TutorCraftEase: Enhancing Pedagogical Question Creation with Large Language Models

Wenhui Kang
Institute of Software, Chinese Academy of Sciences
Beijing, China
ks_moth@163.com

Lin Zhang[*]
University of Stuttgart
Stuttgart, Germany
zhangln@studi.informatik.uni-stuttgart.de

Xiaolan Peng[†]
Institute of software,Chinese Academy of Sciences
Beijing, China
xiaolan@iscas.ac.cn

Hao Zhang
Institute of software,Chinese Academy of Sciences
Beijing, China
zhanghao2018@iscas.ac.cn

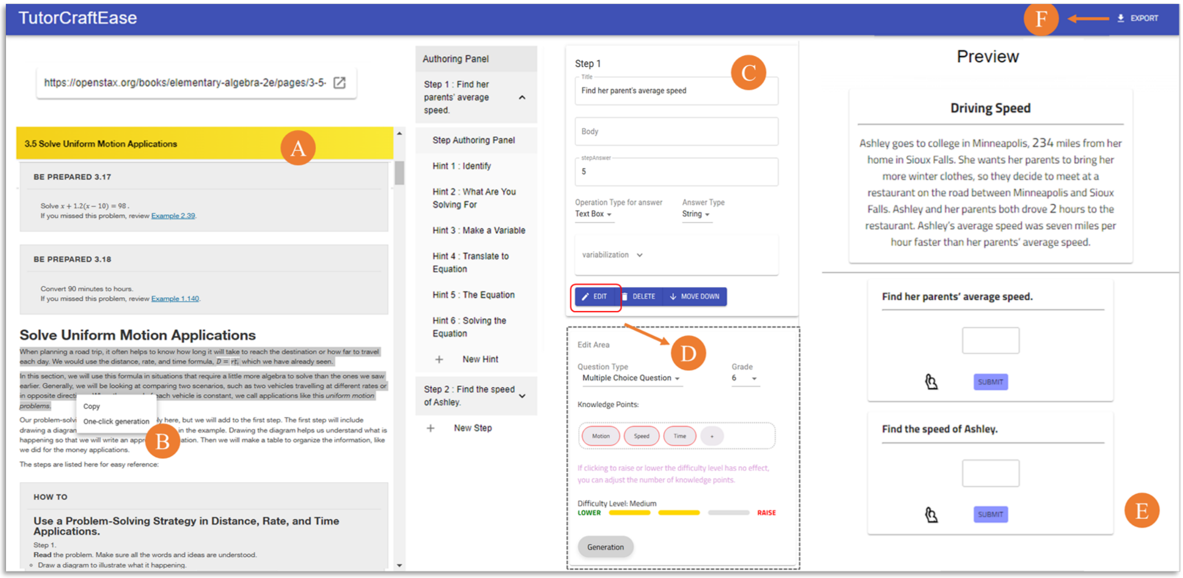
Anchi Li
College of Computer Science
Beijing University of Technology
Beijing, China
lac0405@outlook.com

Mengyao Wang
Institute of Automation, Chinese Academy of Sciences
Beijing, China
wangmengyao2023@ia.ac.cn

Jin Huang[†‡]
Institute of software,Chinese Academy of Sciences
Beijing, China
huangjin@iscas.ac.cn

Feng Tian[§]
Institute of software, Chinese Academy of Sciences
Beijing, China
tianfeng@iscas.ac.cn

Guozhong Dai
Institute of software, Chinese Academy of Sciences
Beijing, China
dgz@iscas.ac.cn


TutorCraftEase
F
EXPORT
https://openstax.org/books/elementary-algebra-2e/pages/3-5-
A
B
Copy
One-click generation
Authoring Panel
Step 1 : Find her parents' average speed.
Step Authoring Panel
Hint 1 : Identify
Hint 2 : What Are You Solving For
Hint 3 : Make a Variable
Hint 4 : Translate to Equation
Hint 5 : The Equation
Hint 6 : Solving the Equation
New Hint
Step 2 : Find the speed of Ashley.
New Step
C
EDIT
DELETE
MOVE DOWN
D
Generation
Preview
SUBMIT
SUBMIT
E

3.5 Solve Uniform Motion Applications

BE PREPARED 3.17

Solve $x + 1.2(x - 10) = 98$.
If you missed this problem, review Example 2.39.

BE PREPARED 3.18

Convert 90 minutes to hours.
If you missed this problem, review Example 1.140.

Solve Uniform Motion Applications

When planning a road trip, it often helps to know how long it will take to reach the destination or how far to travel each day. We would use the distance, rate, and time formula, $D = rt$, which we have already seen.

In this section, we will use this formula in situations that require a little more algebra to solve than the ones we saw earlier. Generally, we will be looking at comparing two scenarios, such as two vehicles travelling at different rates or in opposite directi ach vehicle is constant, we call applications like this *uniform motion problems*.

Our problem-solvi ly here, but we will add to the first step. The first step will include drawing a diagran in the example. Drawing the diagram helps us understand what is happening so that we will write an appr ation. Then we will make a table to organize the information, like we did for the money applications.

The steps are listed here for easy reference:

HOW TO

Use a Problem-Solving Strategy in Distance, Rate, and Time Applications.

Step 1.
Read the problem. Make sure all the words and ideas are understood.
- Draw a diagram to illustrate what it happening.

Step 1
Title: Find her parent's average speed
Body
stepAnswer: 5
Operation Type for answer: Text Box
Answer Type: String
variabilization

Edit Area
Question Type: Multiple Choice Question
Grade: 6
Knowledge Points: Motion, Speed, Time, +

If clicking to raise or lower the difficulty level has no effect, you can adjust the number of knowledge points.

Difficulty Level: Medium
LOWER RAISE

Driving Speed

Ashley goes to college in Minneapolis, 234 miles from her home in Sioux Falls. She wants her parents to bring her more winter clothes, so they decide to meet at a restaurant on the road between Minneapolis and Sioux Falls. Ashley and her parents both drove 2 hours to the restaurant. Ashley's average speed was seven miles per hour faster than her parents' average speed.

Find her parents' average speed.

Find the speed of Ashley.

Figure 1: TutorCraftEase's interface consists of three panels: The *Reference Panel* (A) enables users to load online textbooks via URL, where selecting text triggers an LLM-powered question generation menu (B), offering options for text copying or pedagogical question creation. The *Authoring Panel* (C) enables navigation and refinement of LLM-generated questions, allowing interactive editing of question attributes (D), modification of existing questions, or creation of new ones from scratch. The *Preview Panel* (E) provides real-time editing review, while the *Exportation Button* (F) allows final question export.

Abstract

Pedagogical questions are crucial for fostering student engagement and learning. In daily teaching, teachers pose hundreds of questions to assess understanding, enhance learning outcomes, and facilitate the transfer of theory-rich content. However, even experienced teachers often struggle to generate a large volume of effective pedagogical questions. To address this, we introduce TutorCraftEase, an interactive generation system that leverages large language models (LLMs) to assist teachers in creating pedagogical questions. TutorCraftEase enables the rapid generation of questions at varying difficulty levels with a single click, while also allowing for manual review and refinement. In a comparative user study with 39 participants, we evaluated TutorCraftEase against a traditional manual authoring tool and a basic LLM tool. The results show that TutorCraftEase can generate pedagogical questions comparable in quality to those created by experienced teachers, while significantly reducing their workload and time.

CCS Concepts

• **Human-centered computing → Interactive systems and tools**; HCI design and evaluation methods.

Keywords

large language models, intelligent tutoring systems, human-AI collaboration

ACM Reference Format:
Wenhui Kang, Lin Zhang, Xiaolan Peng, Hao Zhang, Anchi Li, Mengyao Wang, Jin Huang, Feng Tian, and Guozhong Dai. 2025. TutorCraftEase: Enhancing Pedagogical Question Creation with Large Language Models. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26–May 01, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 22 pages. https://doi.org/10.1145/3706598.3713731

1 Introduction

Questioning and answering are central to formal pedagogy [29, 88, 109]. Educators use questions to assess students' knowledge, enhance understanding, and stimulate critical thinking [12]. On a typical school day, high school teachers ask an average of 395 questions, while primary school teachers ask about 348 questions [9, 33]. These questions, known as pedagogical questions [25, 26, 91], serve various instructional purposes, ranging from checking classwork to promoting thoughtful reflection. They are also a major component of Intelligent Tutoring Systems (ITS) [28, 71, 97], where well-crafted pedagogical questions are used to spark students' curiosity and help them gain new insights on a large scale [66, 89].

Despite the fact that novel interactive systems have greatly enhanced the effectiveness of pedagogical questions in education and training, creating a large number of these questions in a limited time remains a challenge for novice teachers. Reports indicate that traditional ITS questions creation tools require up to 300 hours of development to produce just one hour of teaching content [72], posing a substantial barrier for teachers with heavy workloads. Although tools such as Cognitive Tutor Authoring Tools (CTAT) [3], ASPIRE [67], Simulated Learners [105], and OAtutor [77] were developed to simplify the content creation process, they often come with steep learning curves and require programming or specialized editing skills, imposing a considerable psychological burden on users.

The emergence of large language models (LLMs) has opened new frontiers in assisting with content creation tasks, leading to the development of various human-AI co-creation tools for writing [24], brainstorming [116], and visual art design [14]. Moreover, LLMs have begun to revolutionize educational applications, enabling automated feedback systems [65], and the creation of reading comprehension exercises [107]. This drives us to consider using LLMs to support the creation of pedagogical questions in ITS systems, enabling novice teachers to quickly generate high-quality questions and enrich the question banks of these systems.

