning also faces challenges. Resistance to change from institutions and educators, the need for proper professional training, and concerns about student data privacy are obstacles that need to be overcome. Additionally, technology must not replace the human element in education. Instead, it should be a tool that complements and amplifies the educator's skills. With the growing integration of technology in education, we are likely to see an even greater adoption of adaptive learning. As Artificial Intelligence (AI) and Machine Learning continue to advance, adaptive systems will become even more accurate and effective in meeting the needs of students [20, 21].

For educational institutions, the key will be to balance technology integration with effective pedagogical practices, ensuring that students absorb information and develop critical thinking and problem-solving skills [22] around them and beyond. Adaptive learning represents a revolution in how we approach education. We can make learning more effective, engaging, and meaningful by placing the student at the center of the learning process and adapting it to their individual needs. As we navigate the ever-evolving frontier of educational technology, we must focus on the ultimate goal: creating learning experiences that enable all students to reach their full potential.

3 Large language models

In the field of information technology, some initiatives employ adaptive learning [23]. Below are some examples:

- **Intelligent Tutoring Systems:** These systems are designed to provide personalized instructions. By tracking the learner's progress, they identify areas of difficulty and offer additional resources and exercises to aid in understanding the content.
- **Adaptive E-Learning Platforms:** These platforms employ machine learning algorithms to analyze the learner's performance and adapt the content based on identified knowledge gaps. This way, they can focus on the areas that need more attention.
- **Adaptive Educational Gamification:** Educational games can be designed with adaptive features, adjusting the difficulty level and challenges according to the student's progress. The goal is to keep them motivated and engaged while offering personalized learning opportunities.

These initiatives in information technology are transforming education, enabling students to learn more efficiently and effectively while respecting their learning trajectories and maximizing each individual's potential.

Large Language Models (LLMs) [5] are machine learning models that perform various tasks in natural language processing, such as text generation, text classification, conversational agents, and translation. LLMs are trained with a large number of parameters to match the expected behavior of the model. In the machine learning domain, it is common to train generic models referred to as Foundation Models, which can subsequently serve as the foundation for specific applications. These base models are trained on large datasets for a general task, such as natural language processing (NLP) or computer vision, to learn rich and general data representations.

Foundation models like GPT-3 (Generative Pre-trained Transformer 3) [24] from OpenAI are trained on many unlabeled data, such as internet texts or large image databases, using unsupervised learning techniques. These models learn to capture patterns, structures, and relationships in the data, enabling them to understand and generate relevant information. The foundation of the model is a Transformer architecture [25], a neural architecture designed to process sequential data, like text. This architecture employs self-attention layers to capture relationships between words in the text, enabling the model to capture linguistic structures and patterns, allowing for a more comprehensive and context-aware understanding during text generation.

LLMs initially learn unsupervised by masking tokens (sets of characters or words) in a sentence and calculating the probability of tokens that can fit into that space while maintaining the same context, thereby predicting the next token in a sentence. This process is repeated multiple times until the LLM reaches an acceptable confidence level in prediction. Once the base models have been pre-trained, they can be adapted to specific tasks through fine-tuning. In this process, the models are trained on smaller, labeled datasets specific to a particular task. By adjusting the weights and parameters of the base model, it is possible to tailor it to perform more specific tasks, such as text classification, object recognition, or automatic translation.

Base models are an efficient approach to machine learning solutions development because training a base model typically requires significant computational resources and large datasets. By utilizing a pre-trained base model, developers can save time and

resources by leveraging the base model's prior knowledge and focusing on the specific task. Additionally, base models are often designed modularly, allowing for reuse in different contexts and tasks.

Several current works are using LLMs to support adaptive learning through AI techniques. Liu et al. [26] explore the adaptive clustering-based learning approach to detect and categorize visual relationships in images, focusing on identifying and classifying interactions between pairs of objects using a network that divides the relationship space into distinct subspaces. Sudhakaran et al. [27] present a new procedural content generation (PCG) approach using tuned GPT-2 models. The central theme revolves around addressing the limitations of traditional PCG algorithms by integrating large language models (LLMs) to increase diversity and control over generated content. The work of Wen et al. [28] aims to incorporate the diversification of categories into recommendation systems without compromising the accuracy of recommendations. The methodology involves a two-pronged approach: first, developing a Diversity-Aware Causal User Model (DCUM) that estimates user satisfaction by unraveling the causal effects of users' inherent preferences and the monotony effect. Second, implementing a ChatGPT-aided state encoder (CSE) provides nuanced representations of user states, incorporating information from multiple categories that align with potential user preferences. Shu et al. [29] present a large instruction-tuned language model to improve text rewriting, especially between sentences. This model introduces new data generation and training strategies to enhance the ability of large language models (LLMs) to perform rewriting tasks according to specific guidelines.