This paper presents an interactive generation system called TutorCraftEase, driven by LLM (GPT-4o [76]), to quickly create pedagogical questions. Our system is based on OATutor, a fully implemented adaptive tutoring system grounded in the principles of ITS that provides researchers with a flexible environment to verify the effects of pedagogical questions. Unlike reading comprehension exercises [107], the OATutor system standardizes questions by breaking each one into multiple solution steps, with each step including a title, body, and answer, along with hints and scaffolding. This standardized approach helps teachers quickly identify students' difficulties in understanding and allows for timely intervention and adjustment of teaching strategies.

The design and implementation of *TutorCraftEase* are rooted in the principles of Human-AI Interaction (HAI) [54, 108], aiming to provide robust AI support to alleviate the challenges associated with pedagogical questions creation. User interface of *TutorCraftEase* consists three main panels (Figure 1): the Reference Panel, the Authoring Panel, and the Preview Panel. These panels are designed to support the iterative process of questions creation, examination, and revision. The Reference Panel serves as a gateway for users to browse and interact with online textbooks, allowing them to select specific text segments as input for the LLM-assisted question creation process. This process adaptively produces pedagogical questions tailored to the chosen text, ranging from detailed questions to specific steps and hints, thus catering to a wide range of user creation needs. Employing a systematic approach known as prompt chaining [34, 106], the process meticulously crafts a question's title, body, steps, and hints in a logical sequence. To ensure the generated content aligns with the format requirements of OATutor, few-shot prompting techniques [16], alongside a format corrector, are utilized. Furthermore, the authoring panel provides interactive editing functionality, allowing users to manually modify generated questions or create new questions by editing the required attributes if

[*] Lin Zhang is also with Institute of software, Chinese Academy of Sciences.
[†] Co-corresponding authors
[‡] Jin Huang is also with the School of Computer Science and Technology, University of Chinese Academy of Sciences.
[§] Feng Tian is also with the School of Artificial Intelligence, University of Chinese Academy of Sciences.

CHI '25, Yokohama, Japan

ACM ISBN 979-8-4007-1394-1/25/04
https://doi.org/10.1145/3706598.3713731

the existing question does not meet the requirements. The preview panel is used to review the question generated by the LLM.

We conducted a user study with 36 + 3 participants to compare TutorCraftEase with the original spreadsheet-based authoring tool used in OATutor [77] and another tool with basic LLM support. In this study, 36 participants took part in a within-subjects study to evaluate the efficiency and user experience of TutorCraftEase in creating pedagogical questions. 3 experienced teachers were invited to assess the quality of the pedagogical questions generated by the three tools. The results demonstrated TutorCraftEase is efficient, producing pedagogical questions more quickly and with less effort, while maintaining quality comparable to that of questions created by experienced teachers. Furthermore, self-reported feedback from the study underscored a strong preference for TutorCraftEase among the participants, who particularly praised its usability and the streamlined process for creating pedagogical questions. Participants also noted that TutorCraftEase enabled them to produce pedagogical questions more closely aligned with their creative objectives and to broaden their educational perspectives.

The contributions of this paper are as follows:

- We present TutorCraftEase, an authoring tool that simplifies the creation of pedagogical questions. It offers features such as automated question generation and interactive editing of question properties, thereby enhancing the efficiency of collaborative question creation by humans and LLMs.
- We conducted a user study with 39 participants to evaluate TutorCraftEase's performance. We found that the quality of pedagogical questions generated by TutorCraftEase is consistent with those created manually. Additionally, TutorCraftEase has played a positive role in enhancing collaboration between teachers and LLMs in the creation of pedagogical questions and in broadening educational perspectives.
- We discuss the existential issues of LLMs in the authoring of pedagogical questions, including the effective transformation of LLM output into pedagogical questions, balancing AI-driven creation with maintaining user autonomy, teacher-centered interactive design, and the opportunities and reflections that LLMs bring to education.

2 Related Work

2.1 ITS Authoring Tools

The principles of Intelligent Tutoring Systems [7] have been extensively explored over several decades, leading to diverse implementations aimed at enhancing educational outcomes [100]. Research indicates that ITSs can significantly improve student learning outcomes through on-demand instruction and feedback [30, 62, 94].

However, the authoring tools in ITS are often complex and demand considerable effort from content authors [72], such as ASPIRE [68] requires author to provide a high-level description of the domain, as well as examples of problems and their solutions. Open Adaptive Tutor (OATutor) [77] requires author to create content with Google Spreadsheet. To address these issues, on the one hand, researchers have proposed new content authoring methods, including example-tracing method [4] and SimStudent's tutor authoring method [64], and on the other hand, they have sought to streamline the authoring process and enhance efficiency through the development of more intuitive authoring tools [63, 105], such as tool for assist ITS to generate whole interface and specific components based on high-level requirements [18].

The Cognitive Tutor Authoring Tools (CTAT) represent a significant advancement in this area, simplifying the content creation process by employing example-tracing methods in place of traditional programming. This innovation has dramatically reduced the estimated development time from 200-300 hours for one hour of instruction to just 50-100 hours [3]. Despite these improvements, CTAT, similar to the original authoring tool for OATutor [77], relies on Spreadsheets programming. This approach can introduce inefficiencies, particularly when creating a vast array of questions [105]. In contrast, the ASSISTment Builder [86] represents a further evolution in ITS authoring tools by employing a web-based interface, thereby obviating the need for conventional programming. This platform supports the full life cycle of ITS content creation, from initial development to ongoing maintenance and enhancement as the content is actively used by students. While this approach markedly reduces content creation time to approximately 40 hours for one hour of instruction, it introduces a learning cost for new users [77]. Familiarizing oneself with the ASSISTment Builder's interface and functionalities can be time-consuming, potentially offsetting some of the efficiency gains until users overcome the initial learning curve.

2.2 LLM-Based productivity Tools

Large language models (LLMs) have made significant strides in enhancing productivity and efficiency in recent years, leveraging their strengths in information retrieval, automated text generation, and language understanding [24, 32, 37, 83, 99, 116]. For example, LLMs can assist writers with tasks such as text rewriting, expansion, and narrative continuation [110]; facilitate the collaborative generation of research questions between humans and LLMs [56]; and help journalists discover novel reporting angles from press releases [81]. Beyond directly utilizing the text generated by LLMs, their capabilities can also be integrated as an agent within platforms or applications, further improving users' memory and planning abilities [27, 102]. This integration not only broadens the LLM application scenarios, but also improves user efficiency in task execution through intelligent assistance. In addition, combining the reasoning and semantic extraction capabilities of LLMs with existing algorithms also improves system performance, such as integrating graph-structured representations with LLM-generated text [80], or aligning semantic signals from LLMs with the structural features of Graph Neural Networks (GNNs) [114].

However, in practical applications, the performance of large language models (LLMs) is often constrained by their prompts. For non-experts in computer science, designing and customizing appropriate prompts presents a significant challenge [45, 111]. To assist non-AI experts in addressing this issue, researchers have explored methods to guide LLM output through natural language and interactive prompt-based approaches [111], developed interactive tools to iteratively refine prompts by incorporating custom evaluation criteria [45] and have explored various prompt techniques, including few-shot learning [16], chaining prompts [34, 104, 106], and fill-in-the-blank methods [57].

Furthermore, in the design of LLM-based productivity tools, researchers emphasize the importance of providing effective prompts and support of the user experience (UX) to guide users in fully leveraging LLM capabilities [110, 112]. In the context of 'human-AI collaboration', clearly defining roles and responsibilities can facilitate collaborative creation [41, 59], while enhancing the discoverability, visualization, and interpretability of AI-generated content can improve user understanding and interaction with the system [42, 60].

2.3 LLM-based tools for Educational Purposes

In recent years, the application of large language models (LLMs) in the field of education has been steadily increasing, demonstrating their significant potential to enhance both teaching effectiveness and learning experiences [17]. Educators are increasingly recognizing that LLMs can support the learning process in various ways. For example, LLMs have been used to analyze student preferences [11], provide chatbot services for teachers [5], generate code explanations and teaching materials [43], and assist with academic tasks such as literature reviews [101]. In addition, LLMs have been applied in areas such as adolescent cyberbullying education [36] and providing feedback on learning outcomes [65], further demonstrating their broad applicability.

However, the application of LLMs in education has also sparked discussions among educators about their impact [35], particularly concerning course design, assessment methods, and student abilities [5, 46, 61]. Despite educators' differing attitudes toward the use of LLMs in education, these models have demonstrated impressive capabilities in generating human-like text, understanding context, and solving complex tasks, which can significantly contribute to students' learning process [39, 82, 101].

Creating pedagogical questions using text generation and complex question-solving capabilities of LLMs is a key focus of our work. However, current LLM-generated questions, such as reading quiz questions [60], question-answer pairs [55], and English practice questions [107], typically consist only of questions and answers. While useful for practice, they lack step-by-step guidance for teachers and do not identify specific student difficulties or provide auxiliary materials, such as hints or scaffolding, for incorrect answers. Additionally, there is limited exploration of how LLMs can enhance pedagogical question creation for Intelligent Tutoring Systems (ITS), particularly in streamlining and enhancing question development process.

To address the challenges of insufficient supporting materials, limited step-by-step guidance, and inefficiencies in question creation, we developed a pedagogical question creation tool that harnesses the generative capabilities of LLMs and adopts the 'title-body-solution steps' framework from OATutor [77]. The tool incorporates methods such as fill-in-the-blank [57] and chain-of-thought [34], utilizing a custom-designed prompt template and a designed question decomposition method to ensure precise question generation and provide step-by-step guidance. By facilitating interactive question editing rather than direct interaction with LLM, it enhances usability and broadens the application of LLM in streamlining and supporting pedagogical question creation.

3 TutorCraftEase

3.1 Design Considerations

In the design and implementation of TutorCraftEase, we considered two aspects of how to design LLM-based pedagogical question creation and how to adhere to interaction design principles that facilitate the creation and editing of questions.

3.1.1 LLM-based creation for pedagogical question.

To enhance the creation of LLM-based pedagogical questions, we build on the question structure from OATutor [77] (detailed in Appendix A) and encourage authors to adjust the teaching granularity as needed. However, the creation of pedagogical questions involves a multi-layered structure, sequentially generating information through LLMs can be inefficient in systems requiring immediate responses. This presents a challenge in designing LLM prompts that align with the process of pedagogical question creation.

To address this challenge, designing precise and well-structured prompts is crucial to generate accurate responses from LLMs [58]. The prompts must strike a balance between complexity and simplicity [17]: overly complex prompts may risk misunderstanding and extended response times, while overly simple prompts may result in general output. To ensure that the generated questions are contextually and semantically relevant and consistent with the materials chosen by the author, it is essential to clearly define the role, task, and expected output of the LLM in the creation process [90]. Additionally, we should integrate discrete prompts for the creation of pedagogical questions into a cohesive chain prompt [53], thus simplifying the creation process and improving efficiency.

3.1.2 Human-AI Interaction System Design Principles.

The integration of AI into HCI systems has led to significant advancements, yet presents unique design challenges, particularly due to the inherent uncertainties in AI capabilities and the complexity of its outputs [108]. In the context of pedagogical questions creation, where accuracy and reliability are paramount, the improper application of AI technologies poses significant risks [60]. Therefore, to navigate these challenges and harness the potential of AI effectively, it is crucial to follow a set of well-established AI system design principles. These principles provide a solid foundation for developing systems that are both user-friendly and resilient to the pitfalls of AI integration.

In the process, we should adhere to the foundational principles of Human-AI Interaction (HAI) within the field of Human-Computer Interaction (HCI), focusing on the following aspects: first, ensuring that the system can efficiently correct errors and establish clear boundaries for AI intervention [6, 40]; second, emphasizing the predictability and controllability of the system to ensure that users can manage the AI's behavior [74]; and third, applying scaffolded prompt engineering to guide users in effectively leveraging AI technologies, thereby enhancing their interaction experience [110, 112]. These strategies provide a solid foundation for the effective application of large language models (LLMs) and further optimize the process of teaching question creation.

3.1.3 The Overall Design of TutorCraftEase. Guided by insights from LLMs and the principles of Human-AI Interaction (HAI) systems, we discussed the overall design for TutorCraftEase. These design proposals aim to optimize the process of creating pedagogical questions for authors by effectively leveraging LLMs, thereby enhancing productivity and ensuring that the quality of the generated questions is comparable to those created manually by teachers.

- **The design of LLM prompt.** To meet the needs of creating pedagogical questions structured in a 'title-body-solution steps' format, LLM prompts are designed to chain together different levels of question structures. Additionally, to better decompose the step within the pedagogical question structure, task analysis and tree decomposition methods are employed to decompose step into a mixture of hints and scaffolding.
- **User interaction with LLM.** Pedagogical question creation tools, which focus on seamlessly integrating AI assistance into the creative process of authors, aim to allow users to easily create, modify, and refine pedagogical questions with minimal effort. In TutorCraftEase, we enhance the interactive performance between humans and LLMs through interactive editing of pedagogical question attributes, as well as real-time monitoring and error correction during the creation process.
- **User interface of TutorCraftEase.** TutorCraftEase is a comprehensive full-stack web application featuring a user interface with a reference panel, an authoring panel and a preview panel. These panels streamline the process of selecting material from textbooks, refining knowledge to create pedagogical questions, manually editing them, and conducting reviews.

3.2 pedagogical question Generation with LLM

For creating the structure of 'title-body-solution steps' in pedagogical questions, we utilize a technique known as chaining prompts [34, 104, 106] and a one-shot approach [16]. They enable us to guide the LLM through a sequence of related prompts, ensuring the generation of coherent and contextually relevant content. For instance, Figure 2 illustrates the process to generate a pedagogical questions based on selected textbook text. This process begins by prompting the LLM to create a concise summary, limited to five words, which serves as the title of pedagogical question. This is followed by a more detailed summary, capped at 30 words, which forms the body. The LLM is then prompted to create a question-answer pair as a solution step, and multiple general hints and scaffolding for this solution step.

To enhance the generation of pedagogical questions, we developed a RICTEF (Role, Input, Constraint, Task, Example, Format) prompt template. As shown in Table 7, we provide a detailed explanation of the purpose of each factor in the RICTEF template, along with corresponding examples to support the explanation. For a detailed description, refer to Appendix B.1. Additionally, to support interactive editing during question creation, we use a fill-in-the-blank approach [23, 57, 113] for custom constraints. The format is as follows: Please design a ${num}-grade ${course} ${question type} question aimed at helping students master knowledge related to ${knowledge points}. The definition of ${question type} contains ${define}, and its elements is ${elements}. Here, 'num' denotes the corresponding grade level in K-12 education, and 'course' refers to the current course for which the pedagogical question is being created. The definition of question type is as detailed in the supplementary material.

For the creation of hints and scaffolding, we use a tree-based decomposition method to break down a complex pedagogical question into a three-layer question tree by task analysis [8, 20]. This process is formally expressed as follows:

$$P = (P_0, \{P_1, P_2, ..., P_n\}) \tag{1}$$

where, P_0 represents the root question, which is the pedagogical question that needs to be decomposed, P_i is the $i-th$ sub-question tree, and n denotes the number of sub-question trees (called solution steps in the structure of pedagogical question) derived from decomposing the complex question. For each P_i, we further decompose it into a series of hints (or scaffolding), which form the leaf nodes of the question tree, represented as:

$$P_i = \{H_{i1}, H_{i2}, ..., H_{im}\} \tag{2}$$

where m is the number of hints for the $i-th$ sub-question tree, and H_{ij} represents the $j-th$ hint in the $i-th$ sub-question tree. Finally, the solution of question P can be obtained by combing the solutions of its hints.

$$Solution(P) = \sum_{i=0}^{n} Solution(P_i) \tag{3}$$

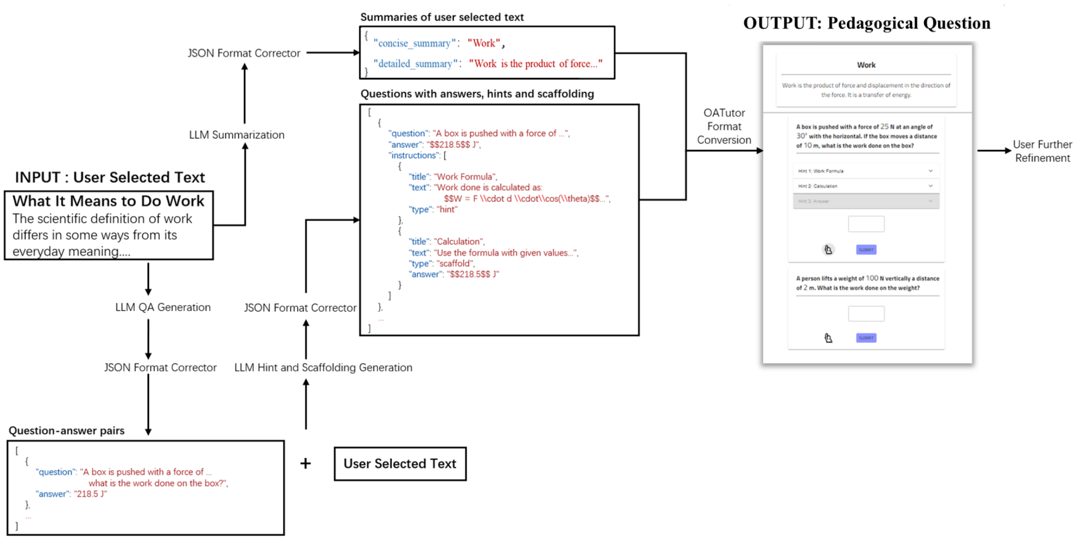
In generating hints and scaffolding, we also achieve this by imposing constraints on the prompt, including the number of decomposition levels, subquestions, and solution steps. Additionally, we include selected texts from reference panel, the complex pedagogical question and its answer for which hints and scaffolding are to be created in the prompts, fostering a seamless integration within the structure of pedagogical questions.

Finally, we structure the format of the generated questions from the LLM output (detailed in the Appendix B.2) and consolidate the prompts used for generating different granularities of pedagogical questions and the output formats of pedagogical question into a unified prompt, which serves as the input for the LLM to generate all components of the pedagogical question.

3.3 Interface Design and Development

With design goals identified in Section 3.1.3, we develop TutorCraftEase: a dynamic, web-based generative tool designed to assist teachers in producing pedagogical questions tailored to the OATutor format. TutorCraftEase empowers authors to efficiently generate and structure pedagogical questions, drawing directly from textbook texts to populate a 'title-body-solution steps' hierarchical framework, and facilitates real-time review.