The tool proposed in this work, AdaptiveGPT, can complement all these works mentioned, incorporating their capabilities with new models.

4 LLM adaptive learning proposal

OpenAI provides an API allowing user-developed applications to communicate with fine-tuned models tailored to specific domains, such as natural language processing, image generation and editing, audio-to-text conversion, and translation. This work uses an API to interact with the GPT-3.5 model, which is optimized for conversational applications in the form of chatbots, capable of understanding and generating natural language code.

For this work, GPT-3.5 allows the submission of prompts [30], which are suitable commands to guide the responses generated by the model. We chose OpenAI's GPT-3.5 model due to the speed of response times and also the quality of responses to prompts [31, 32]. We prefer to use an already trained foundational model rather than fine-tuning the model. This is because the proposal of this work focuses on easy access and easy use by teachers. Therefore, using the model already trained by OpenAI is the best option, as everyone who uses the tool will have access to the same model and will not need to worry about AI processes and knowledge in training neural networks [33]. The OpenAI API returns the answers to the submitted queries (prompts) in the format of a JSON (JavaScript Object Notation) file [34], which is an open standard attribute-value format for data exchange between applications.

A proposed implementation can be observed in the sequence diagram in Fig. 1. A sequence diagram is a Unified Modeling Language (UML) interaction diagram that describes how and in what order a group of system objects work together [35]. In this scenario, a teacher can use a prompt to provide context to the pre-trained generative transformer model (Generative Pre-trained Transformer—GPT), such as the OpenAI

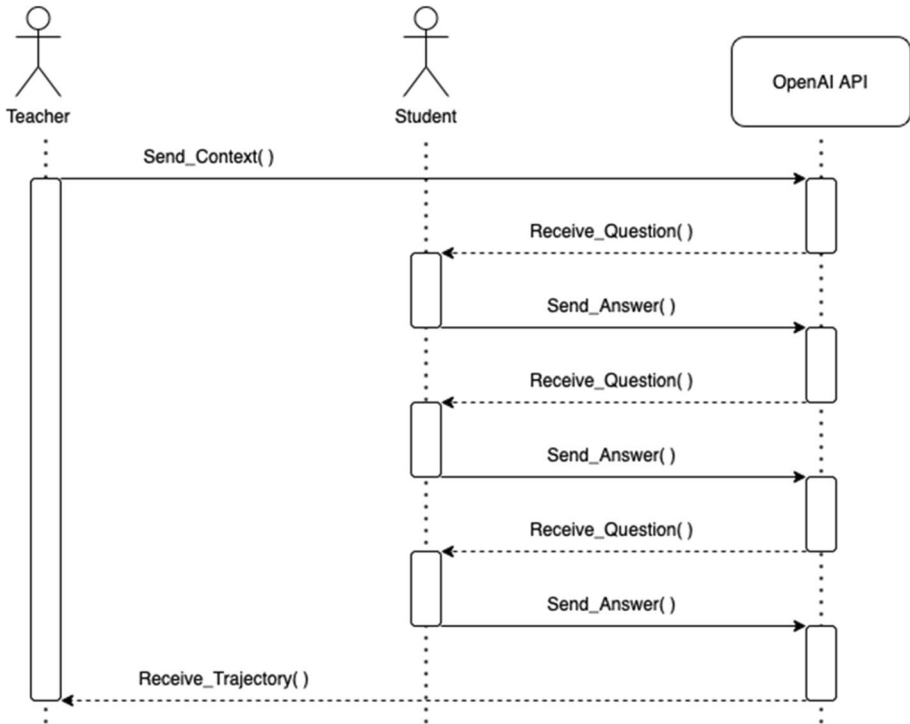


Fig. 1 Sequence diagram of the interaction among teacher, student, and OpenAI API

API. Based on this scenario, the teacher can guide the model in creating adaptive activities for the student using new prompts. At the end of the student’s interaction with the activities generated by GPT, the model returns to the teacher the entire pedagogical path (trajectory) followed by the student.

For adaptive learning to be effective, it is necessary for the prompts in the developed application to guide the difficulty level of the subsequent activity based on some criteria. GPT offers a range of criteria that can be analyzed in the written text, so we chose reflective activities (rather than objective ones). By discussing the questions textually, it is possible to use GPT to enable assessment beyond right and wrong answers.

If we consider only right and wrong answers as the determining factor for adapting activities to students and taking into account that there are questions with different levels of difficulty (easy, medium, and hard), we could have an application that, after providing context to GPT, begins the activity by generating an easy question. Each time the student answers correctly, GPT would increase the difficulty level until the student can answer a difficult-level question. This is a simplified version among all the text analysis capabilities of GPT and is the chosen approach in this work for a proof of concept.