As depicted in Figure 1, the user interface of TutorCraftEase consists three main panels: the Reference Panel, the Authoring Panel, and the Preview Panel. These panels are designed to support seamless navigation between materials reference, active pedagogical question creation, and immediate questions preview. The Reference Panel (Figure 3) allows authors to pinpoint and select textbook segments for LLM-assisted question creation, with a context-sensitive



What It Means to Do Work
The scientific definition of work differs in some ways from its everyday meaning....

```
{ "concise_summary": "Work",
  "detailed_summary": "Work is the product of force..."
}
```

```
[
  {
    "question": "A box is pushed with a force of ...",
    "answer": "$$218.5$$ J",
    "instructions": [
      {
        "title": "Work Formula",
        "text": "Work done is calculated as:
                 $$W = F \\cdot d \\cdot\\cos(\\theta)$$...",
        "type": "hint"
      },
      {
        "title": "Calculation",
        "text": "Use the formula with given values...",
        "type": "scaffold",
        "answer": "$$218.5$$ J"
      }
    ]
  },
  ...
]
```

```
[
  {
    "question": "A box is pushed with a force of ...
                 what is the work done on the box?",
    "answer": "218.5 J"
  },
  ...
]
```

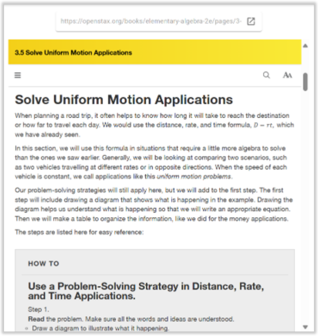
Work

Work is the product of force and displacement in the direction of the force. It is a transfer of energy.

A box is pushed with a force of 25 N at an angle of 30° with the horizontal. If the box moves a distance of 10 m, what is the work done on the box?

A person lifts a weight of 100 N vertically a distance of 2 m. What is the work done on the weight?

Figure 2: The pipeline to generate a pedagogical question.


https://openstax.org/books/elementary-algebra-2e/pages/3-
3.5 Solve Uniform Motion Applications
HOW TO

Solve Uniform Motion Applications

When planning a road trip, it often helps to know how long it will take to reach the destination or how far to travel each day. We would use the distance, rate, and time formula, $D = rt$, which we have already seen.

In this section, we will use this formula in situations that require a little more algebra to solve than the ones we saw earlier. Generally, we will be looking at comparing two scenarios, such as two vehicles travelling at different rates or in opposite directions. When the speed of each vehicle is constant, we call applications like this *uniform motion problems*.

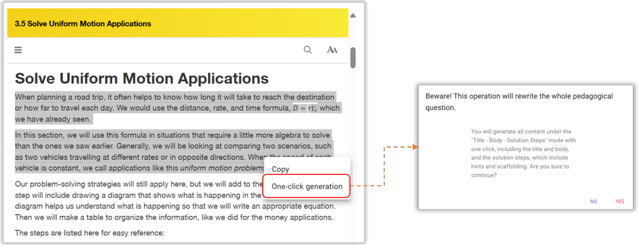
Our problem-solving strategies will still apply here, but we will add to the first step. The first step will include drawing a diagram that shows what is happening in the example. Drawing the diagram helps us understand what is happening so that we will write an appropriate equation. Then we will make a table to organize the information, like we did for the money applications.

The steps are listed here for easy reference:

Use a Problem-Solving Strategy in Distance, Rate, and Time Applications.

Step 1.
Read the problem. Make sure all the words and ideas are understood.
- Draw a diagram to illustrate what it happening.

Figure 3: A Reference Panel showcasing original document created for pedagogical questions.



Solve Uniform Motion Applications

When planning a road trip, it often helps to know how long it will take to reach the destination or how far to travel each day. We would use the distance, rate, and time formula, $D = rt$, which we have already seen.

In this section, we will use this formula in situations that require a little more algebra to solve than the ones we saw earlier. Generally, we will be looking at comparing two scenarios, such as two vehicles travelling at different rates or in opposite directions. When the speed of each vehicle is constant, we call applications like this *uniform motion problem*

Our problem-solving strategies will still apply here, but we will add to the step will include drawing a diagram that shows what is happening in the diagram helps us understand what is happening so that we will write an appropriate equation. Then we will make a table to organize the information, like we did for the money applications.

The steps are listed here for easy reference:

Beware! This operation will rewrite the whole pedagogical question.

You will generate all content under the "Title - Body - Solution Steps" mode with one click, including the title and body, and the solution steps, which include hints and scaffolding. Are you sure to continue?

Figure 4: The LLM-assisted functionality menu for pedagogical question generation.

menu appearing after selection to guide them through the generation process. The Authoring Panel (Figure 5) offers a flexible workspace where authors can adjust question properties before generation or refine existing questions to meet pedagogical needs. The Preview Panel (Figure 7) provides a real-time display of edited questions, allowing users to verify their accuracy and instructional effectiveness.

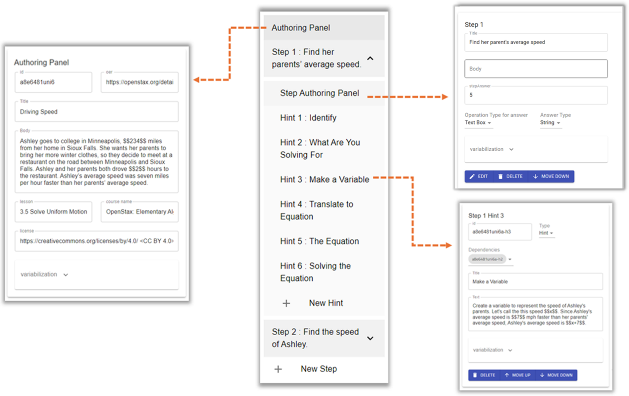
3.3.1 *Reference Panel.*

The Reference Panel (Figure 3) is designed to provide users with an easy way to browse and select materials for generating pedagogical questions. Users can enter the URL of the required materials to directly access textbook pages on the Open Educational Resources (OER) platform, allowing them to quickly retrieve relevant content. To streamline the generation process, TutorCraftEase introduces an LLM-assisted functionality menu, offering a 'one-click generation' button and a 'copy' buttons to quickly create pedagogical questions and copy reference materials, as illustrated in Figure 4. The generated questions consist of applied questions that emphasize the practical application and contextualization of knowledge, making them more effective in assessing students' ability to solve complex problems. The menu does not include options for modifying question types or attributes; instead, these functions are centralized in the Authoring Panel. This design aims to simplify the initial interaction with the reference materials panel and encourages teachers to refine and personalize the generated questions within the Authoring Panel.

During the generation process, the LLM prompts for creating pedagogical questions based on the 'title-body-solution steps' structure were customized, with constraints added to the decomposition of solution steps to optimize the relevance of the pedagogical questions. At the same time, to ensure a coherent authoring process, the system automatically populates the generated questions in the Authoring Panel for easy viewing and further editing.

3.3.2 *Authoring Panel.*

The Authoring Panel (Figure 5) is designed to provide authors with a comprehensive and flexible pedagogical question creation workspace. It features a hierarchical navigation menu that facilitates seamless switching between solution steps, hints and scaffolding. The navigation menu enables authors to

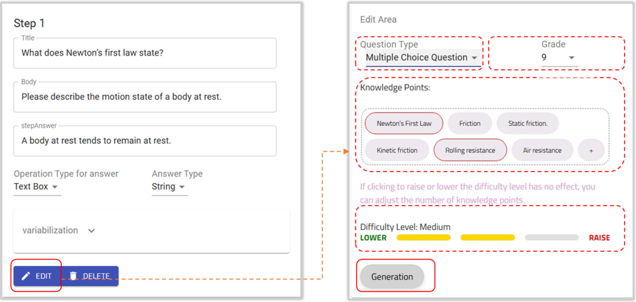


Authoring Panel
id: a8e6481uni6
oer: https://openstax.org/detai
Title: Driving Speed
Body: Ashley goes to college in Minneapolis, $$234$$ miles from her home in Sioux Falls. She wants her parents to bring her more winter clothes, so they decide to meet at a restaurant on the road between Minneapolis and Sioux Falls. Ashley and her parents both drove $$2$$ hours to the restaurant. Ashley's average speed was seven miles per hour faster than her parents' average speed.
lesson: 3.5 Solve Uniform Motion
course name: OpenStax: Elementary Al
license: https://creativecommons.org/licenses/by/4.0/ <CC BY 4.0>

Step 1
Title: Find her parent's average speed
Body
stepAnswer: 5
Operation Type for answer: Text Box
Answer Type: String

Step 1 Hint 3
id: a8e6481uni6a-h3
Type: Hint
Dependencies: a8e6481uni6a-h2
Title: Make a Variable
Text: Create a variable to represent the speed of Ashley's parents. Let's call the this speed $$x$$. Since Ashley's average speed is $$7$$ mph faster than her parents' average speed, Ashley's average speed is $$x+7$$.

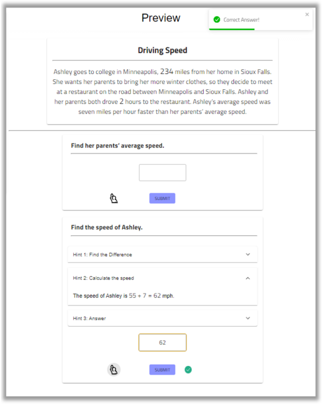
Figure 5: The Authoring Panel showcasing a hierarchical navigation menu for managing soultion steps and hints.



Step 1
Title: What does Newton's first law state?
Body: Please describe the motion state of a body at rest.
stepAnswer: A body at rest tends to remain at rest.
Operation Type for answer: Text Box
Answer Type: String

Edit Area
Question Type: Multiple Choice Question
Grade: 9
Knowledge Points:
If clicking to raise or lower the difficulty level has no effect, you can adjust the number of knowledge points.
Difficulty Level: Medium

Figure 6: Interactive editing area for attributes of newly generated questions.



Preview

Correct Answer!

Driving Speed

Ashley goes to college in Minneapolis, 234 miles from her home in Sioux Falls. She wants her parents to bring her more winter clothes, so they decide to meet at a restaurant on the road between Minneapolis and Sioux Falls. Ashley and her parents both drove 2 hours to the restaurant. Ashley's average speed was seven miles per hour faster than her parents' average speed.

Find her parents' average speed.

Find the speed of Ashley.

Hint 1: Find the Difference

Hint 2: Calculate the speed

The speed of Ashley is 55 + 7 = 62 mph.