However, GPT enables more sophisticated discourse analyses, such as describing whether the perception of humor in the response is positive, negative, or neutral and even identifying the sentiment conveyed in the text. This allows for a vast array of conditions for adaptive learning, such as associating question correctness with emotions and moods or even the time taken for a response to be submitted. All of these new nuances can be guided through the prompts submitted by the application.

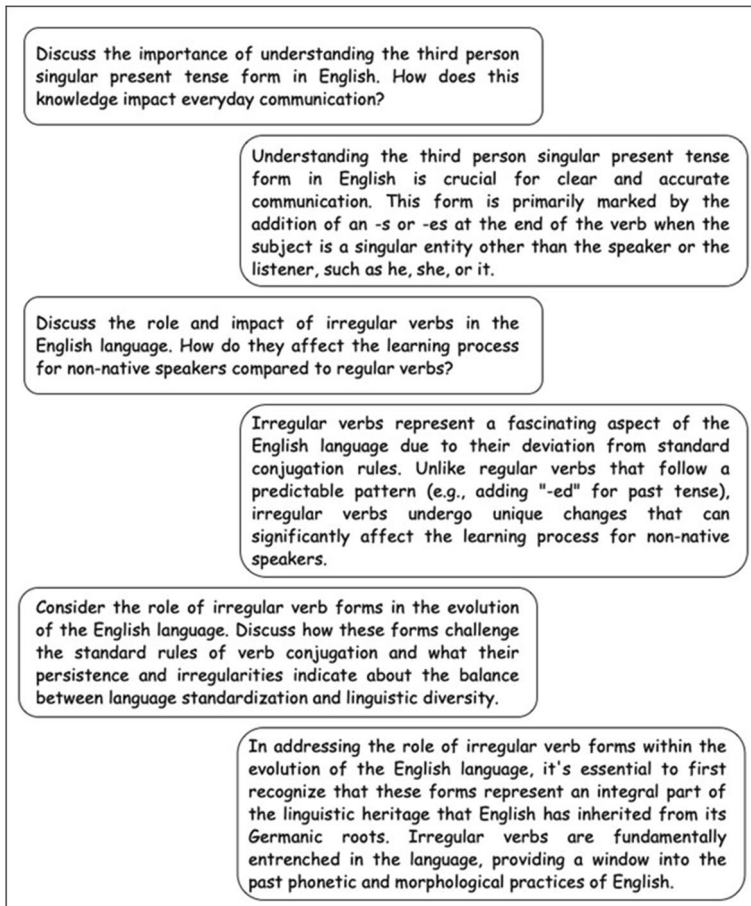


Fig. 2 Interaction between Student and AI using AdaptiveGPT

5 AdaptiveGPT

To demonstrate the feasibility of the proposal in this work, we developed AdaptiveGPT, a proof of concept¹ which autonomously and dynamically adapts the difficulty level of questions sent to students. The Python version implemented and evaluated in this work is available at GitHub² [36] that is functional and can be executed,³ however, in order to simplify the understanding of the proposal, we have created a structured pseudocode, which can be seen in 1. Unlike the sequence diagram presented in Fig. 1, AdaptiveGPT does not distinguish between the scope of the teacher and the student, as it is a proof of concept that allows for the manipulation of messages exchanged between the application and the

¹ A Proof of Concept (PoC) is a term used to describe a demonstration conducted to validate the viability of an idea or concept, whether in the field of Information Technology or business.

² https://github.com/andreiasachete/AdaptiveGPT/blob/main/AdaptiveGPT_en.ipynb

³ It is necessary for the user to have an API access key from OpenAI and insert it in the designated location in the code.

OpenAI API. However, it can serve as a foundation for future implementations that make this distinction (Figs. 2, 3, 4 and 5).

AdaptiveGPT was developed to adapt questions to students within three difficulty levels (easy, medium, and hard). The algorithm always starts with the easy level and maintains a loop until the student correctly answers a difficult question. To achieve this, the user initially sends context to the OpenAI API because the AI needs to understand the corpus boundaries and the current difficulty level to generate questions. Therefore, the SendContext() function submits the context to the OpenAI API within the loop that follows the difficulty level. The SendPromptQuestion() function requests an appropriate question from the API that matches the current difficulty level (initially declared as 'easy').