Hint 3: Answer

62

Figure 7: The Preview Panel in TutorCraftEase, enabling authors to review and test their edited pedagogical question in real-time.

select and focus on the specific element of a question they wish to edit, enhancing workflow efficiency.

In this panel, three interactive ways are provided for working with LLM-generated pedagogical questions: 1) selecting the properties needed to generate a new question and regenerating it using the editing tool shown in the Figure 6; 2) manually modifying LLM-generated pedagogical questions to address potential inaccuracies; and 3) crafting pedagogical questions from scratch, leveraging one's own expertise and creativity. In addition, this panel supports diverse content forms, accommodating authors' unique instructional objectives. For instance, it allows for the creation of questions in formats such as multiple choice or textbox, and it also enables the crafting of plain hints or scaffolding.

As Figure 6 shows, new pedagogical questions can be generated by controlling its properties, which include question type, grade, knowledge points and difficulty level. The question type is controlled via a dropdown menu, offering options to select question type such as single-choice, multiple-choice, fill-in-the-blank, true/false, calculation, and applied question. The grade level is selected based on the materials chosen from the reference panel, ranging from the grade level corresponding to the selected material up to grade 12. Lower-grade selections are not allowed because students in lower grades may not have learned the knowledge associated with the selected material.

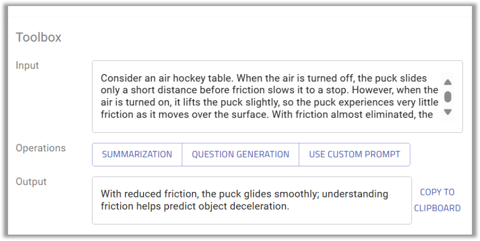
$$Dif = \frac{Dif_b + N \cdot w \cdot Dif_t}{100} \tag{4}$$

$$Dif_t = \begin{cases} 2, t = \text{single-choice question} \\ 2.5, t = \text{true/false question} \\ 3, t = \text{fill-in-the-blank question} \\ 3.5, t = \text{multiple-choice question} \\ 4.5, t = \text{calculation question} \\ 6, t = \text{applied question} \end{cases}$$

where Dif_b is the basic difficulty of question, and N is the number of knowledge points, and w is the difficulty factor per knowledge point, Dif_t is the difficulty factor of question type, determined by calculating the ratio of the number of questions of the same type to their corresponding scores in the exam.

Furthermore, the Authoring Panel integrates advanced features from OATutor[77], such as LaTeX support, hint dependencies, and variabilization, which broaden the range of pedagogical questions that can be created. These capabilities ensure that the Authoring Panel not only facilitates the efficient correction and enhancement of LLM-generated material, but also empowers authors to meticulously craft and customize pedagogical questions, adhering to best practices, and fostering an engaging learning environment.

3.3.3 Preview Panel. The Preview Panel (Figure 7) is designed not only to allow authors to instantly view the pedagogical questions they are refining, but also to interactively test the questions they have crafted, emphasizing the importance of facilitating real-time examination and interaction with edited content. Utilizing the open-source framework of OATutor [92], the Preview Panel precisely emulates the appearance and behavior of pedagogical questions within the actual OATutor environment. This emulation extends to the accurate rendering of LaTeX expressions inputted via the Authoring Panel, ensuring that mathematical formulas and other LaTeX-based content are correctly displayed. Beyond mere visualization, this



Consider an air hockey table. When the air is turned off, the puck slides only a short distance before friction slows it to a stop. However, when the air is turned on, it lifts the puck slightly, so the puck experiences very little friction as it moves over the surface. With friction almost eliminated, the

With reduced friction, the puck glides smoothly; understanding friction helps predict object deceleration.

Figure 8: The interface for using LLM features in the LLM Toolbox includes summarization, question generation, and use custom prompt.

interactive testing feature in the Preview Panel empowers authors to verify the functionality, accuracy, and educational efficacy of questions and materials, enhancing the validation process before the finalization of pedagogical questions.

3.3.4 Data Export. The Data Export feature streamlines the process of integrating authored pedagogical questions into the OATutor system. With a simple click of the 'Export' button, authors can package their completed pedagogical questions into a zip file. This file can then be easily unzipped and imported into OATutor by placing it in the content pool directory.

4 User Study

To evaluate the efficacy of TutorCraftEase, three experimental conditions were assessed through a within-group user study and a blind test survey: TutorCraftEase, LLM Toolbox (a tool providing basic large language model support), and Spreadsheet (the conventional manual authoring tool used by OATutor). The within-group user study evaluated the performance of three experimental conditions in creating pedagogical questions, while the blind test survey assessed the quality of the questions generated under these conditions.

4.1 Conditions and Materials

4.1.1 LLM Toolbox. The LLM Toolbox differs from TutorCraftEase in that it does not include LLM-assisted content generation and automatic content integration, but provides a direct way to use the LLM. Figure 8 shows the interface for accessing the large language model (GPT-4o) features in the LLM Toolbox, including 'summarization', 'question generation', and 'use custom prompt' operations. The 'summarization' feature condenses selected text into brief summaries, which is useful for creating question titles or step descriptions. The 'question generation' creates relevant questions and answers, assisting in the formulation of steps and scaffolding. The 'use custom prompt' option allows the author to use custom-designed prompts to guide and utilize the LLM. Additionally, similar to TutorCraftEase, the LLM Toolbox also supports filling the toolbox input fields by selecting materials from the reference panel via an interactive menu or manual copying.

4.1.2 Spreadsheet. The Spreadsheet replicates the Spreadsheet programming methodology employed by the OATutor editorial team, as detailed by Pardos et al. [77]. Participants in this condition utilize a Spreadsheet interface, as exemplified in Figure 10 (in Appendix A), to edit and structure pedagogical questions. This approach requires the use of a conversion script to translate Spreadsheet data into a format compatible with OATutor.

4.1.3 Reference Material. For our study, we selected sections 4.2, 4.3, and 4.4 from OpenStax's open-access high-school physics textbook [93], focusing on Newton's three laws of motion, and translated these sections into Chinese. Before the experimental session, we distributed the translated textbook in PDF format to participants, asking them to thoroughly review the material. During the sessions, we equipped them with the same Reference Panel used in both TutorCraftEase and the LLM Toolbox for content reference when engaging with Spreadsheet authoring. This ensured consistent access to textbook content across all tools, facilitating a fair comparison of their question creation capabilities. The selected textbook sections were preloaded into each tool's Reference Panel for immediate use during the study.

4.2 Participants

As mentioned before, we conducted two experiments: a within-group user study and a blind test survey. Therefore, we recruited two groups of participants.

For the within-group user study, 36 participants (age of M = 29.18, SD = 7.50; 20 females and 16 males) were recruited to create pedagogical questions. All of the participants are physics teachers: 17 teach physics in high school (K10-K12), and 19 teach physics in middle school (K7-K9). Of them, 26 held bachelor's degrees, 7 had post-graduate degrees, and 3 had college diploma or lower-level education.

For the blind test survey, 3 veteran physics teachers, each with experience in creating pedagogical questions and over five years of teaching experience, were recruited to assess the quality of the questions generated.

4.3 Procedure

In the within-group user study, each participant was instructed to create a total of 9 pedagogical questions based on the textbook materials displayed on the reference panel of each system, with 3 questions created under each experimental condition. The procedure of each participant was as follows: 1) they received a brief tutorial on the system to ensure familiarity with its functionality and were informed their screens would be recorded during the experiment; 2) they used three tools to create 9 pedagogical questions. To mitigate order effects, we counterbalanced the sequence in which participants experienced each condition and randomized the assignment of textbook sections to conditions; 3) Upon completing the tasks, participants completed a post-task questionnaire assessing their user experience, as detailed in Table 1, and ranked the preferred order in which they would use the three tools. Subsequently, participants shared opinions through a semi-structured interview, as detailed in the qualitative analysis section. Each participant took approximately 90 minutes to complete the user study and received 200 RMB for their time.

The post-task questionnaire was constructed by adopting standard metrics of helpfulness, efficiency, usability, enjoyment, and satisfaction in the System Usability Scale (SUS) [10, 96], and effort, mental demand from NASA Task Load Index (NASA-TLX) [13]. Certain metrics from both scales, such as system consistency, temporal demand, and success rate, among others, are not involved as they are not suitable for evaluating the tool in our context. Moreover, the tool we designed focuses on human collaboration with LLMs to generate pedagogical questions. Its performance is reflected mainly in the effectiveness of human-AI collaboration, the degree of control over the AI/LLM, the quality of the generated questions, and the user's sense of achievement during the creation process. Therefore, metrics of creative achievement [38, 52], ownership [103], quality [69, 70], controllability [87], collaboration [44, 52] are also included to provide a more comprehensive evaluation.

In the blind test survey which we adopted from an established method [73], we asked three veteran physics teachers to evaluate 90 pedagogical questions. These questions were randomly selected from the set of questions created by the three tools in the within-group user study, with each tool providing 30 questions. Throughout the blind test survey, each participant evaluated the same set of 90 questions by 1) receiving instructions on how to assess pedagogical questions and a detailed description of three tools; 2) evaluating the quality of each pedagogical question (rated on a 5-point scale from 1 (very low) to 5 (very high)) using a custom-developed interface designed for easy rating and automatic result recording. Each teacher spent about 60 minutes completing the test and was compensated with 300 RMB.

This study was conducted in accordance with the ethical guidelines of the local ethics committee. Prior to participation, all subjects provided informed consent, and their confidentiality and anonymity were strictly maintained throughout the study.

5 Results

In the within-group user study, the pedagogical questions created by participants, their user behaviors, subjective ratings, and open-ended opinions were recorded. In the blind test survey, the quality ratings of the pedagogical questions were collected. To present these results more clearly, we have organized the data into quantitative and qualitative analyses in the following sections.

5.1 Quantitative results

5.1.1 Blind test. In the blind test, participants rated the pedagogical questions created in the within-group user study. The 36 participants in the within-group user study generated a total of 325 valid pedagogical questions: 111 with TutorCraftEase, 106 with the LLM Toolbox, and 108 with the Spreadsheet. Some submissions were excluded as invalid due to duplicate uploads or formatting errors. Examples of pedagogical questions created by participants using TutorCraftEase are detailed in Appendix C. In the blind test survey, three veteran physics teachers evaluated the quality of 90 randomly selected pedagogical questions independently. The inter-rater concordance among the three rating teachers was 0.84, calculated by Fleiss' Kappa coefficient. As the data do not follow a normal distribution (by the Kolmogorov-Smirnov test with $\alpha = 0.05$), a Friedman test (non-parametric repeated measures) was used to analyze the differences across the three conditions.

The results showed that the quality of questions generated by TutorCraftEase ($M = 2.57, SD = 0.14$) was comparable to those created manually by participants using a Spreadsheet ($M = 2.58, SD = 0.24$), while the LLM Toolbox ($M = 2.55, SD = 0.28$) produced the lowest quality. Based on the average values from the blind test, the performance on pedagogical questions from the three tools falls between the middle and high levels. The Friedman test results revealed no significant differences in quality among TutorCraftEase, LLM Toolbox, and Spreadsheet ($\chi^2 = 0.041, p = .980$). Post hoc pairwise comparisons also revealed that there are no significant differences between the tools: $z = -0.215$ ($p = .830$) between TutorCraftEase and LLM Toolbox, $z = -0.206$ ($p = .837$) between TutorCraftEase and Spreadsheet, $z = -.038$ ($p = .970$) between LLM Toolbox and Spreadsheet. This further suggests that the quality of the questions generated by TutorCraftEase can align with the quality of the pedagogical questions created manually.

5.1.2 Subjective rating. As shown in Figure 9, subjective ratings of 13 metrics from the post-task questionnaire of 36 participants are presented (3 tools × 13 questionnaire metrics × 36 participants), with the width of each color representing the number of participants for each rating. As the data did not follow a normal distribution, a Friedman test (non-parametric repeated measures) was conducted to compare differences across the three tools on the given metrics.

As shown in Table 2, except for **ownership**, TutorCraftEase outperformed the LLM Toolbox and Spreadsheet in all other aspects. In particular, TutorCraftEase received a higher rating in the metrics of **helpfulness**, **efficiency**, **usability**, **enjoyment**, **effectiveness**, **effor**, and **mental demand**, exceeding the other tools by more than one point on average. Additionally, we found TutorCraftEase scores on par with the other two tools in terms of quality, creative achievement, and controllability. However, TutorCraftEase scored lower in ownership ($M = 3.81, SD = 1.704$) compared to Spreadsheet ($M = 5.06, SD = 2.028$). While TutorCraftEase enables participants to control the creation of pedagogical questions by setting parameters such as difficulty level, knowledge points, and question types, participants still reported a lack of ownership over the questions created with the tool, perceiving the output as not being fully directed by themselves. It is worth noting that the participants rated the quality of pedagogical questions created using TutorCraftEase as the highest ($M = 5.00, SD = 1.912$), surpassing those created with LLM Toolbox and Spreadsheet by 0.33 and 1.17, respectively.

As shown in Table 3, the Friedman test results revealed significant differences across the 12 metrics for TutorCraftEase, LLM Toolbox, and Spreadsheet, except for controllability ($\chi^2 = 0.066, p = .968$). Post hoc pairwise comparisons revealed that there are no significant differences on usability ($z = -1.768, p = .231$), creative achievement ($z = -1.650, p = .297$), ownership ($z = 1.120, p = .789$), quality ($z = -0.236, p = .814$), collaboration ($z = -0.766, p = .444$), and recommendation ($z = -1.768, p = .231$) between TutorCraftEase and LLM Toolbox. No significant differences were found on quality ($z = -2.062, p = .118$) between TutorCraftEase and Spreadsheet, and on mental demand ($z = -1.473, p = .422$),creative achievement ($z = -1.296, p = .585$), quality ($z = -1.827, p = .203$), and collaboration ($z = -2.003, p = .135$) between LLM Toolbox and Spreadsheet. Interestingly, while the Friedman test showed a significant difference on quality ($\chi^2 = 7.196, p = .027$) across

Table 1: The post-questionnaire for the three tools.

Metrics	Statement (7-point Likert scale)
Helpfulness*	I think this tool is very helpful for creating pedagogical questions.
Efficiency*	I think this tool is efficient in creating pedagogical questions.
Usability*	I think this tool is easy to use.
Enjoyment*	I enjoy creating pedagogical questions with this tool.
Effort*	Using this tool does not require a considerable amount of effort.
Mental Demand*	Using this tool does not requires significant mental or cognitive effort.
Creative Achievement [38, 52]	The pedagogical questions created with this tool feels like my own achievement.
Ownership[103]	I am able to create a pedagogical questions I envision using this tool.
Quality [69, 70]	The pedagogical questions created with this tool can meet pedagogical needs.
Controllability [87]	The pedagogical questions creation process is under my control when using this tool.
Collaboration [44, 52]	I feel that I am collaborating with AI when using this tool.
Satisfaction*	I am satisfied with this tool's performance
Recommendation	I would recommend this tool to others.

* metrics derived from SUS [10, 96], NASA Task Load Index [13], and instrumental papers [31, 48, 50].

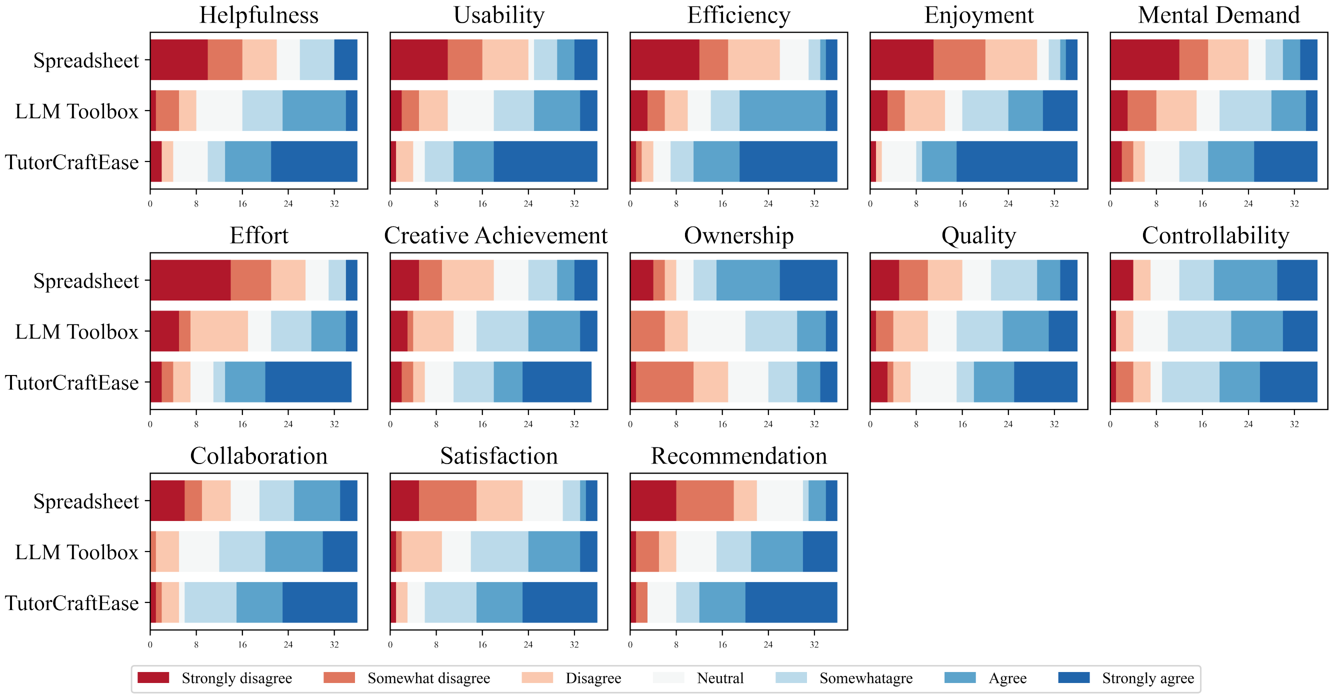


Figure 9: User ratings of the three conditions as derived from the post-task questionnaire.

the three tools, no significant differences were found in post hoc pairwise comparisons.

Further, the participants' ranking of their willingness to use the three tools underscores a clear preference for TutorCraftEase. Of the 36 participants, 22 ranked TutorCraftEase as their top choice, compared to 9 who preferred the LLM Toolbox and 5 who selected the Spreadsheet. Those who favored spreadsheets tended to be veteran educators skilled in question design but less proficient with computers, or teachers who frequently work with large amounts of data and statistical analyzes.

5.1.3 User behavior. Participants' interactive behaviors during the creation of pedagogical questions, including the number of solution steps created, hints and scaffolding created, time spent, and the total word count of the questions, as well as the request made with LLM, were automatically recorded. As the data did not follow a normal distribution, a Friedman test (non-parametric repeated measures) was conducted to compare differences.