Algorithm 1 AdaptiveGPT.

```

Ensure: difficulty = 'easy', trajectory[ ], question, answer
1: read context
2: Send_Context(context)
3: while difficulty != 'end' do
4:   SendPromptQuestion(difficulty)
5:   Receive_Question(question)
6:   Answer(question)
7:   Send_Answer(question)
8:   Receive_Evaluation(answer)
9:   if answer = correct and difficulty = 'easy' then
10:     difficulty = 'medium'
11:   else
12:     if answer = correct and difficulty = 'medium' then
13:       difficulty = 'hard'
14:     else
15:       if answer = correct and difficulty = 'hard' then
16:         difficulty = 'end'
17:       end if
18:     end if
19:   end if
20:   trajectory.add[difficulty]
21: end while
22: trajectory[ ]

```

The OpenAI API creates and sends a question to the student that is compatible with the context and the current difficulty level. The student, in turn, responds reflectively (in a written, discursive manner). This response is submitted to evaluate its correctness by the AI through the Send Answer() function. The algorithm receives the correctness of the response. If correct, it uses conditionals to increase the difficulty level within the loop until the student correctly answers a difficult-level question.

If the student does not answer the question correctly, the AI will create a new question at the same difficulty level. In the best-case scenario, a student will answer three questions about the provided context because they will be given an easy question. If they answer correctly, they will be given a medium-level question. If they continue to answer correctly, a difficult-level question will be presented. If the student answers the difficult-level question correctly, the algorithm concludes the learning path. During each interaction, regardless of

	Strongly Disagree 1			Strongly Agree 5		
1. I would like to use this system frequently.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. The system is unnecessarily complex.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. The system is easy to use.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. I would need help from someone with technical knowledge.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. The system's various functions are integrated.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6. The system has a lot of inconsistency.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7. I imagine people will learn to use the system quickly.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8. The system is difficult to use.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9. I felt very confident using the system.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10. I needed to learn several new things before using the system.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fig. 3 SUS questions form

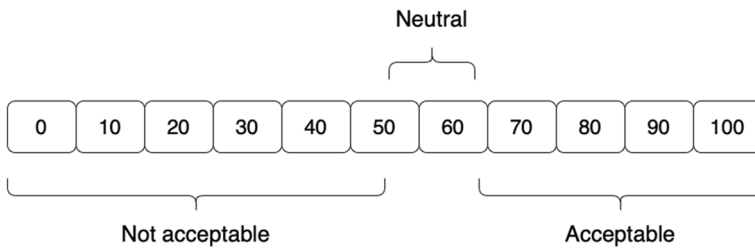
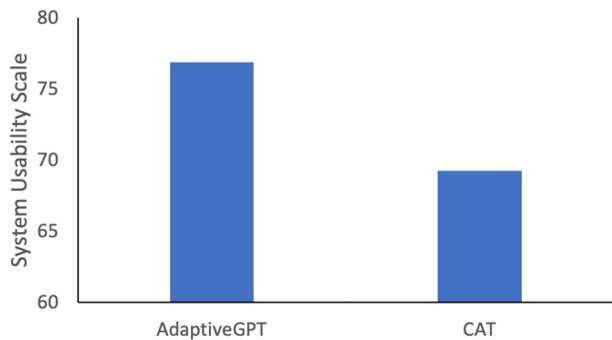


Fig. 4 System Usability Scale

Fig. 5 Usability comparison between AdaptiveGPT and CAT



the difficulty level, the difficulty level is stored, and at the end of the assessment, the learning trajectory is presented.

However, since the trajectory is individualized, and each student will have his or her own path, learning trajectories, in general, can be quite heterogeneous. The AI that underpins this proposal can dynamically adjust to each of these behaviors.

An example of the use of the AdaptiveGPT, with the exchange of prompts between students and AI, can be seen in Fig. 6. The context inserted by the teacher in this example

was “verbs in English” taken from Wikipedia.⁴ On the left side, the AI initially generates an easy-level question about the context. On the right side are student interactions. The student correctly answers the easy-level question, and when the AI recognizes that the answer is correct, the AI generates a medium-level question. When the student answers the medium-level question correctly, the AI generates a difficult-level question. The interaction ends when the student answers the difficult-level question, and the AI recognizes this answer as correct.

In this example, the student answered all the questions correctly. If the student answers incorrectly, the AI generates questions of the same level until the student answers correctly; that is, the AI only advances in difficulty once the student manages to get a question of the same level correct.

6 Evaluation

In this section, the methodology for evaluating the usability of AdaptiveGPT is presented, along with a discussion of the results obtained.

7 Method

Between 2023 and 2024, a group of 30 teachers from various fields of knowledge belonging to an educational institution in the municipality of Alegrete, RS, Brazil, conducted a usability evaluation of AdaptiveGPT. The teachers had access to the code and executed it using the Google Colab platform. They used the System Usability Scale (SUS) [37], a usability evaluation methodology for systems commonly used in the User Experience (UX) Design community. SUS was developed in 1986 by John Brooke and consists of 10 questions to be answered on a Likert scale. SUS can assess three fundamental points: Effectiveness, Efficiency, and Satisfaction. Effectiveness assesses whether the user can complete a task and achieve a goal; efficiency evaluates the amount of resources spent to achieve a goal; and satisfaction assesses the level of comfort for the user in achieving a goal.

Figure 3 presents the questionnaire used for usability assessment. Responses are weighted from 1 (Strongly Disagree) to 5 (Strongly Agree). The final SUS score is calculated based on the following criteria:

- We add up the points obtained in the odd-numbered questions and, from this sum, the value 5 is subtracted.
- We add up the points obtained in the even-numbered questions and 25 is subtracted from this subtotal.
- The points resulting from the two calculations above are added together and
- multiplied by 2.5.

The final result will be a value between 0 and 100, and the usability of the evaluated system can be compared to the scale in Fig. 4. Furthermore, the SUS result was used to compare AdaptiveGPT against a state-of-the-art tool in a broader evaluation that considers user experience.

⁴ https://en.wikipedia.org/wiki/English_verbs

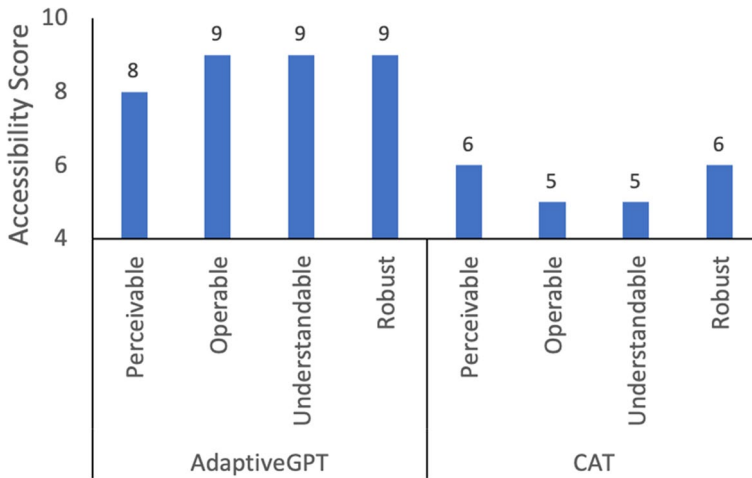


Fig. 6 Accessibility comparison between AdaptiveGPT and CAT

8 Results

After making the AdaptiveGPT available to the teachers, we explained how the code execution, context inclusion, and tracking of the learning path occur. Subsequently, the teachers used AdaptiveGPT and answered the 10 SUS questions to evaluate the usability of the proposal. The values of this distribution were placed in the set of equations in Eq. 1 to calculate the SUS score and assess the usability of the implemented proposal.

$$SUS = \left(\left(\left(\frac{\sum Q.1 + \sum Q.3 + \sum Q.5 + \sum Q.7 + \sum Q.9}{20} \right) - 5 \right) + \left(25 - \left(\frac{\sum Q.2 + \sum Q.4 + \sum Q.6 + \sum Q.8 + \sum Q.10}{20} \right) \right) \right) \times 2.5 \quad (1)$$

$$SUS = \left(\left(\left(\frac{4.75 + 4.2 + 4.3 + 4.3 + 4.55}{20} \right) - 5 \right) + \left(25 - \left(\frac{2.5 + 2.2 + 2.25 + 2.05 + 2.35}{20} \right) \right) \right) \times 2.5 \quad (2)$$

$$SUS = (((22.1) - 5) + (25 - (17.1))) \times 2.5 \quad (3)$$

$$SUS = 76.87 \quad (4)$$

We present the resolution of the equation that evaluates the SUS in Eqs. 1, 2 and 3. We can see, in the set of equations presented that the total score obtained by the SUS questionnaire reached 76.87 points (Eq. 4). Based on the SUS score, we can see that, in terms of usability, AdaptiveGPT is practically in the center of the acceptable range of the SUS. This means that AdaptiveGPT is effective, efficient, and satisfactory for users (teachers) with different expertise levels.

After teachers assessed the usability of AdaptiveGPT, they were asked to test the Computer-Adaptive Testing⁵ (CAT) tool, designed to support adaptive learning in Moodle. The

⁵ https://github.com/middlebury/moodle-mod_adaptivequiz

objective was to compare it with AdaptiveGPT, focusing on user experience (UX), including usability, accessibility, and graphic design.