Table 2: The mean and standard deviation of subjective evaluation scores for three tools across 13 metrics.

item	TutorCraftEase		LLM Toolbox		Spreadsheet	
	Mean	SD	Mean	SD	Mean	SD
Helpfulness	5.56	1.715	4.58	1.556	3.17	1.978
Usability	5.81	1.470	4.89	1.469	3.03	1.699
Efficiency	5.78	1.606	4.61	1.793	2.75	1.730
Enjoyment	6.00	1.512	4.44	1.889	2.61	1.678
Mental Demand	5.17	1.813	4.03	1.732	3.03	2.021
Effort	5.33	1.927	3.89	1.785	2.53	1.715
Creative Achievement	5.14	1.823	4.50	1.699	3.75	1.857
Ownership	3.81	1.704	4.25	1.442	5.06	2.028
Quality	5.00	1.912	4.67	1.656	3.83	1.859
Controllability	5.17	1.715	5.14	1.376	4.94	1.851
Collaboration	5.56	1.576	5.11	1.369	4.06	1.970
Satisfaction	5.64	1.437	4.69	1.470	3.11	1.600
Recommendation	5.69	1.636	4.78	1.709	3.03	1.797

TutorCraftEase's integration of automated content generation and filling significantly streamlines the creation process, reducing the workload and saving time for creators. As shown in Table 4 and Table 5, creating a pedagogical question with TutorCraftEase ($M = 339.65s, SD = 235.36$) takes less time than using the LLM Toolbox ($M = 434.01s, SD = 366.49$) or the Spreadsheet ($M = 533.67s, SD = 827.67$). The Friedman test revealed significant differences in time, with post-hoc comparisons showing significant differences between TutorCraftEase and both LLM Toolbox and Spreadsheet. Additionally, TutorCraftEase generated more solution steps, hints, and scaffolding than the other two tools. The Friedman test also revealed significant differences in these aspects, with post-hoc comparisons showing no significant difference between LLM Toolbox and Spreadsheet. In terms of the number of characters, TutorCraftEase had more than LLM Toolbox but fewer than Spreadsheet. No significant differences were found in pairwise comparisons or among the three tools.

Although participants were instructed to include at least one hint and one scaffolding for each question, the results showed that 3.8% of TutorCraftEase-generated questions lacked a hint, and 4.4% lacked a scaffolding. For the LLM Toolbox, 12.6% lacked a hint, and 9.3% lacked a scaffolding. Similarly, 6.9% of Spreadsheet-generated questions lacked a hint, and 8.8% lacked a scaffolding. The lower rate of missing hints and scaffolding in TutorCraftEase is due to its approach of breaking down complex questions into smaller sub-questions and using specialized prompts to guide the LLM in generating necessary hints. Additionally, TutorCraftEase produced a wider variety of question types, such as multiple-choice, true/false, and fill-in-the-blank, while the LLM Toolbox and Spreadsheet primarily generated applied questions.

As shown in Table 6, the average request time for using the 'one-click generation' button in TutorCraftEase is 11.81 seconds, as it generates 3-5 solution steps at once, with an average of 2.4-4 seconds per step. This is similar to the time needed for modifying question attributes or generating solution steps using the LLM ToolBox. We also found that TutorCraftEase interacts less frequently with the LLM than the LLM ToolBox, but its request success rate is lower. This is influenced by the length of the prompt: shorter prompts result in faster LLM response times and higher success rates.

5.2 Qualitative results

In semi-structured interviews, 36 participants were asked to express their overall tool preferences and provide feedback on the functionality and user experience of each tool through open-ended questions. The open-ended questions were as follows:

- What are your thoughts on TutorCraftEase, including its strengths, weaknesses, and whether it meets your expectations?
- Which groups of teachers and specific use cases do you think TutorCraftEase is most suitable?
- What impact do you think LLMs will have on teachers?
- What changes do you think LLM could bring to education?
- What are your expectations for the future development of LLMs?

Data from interviews were collected by having participants respond to online open-ended questions. We conducted an inductive thematic analysis (TA) [15] to analyze the data. To ensure reliability and consistency, the coding team consisted of two authors. The process began by segmenting the raw data into smaller units and assigning keywords or phrases (codes) to capture their core meaning, along with the corresponding participant information. These codes were then grouped into subthemes based on frequency and relevance, which were subsequently synthesized into overarching themes. Finally, we performed a consistency analysis of the coder themes using the Cohen Kappa coefficient (κ =0.94).

5.2.1 *TutorCraftEase enhances efficiency, significantly reducing teachers' workload*.

Participants agreed with the motivation behind our design of TutorCraftEase and gave it high praise, believing that TutorCraftEase effectively alleviates both their mental and physical workload when creating pedagogical questions, thereby enhancing their creative efficiency. As described by some participants,

> *"TutorCraftEase can effectively reduce teachers' mental exertion and lighten their workload"* (P9). *"TutorCraftEase undoubtedly alleviates the burden on teachers both mentally and physically"* (P10). *"TutorCraftEase frees teachers from the heavy task of creating pedagogical questions"* (P19). *"TutorCraftEase meets my expectations of artificial intelligence; it can quickly and easily generate questions based on information, and all I need to do is review and make adjustments"* (P30).

Specifically, some participants shared comparative insights based on their experiences with the different orders in which the tools were used. For example,

> *"After enduring the tediousness of Spreadsheet, TutorCraftEase feels truly user-friendly and features a very comfortable interface"* (P5). *"I thought the LLM Toolbox had done a good job, but after using TutorCraftEase later, I suddenly felt that the LLM Toolbox's capabilities still need improvement"*(P14). *"Having used TutorCraftEase made me dissatisfied with the LLM Toolbox"*(P24).

Table 3: Friedman Test Results and Pairwise Comparisons for 13 Metrics

Metrics	Friedman Test (df = 2)		Pairwise Comparisons, Z-score (Bonferroni-adjusted p)		
	χ^2	p	TutorCraftEase vs LLM Toolbox	TutorCraftEase vs Spreadsheet	LLM Toolbox vs Spreadsheet
Helpfulness	34.373	.000	-2.475 (.040)	-5.303 (.000)	-2.828 (.014)
Efficiency	47.814	.000	-2.770 (.017)	-6.246 (.000)	-3.477 (.002)
Usability	39.22	.000	-1.768 (.231)	-5.657 (.000)	-3.889 (.000)
Enjoyment	41.48	.000	-3.064 (.007)	-5.951 (.000)	-2.887 (.012)
Effort	34.319	.000	-2.770 (.017)	-5.185 (.000)	-2.416 (.047)
Mental Demand	24.392	.000	-2.534 (.034)	-4.007(.000)	-1.473 (.422)
Creative Achievement	12.194	.002	-1.650 (.297)	-2.946 (.010)	-1.296 (.585)
Ownership	21.481	.000	1.120 (.789)	3.830 (.000)	2.711 (.020)
Quality	7.196	.027	-0.236 (.814)	-2.062 (.118)	-1.827 (.203)
Controllability*	0.066	.968	-	-	-
Collaboration	11.327	.003	-0.766 (.444)	-2.770 (.017)	-2.003 (.135)
Satisfaction	44.538	.000	-2.475 (.040)	-5.657 (.000)	-3.182 (.004)
Recommendation	37.922	.000	-1.768 (.231)	-5.127 (.000)	-3.359 (.002)

* For Controllability, since the overall test retained the null hypothesis of no difference, the Friedman test did not perform multiple comparisons. We conducted Wilcoxon signed-rank tests, and the results shown: z= -.082 (p=.934) between TutorCraftEase and LLM Toolbox, z= -.333 (p=.739) between TutorCraftEase and Spreadsheet, and z= -.532 (p=.595) between LLM Toolbox and Spreadsheet.

Table 4: Comparative analysis of question creation metrics across tools: a statistical overview of participant performance among three tools.

Metric	TutorCraftEase	LLM Toolbox	Spreadsheet
AVG Number of Solution Steps per question	2.60 (0.93)	2.01 (1.21)	2 (1.36)
AVG Number of Hints per question	2.87 (1.25)	2.40 (1.82)	2.32 (1.96)
AVG Number of Scaffolding per question	2.83 (1.18)	1.75 (1.14)	1.81 (1.57)
AVG Number of Chinese Characters per question	320.26 (190.72)	296.09 (219.841)	381.38 (359.22)
AVG Time to Create a question	339.65s (235.36)	434.01s (366.49)	533.67s (827.34)

Table 5: Friedman test results and pairwise comparisons for user behavior

Metrics	Friedman Test (df = 2)		Pairwise Comparisons, Z-score (Bonferroni-adjusted p)		
	χ^2	p	TutorCraftEase vs LLM Toolbox	TutorCraftEase vs Spreadsheet	LLM Toolbox vs Spreadsheet
Number of Solution steps	21.751	.000	-3.468 (.002)	-3.949 (.000)	-0.481 (.063)
Number of Hints	16.659	.000	-3.056 (.007)	-3.434 (.001)	-0.378 (.706)
Number of Scaffolding	45.886	.000	-5.323 (.000)	-5.597 (.000)	0.275 (.784)
Number of characters	3.608	.165	-	-	-
Time	19.774	.005	-1.578 (.045)	-2.152 (.012)	-1.039 (.135)

* For Number of characters, since the overall test retained the null hypothesis of no difference, the Friedman test did not perform multiple comparisons. We conducted Wilcoxon signed-rank tests, and the results shown: z= -1.500 (p=.063) between TutorCraftEase and LLM Toolbox, z= -0.048 (p=.468) between TutorCraftEase and Spreadsheet, and z= -1.092 (p=.125) between LLM Toolbox and Spreadsheet.

Additionally, participants believed that teachers who exhibit characteristics such as being 'novices', 'innovative', 'technologically inexperienced', or having 'tight work schedules' are most likely to use TutorCraftEase for creating pedagogical questions. According to some participants' views,

> *"TutorCraftEase is suitable for teachers of all subjects in elementary and secondary schools who are busy and need to spend a lot of time and energy creating questions to help students consolidate knowledge"* (P2). *"TutorCraftEase may be more suitable for teachers who enjoy innovation and novelty"* (P4). *"TutorCraftEase may be more suitable for young teachers when designing classroom interactive questions or stimulating the thinking of both teachers and students"* (P20).