- Usability is assessed using the System Usability Scale model presented previously.
- Accessibility is assessed through a POUR model [38], composed of four metrics: Perceivable (this means that users must understand the information being presented), Operable (this means that users must be able to use the interface and it cannot require interaction that a user cannot perform), Understandable (this means that users must be able to comprehend information as well as how to use the interface), and Robust (this means as technologies and user agents advance and evolve, content should remain accessible). To this end, a score from 0 to 10 is assigned to each of the four items.
- Graphic Design was evaluated across four key metrics: color, typography, layout, and images. Each metric was assigned a score from 0 to 10, providing a comprehensive assessment of the visual aspects of the tool.

Figure 5 presents the usability comparison between the two software evaluated. The usability score of AdaptiveGPT, already known in the previous evaluation, was higher than that of CAT. This superiority can be attributed to the intrinsic capabilities of AdaptiveGPT, which uses generative AI to create and evaluate questions in real-time. On the other hand, CAT requires that all questions, at different difficulty levels, be previously registered in the database by the teacher, which significantly increases the tool's implementation time.

Figure 6 presents the accessibility assessment between the two tools, considering the four metrics of the POUR model. AdaptiveGPT stood out compared to CAT. This advantage is also due to the previously mentioned usability issues, as AdaptiveGPT offers all the flexibility of a generative AI model that collaborates with the teacher.

For the third set of UX metrics that concern graphic design, Fig. 7 shows that although the results between AdaptiveGPT and CAT are close, both tools still fail in terms of graphic presentation to users. However, there is a point in favor of AdaptiveGPT because it is still a proof of concept, and one of the future works proposed in this work consists of the evolution of its graphical interface.

9 Conclusions

The importance of adaptive learning lies in the fact that each learner is unique, with different skills, knowledge, and interests. By tailoring education to individual needs, adaptive learning allows students to progress at their own pace and maximize their learning potential. Additionally, it can identify specific knowledge gaps and provide personalized interventions to help them overcome them.

However, several new technologies can aid in developing digital adaptive learning proposals and are not being fully utilized. Few proposals genuinely harness artificial intelligence's capabilities, especially when it comes to generative artificial intelligence that can create textual and visual elements in a way very close to human inventive capabilities. ChatGPT is one of these technologies that has become widespread in academic circles as an option to support pedagogical and educational practices. Moreover, OpenAI,

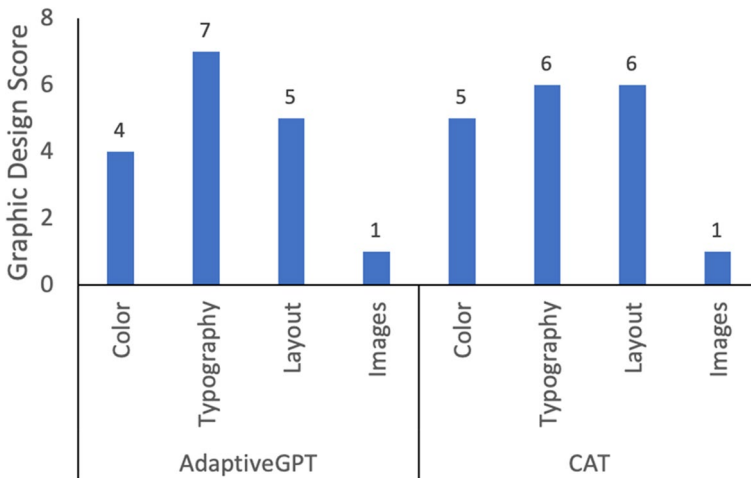


Fig. 7 Graphic Design comparison between AdaptiveGPT and CAT

which created ChatGPT, offers application developers an API that allows communication between applications and a generic neural network trained to respond to various domains.

Within this context, this article proposed adaptive learning through artificial intelligence. To this end, a proof of concept has been developed and is available for testing, use, and future improvements. Our proof of concept, AdaptiveGPT, understands the context provided by the teacher and generates questions for students at increasing levels of difficulty based on the correctness of the student's written responses. The evaluations have shown that the implemented proposal is user-friendly and could easily be incorporated into in-person and remote teaching activities.

During the development of the solution, we encountered some limitations. Most of them are related to the responses provided by the API to the prompts submitted by the application. Sometimes, the API responds with something that is not requested or accepted as an answer, impacting the source code's conditionals that require exact words as responses. Another important point concerns the printing of learning trajectories, which sometimes include some additional information that is irrelevant to the defined key-value pairs. Finally, in some cases where the student answers multiple questions incorrectly, keeping them at the same difficulty level, GPT-3.5 tends to rephrase the same question differently (especially in cases where the initial context is small and leaves little room for variation).

10 Future work

The source code of AdaptiveGPT available on Google Colab can analyze other metrics, as described in the code, such as detecting sentiment in the response text or mood (positive, negative, or neutral). However, since the goal of this work was to develop only a proof of concept, we chose to focus only on the correctness of the response at this moment.

This work is the initial attempt to create adaptive instructional content. The model used in this article is a generalist and still has some limitations. Therefore, in future work, fine-tuning will be performed to make it more specific to the domain of the work, thus avoiding the limitations mentioned above.

Afterward, we intend to incorporate these new criteria into the difficulty level advancement conditions and expand the studies to see what other textual nuances the OpenAI API can detect. We also intend to improve the AdaptiveGPT graphical interface to be compatible with most Virtual Learning Environments.

Data availability Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

- Chen L, Dubrawski A (2017) Learning from learning curves: discovering interpretable learning trajectories, LAK '17. Association for Computing Machinery, New York, pp 544–545. <https://doi.org/10.1145/3027385.3029449>
- Keefe JW, Jenkins JM (2008) Personalized instruction: the key to student achievement, 2nd edn. Rowman & Littlefield Education
- Shubin I, Skovorodnikova V, Kozyriev A, Pitiukova M (2019) Mining methods for adaptation metrics in e-learning. COLINS 2362:1–13
- Lin L, Long D (2023) Generative AI futures: a speculative design exploration. C&C '23:380–383. <https://doi.org/10.1145/3591196.3596616>
- Vaithilingam P, Zhang T, Glassman EL (2022) Expectation vs. experience: Evaluating the usability of code generation tools powered by large language models, CHI EA '22 Association for Computing Machinery, New York. <https://doi.org/10.1145/3491101.3519665>
- Essa SG, Celik T, Human-Hendricks NE (2023) Personalized adaptive learning technologies based on machine learning techniques to identify learning styles: a systematic literature review. IEEE Access 11:48392–48409
- Tong JR, Lee TX (2023) Trustworthy ai that engages humans as partners in teaching and learning. Computer 56:62–73
- Yadlapally DK et al (2023) A review on the potential of ai voice assistants for personalized and adaptive learning in education. In: 7th international conference on computing methodologies and communication (ICCMC), Erode, pp 422–427. <https://doi.org/10.1109/ICCMC56507.2023.10084208>
- Naim A, Alshawaf SM, Kumar Malik P, Singh R (2023) Effective e-learning Practices by Machine Learning and Artificial Intelligence. In: International conference on artificial intelligence and smart communication (ASIC), Greater Noida, pp 491–495. <https://doi.org/10.1109/ASIC56616.2023.10085391>
- Gkountara DN, Prasad R (2022) A review of artificial intelligence in foreign language learning. In: 25th international symposium on wireless personal multimedia communications (WPMC), Hering, pp 134–139. <https://doi.org/10.1109/WPMC55625.2022.10014767>
- Shao S et al (2022) AI-based arabic language and speech tutor. In: IEEE/ACS 19th International Conference on Computer Systems and Applications (AICCSA), Abu Dhabi, pp 1–8. <https://doi.org/10.1109/AICCSA56895.2022.10017924>
- Tang Y, Hare R, Ferguson S (2022) Classroom evaluation of a gamified adaptive tutoring system. In: IEEE frontiers in education conference (FIE), Uppsala, pp 1–5. <https://doi.org/10.1109/FIE6618.2022.9962718>
- Kozlowski N, Unold O (2018) Integrating anticipatory classifier systems with OpenAI gym. In: Genetic evolutionary computation conference companion (GECCO '18), association for computing machinery, New York, pp 1410–1417. <https://doi.org/10.1145/3205651.3208241>

14. Liu Q et al (2019) Exploiting cognitive structure for adaptive learning. In: 25th ACM SIGKDD international conference on knowledge discovery and data mining (KDD '19). Association for Computing Machinery, New York, pp 627–635. <https://doi.org/10.1145/3292500.3330922>
15. Zliobaite I et al (2012) Next challenges for adaptive learning systems. SIGKDD Explor News 14:48–44. <https://doi.org/10.1145/2408736.2408746>
16. Uppsala Borotic G, Jagust T (2022) Enhancing student engagement with personalized gamification and adaptive learning strategies. In: IEEE frontiers in education conference (FIE), pp 1–5. <https://doi.org/10.1109/FIE56618.2022.9962647>
17. Kusumastuti DL, Meyliana HAN, Prabowo H (2021) Models of adaptive learning system in MOOC: A systematic literature review. In: 9th international conference on information and education technology (ICIET). Okayama, pp 242–246. <https://doi.org/10.1109/ICIET51873.2021.9419580>
18. Kuniyoshi K, Kurahashi S (2020) How to use adaptive learning in the classroom? Teaching simulation with adaptive learning on the complex doubly structural network. In: 59th annual conference of the society of instrument and control engineers of Japan (SICE), Chiang Mai, pp 1043–1048. <https://doi.org/10.23919/SICE48898.2020.9240425>
19. Panicker RC, Kumar A, Srinivasan D, John D (2018) Adaptive Learning Analytics in Engineering Education. In: IEEE International conference on teaching, assessment, an learning for engineering (TALE), Wollongong, pp 1193–1196. <https://doi.org/10.1109/TALE.2018.8615200>
20. Mercado J et al (2023) Work in progress: A didactic strategy based on machine learning for adaptive learning in virtual environments. In: IEEE world engineering education conference (EDUNINE), Bogota, pp 1–4. <https://doi.org/10.1109/EDUNINE57531.2023.10102846>
21. Qu Y, Ogunkunle O (2021) Enhancing the intelligence of the adaptive learning software through an AI assisted data analytics on students learning attributes with unequal weight. In: IEEE Frontiers in Education Conference (FIE), Lincoln, pp 1–6. <https://doi.org/10.1109/FIE49875.2021.9637387>
22. Albert D, Steiner CM (2011) Reflections on the evaluation of adaptive learning technologies. In: IEEE international conference on technology for education, Chennai, pp 295–296. <https://doi.org/10.1109/T4E.2011.62>
23. Alrifai M, Dolog P, Nejdil W (2006) Learner profile management for collaborating adaptive eLearning applications. In: International workshop on adaptivity, personalization and the semantic web (APS '06). ACM, pp 31–34. <https://doi.org/10.1145/1149933.1149938>
24. MacNeil S et al (2022) Generating diverse code explanations using the GPT-3 large language model. In: ACM conference on international computing education research - vol 2 (ICER '22). ACM, pp 37–39. <https://doi.org/10.1145/3501709.3544280>
25. Ohagi M, Aizawa A (2022) Pre-trained transformer-based citation context-aware citation network embeddings. In: 22nd ACM/IEEE joint conference on digital libraries (JCDL '22). ACM, New York. <https://doi.org/10.1145/3529372.3533290>
26. Liu A-A et al (2021) Adaptively clustering-driven learning for visual relationship detection. IEEE Trans Multimedia 23:4515–4525
27. Sudhakaran S et al (2024) MarioGPT: Open-ended text2level generation through large language models. In: 37th international conference on neural information processing systems (NIPS '32), vol 36. Curran Associates Inc., Red Hook, pp 544213–554227
28. Wen X et al (2024) CdcM: Chatgpt-aided diversity-aware causal model for interactive recommendation. IEEE Trans Multimedia 26:6488–6500
29. Shu L et al (2024) RewritelM: An instruction-tuned large language model for text rewriting. Proc AAAI Conf Artif Intell 38:18970–18980
30. Beurer-Kellner L, Fischer M, Vechev M (2023) Prompting is programming: A query language for large language models. Proc. ACM Program. Lang, 7, PLDI, 24 p. <https://doi.org/10.1145/3591300>
31. Montagna S et al (2024) LLM-based solutions for healthcare Chatbots: a comparative analysis. In: IEEE international conference on pervasive computing and communication workshops and other affiliated events (Per-Com Workshop), Biarritz, pp 346–351. <https://doi.org/10.1109/PerComWorkshops59983.2024.10503257>
32. Wang J et al (2024) Software testing with large language models: Survey, landscape, and vision. IEEE Trans Software Eng 50:911–936
33. Mishra M et al (2024) Dataagent: evaluating large language models' ability to answer zero-shot, natural language queries. In: IEEE 3rd international conference on ai cybersecurity (ICAIC), pp 1–4
34. Bourhis P, Reutter JL, Suarez F, Vrgoc D (2017) JSON: Data model, query languages and schema specification. In: 36th ACM SIGMODSIGACT-SIGAI symposium on principles of database systems (PODS '17). ACM, New York, pp 123–135. <https://doi.org/10.1145/3034786.3056120>
35. Booch G, Rumbaugh J, Jacobson I (2005) Unified modeling Language user guide, the (2nd Edition) (Addison-Wesley object Technology Series). Addison-Wesley Professional, p 26

36. Nelson MJ, Hoover AK (2020) Notes on using Google colab in AI education. In: ACM conference on innovation and technology in computer science education (ITICSE '20), ACM, New York, pp 533–534. <https://doi.org/10.1145/3341525.3393997>
37. McLellan S, Muddimer A, Peres SC (2012) The effect of experience on system usability scale ratings. *J Usability Stud* 7:56–67
38. Wille K, Dumke RR, Wille C (2016) Measuring the accessibility based on web content accessibility guidelines. In: Joint conference of the international workshop on software measurement and the international conference on software process and product measurement (IWSM-MENSURA), Berlin, pp 164–169. <https://doi.org/10.1109/IWSM-Mensura.2016.032>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.