A very small number of participants expressed uncertainty about whether TutorCraftEase can reduce teachers' mental workload. They thought that TutorCraftEase merely shifts teachers from being question creators to reviewers of questions generated by a large language model, requiring teachers to check and understand questions they not familiar with.

5.2.2 ***TutorCraftEase generates diverse pedagogical questions that meet teaching quality needs.*** Most participants

Table 6: LLM Requests Statistics for TutorCraftEase and LLM Toolbox, where successful TutorCraftEase requests denote requests yielding valid parsable JSON.

TutorCraftEase LLM Requests Statistics

Interactive Behavior	Successful Requests	Total Requests	Success Rate	AVG Response Time
Using 'One-Click generation' Button	215	237	90.72%	11.81s
Modifying the Question Attribute	126	139	90.65%	3.17s

LLM Toolbox Requests Statistics

Interactive Behavior	Successful Requests	Total Requests	Success Rate	AVG Request Time
Summarization	201	202	99.50%	0.95s
Question Generation	506	513	98.63%	1.03s
Use Custom Prompt	181	181	100%	2.69s

highly praised the logical framework (namely, title-body-solution steps) and quality of questions generated by TutorCraftEase, noting that the generated pedagogical questions closely align with the knowledge points, feature clear and concise solution steps, and cover a wide variety of question types. Some participants noted,

> *"TutorCraftEase accurately creates questions based on the input knowledge points, and the generated questions are free of factual errors, scientifically sound, and reasonable"* (P31). *"TutorCraftEase has strong scalability, and the generated pedagogical questions are closely aligned with the knowledge points"* (P8). *"In interactive editing, when creating pedagogical questions with the same attributes twice, the results differ each time, highlighting the diversity in question generation, which is commendable"*(P18). *"The generated questions mostly meet expectations, providing multiple related questions of varying difficulty levels, suitable for different grade levels"* (P22).

There are also a minority of participants (n=9) who felt that the quality of the questions generated by TutorCraftEase did not meet their expectations. The main reasons included:

- Inconsistent question quality. The generated questions sometimes exceeded the intended scope, contained factual inaccuracies, and performed poorly in mathematical and logical reasoning. Issues raised included, *"the extraction of images and contextual content is incomplete, making it difficult to generate high-quality questions"* (P28), *"the difficulty of the questions is unusual for typical exam questions"* (P34), and *"TutorCraftEase struggles with tasks that require deep analytical thinking and reasoning, such as those found in mathematics and physics"* (P15).
- Incomplete consideration of question complexity. The depth and difficulty of the generated questions were often deemed inadequate, with participants commenting, *"the questions lack sufficient complexity and cover too few concepts"* (P24), and *"generated questions do not consider the complexity of calculations, such as adjusting gravitational values to multiples of 9.8 to simplify calculations for students"* (P11).
- Limited question types. The pedagogical questions were text-based and did not support complex information and contextual semantics. This limitation is noted in remarks such as, *"the tool does not support complex tables or customized templates"* (P1), and *"the questions generated were described as simple, single-topic and lacking context"* (P35).

5.2.3 *TutorCraftEase fosters human-LLM collaboration but may effect teachers' creative autonomy.*

Participants praised TutorCraftEase's highly user-friendly design, highlighting that it not only simplifies the process of creating pedagogical questions but also provides options for interactive editing and question regeneration. Additionally, it retains the functionality for manual editing of pedagogical questions, offering users greater flexibility. As emphasized by some participants,

> *"TutorCraftEase enables collaboration between teachers and AI, improving the efficiency and quality of creating pedagogical questions, and being able to become a valuable assistant for teacher"* (P20). *"Through interactive editing, TutorCraftEase facilitates effective collaboration between teachers and LLMs in generating questions, allowing for customization of question types, difficulty levels, and knowledge points"* (P31). *"For me, TutorCraftEase is easy to edit. Besides interactive editing, users can also add relevant hints for the question-solving section, among other features"* (P10).

However, participants believed that the LLM Toolbox and spreadsheets were more effective in helping them create original pedagogical questions, while TutorCraftEase somewhat restricted their creative autonomy. Although both TutorCraftEase and the LLM Toolbox rely on LLM assistance, the LLM Toolbox is perceived to offer greater autonomy. This is because the final creation of pedagogical questions in the LLM Toolbox still requires participants to manually filter and refine the LLM-generated outputs, with the LLM's role limited to extracting knowledge and gathering information. Interestingly, the participants seem to have overlooked the fact that TutorCraftEase also supports manual adjustments to pedagogical questions. As pointed out by some participants,

> *"TutorCraftEase is simple and efficient to operate, but the autonomy is relatively weaker"* (P23). *"If time were not so pressing, I might not prioritize the TutorCraftEase, as it always feels like it's not my own creation"* (P25).

> *"When using Spreadsheets, I feel that the control and initiative over knowledge are still in my hands"* (P5). *"LLM Toolbox merely provides a more flexible template for generating questions, without limiting teachers' ability to choose other options"*(P2).

As a result, some participants expressed a desire for TutorCraftEase to include a feature for manually designing prompts, similar to the LLM Toolbox, in order to enhance their creative autonomy during the creation process. For example, P4 stated *"The LLM Toolbox allows me to personalize the prompts and gradually generate information about the questions"*. However, most of the participants expressed concerns about the design of the prompts themselves, noting *"I do not even know how to design the output format of a prompt"* (P31).

5.2.4 ***LLM-based tools like TutorCraftEase drive educational transformation and have the potential to reshape teaching models.*** Participants believe that LLMs can offer teachers a wealth of resources, broaden their perspectives on teaching, and enhance their skills in instructional design and innovation, especially for young teachers. Furthermore, participants felt that the pedagogical questions generated by LLM could effectively stimulate the critical thinking of students, as answers to these questions are not readily available online. As mentioned by some participants,

> *"LLMs, with their powerful resource integration capabilities, can provide teachers with abundant teaching materials and case studies, helping to broaden their instructional perspectives"* (P17). *"Teachers can also use LLM-based technology to quickly generate diverse instructional design plans, effectively reducing the challenges of lesson preparation caused by resource shortages and content complexity"* (P21). *"In the long term, large language model technology not only has the potential to significantly enhance teaching quality, but also provides scientific support for decision making, aiding educators in their professional growth and skill development"* (P12).

Participants also believe that LLM-based technology allows teachers to manage the pedagogical process more effectively. This is primarily because LLMs can significantly reduce teachers' workload, especially in areas such as creating pedagogical questions, enabling them to concentrate more on overall pedagogical development and deliver more precise instruction. At the same time, participants unanimously agree that the role of LLMs in education is largely supportive rather than substitutive. However, they also emphasize that the implementation of this technology will impose new demands on teachers' skills and professional roles. For example, a participant mentioned

> *"As LLMs-assisted technology continues to mature, it will penetrate the education field and gradually integrate into classrooms. What will truly be phased out are those teachers who are unwilling or unable to adopt new tools. Therefore, it is crucial for teachers to proactively adapt to these trends, engage in continuous learning, and master new tools"* (P35).

In addition, participants believe that LLM-based tools and technologies have a positive impact on education and can bring about profound changes in the field. The application of this technology helps optimize the allocation of educational resources, promotes interdisciplinary integration, and advances educational equity and high-quality development, ensuring that the benefits of technological progress reach every student and teacher. As some participants stated,

> *"The impact of LLM-assisted technology on the education field is profound and multifaceted. Its powerful data processing capabilities, intelligent analytical functions, and highly customizable features have brought about revolutionary changes in the education field"* (P21). *"LLMs-assisted technology will accelerate education's transition into the new era of 'artificial intelligence + education', providing more precise and abundant teaching resources while driving the comprehensive development of the field"* (P2). *"By integrating knowledge from diverse disciplines, LLMs can foster interdisciplinary collaboration, offering students a more comprehensive and multidimensional educational experience"* (P1). *"Furthermore, the application of LLMs helps overcome geographic and economic barriers to educational resources, enabling students in remote areas to access high-quality educational materials and services"* (P28).

Beyond the benefits mentioned above, the participants also expressed concerns about LLM-based technology, warning against the potential *"dangers of AI/LLM"* (P15). For example, some participants worry that the steep learning curve associated with new technologies, along with an overreliance on LLMs, could negatively impact pedagogical methods and diminish teaching quality. In addition, this technology could exacerbate disparities in teachers' abilities, be misused as a tool for student cheating, or even lead to a reduction in employment opportunities, with some participants expressing concerns about potential job displacement.

6 Discussion and implication

6.1 Effective transformation of LLM outputs into pedagogical questions

The mechanisms by which LLMs analyze subtle differences in prompts remain largely a 'black box' [115, 117]. Even when provided with identical prompts, LLM outputs occasionally deviate from expected requirements. Thus, designing effective prompts and transforming LLM outputs into content that meets specific needs has become a core challenge in developing LLM-based tools. For instance, TutorCraftEase aims to generate pedagogical questions with a 'title-body-solution steps' framework based on user-selected content from reference panels. In earlier attempts, we explored specific models like T5 [84] and BART [51] to generate question-answer pairs or to summarize titles and hints for pedagogical questions. However, these approaches fell short in addressing the nuanced requirements of crafting questions. For example, the t5-base-qa-qg-hl model [79] relies heavily on extracting information from input text, lacks the creativity needed to generate complex application or computation problems, and struggles to process lengthy inputs. In contrast, GPT-4o overcomes these limitations, demonstrating the ability to process long texts and generate diverse, complex teaching problems, providing robust support for this task.

To generate high-quality pedagogical questions using GPT-4o, we identified the core elements required for various problem types and implemented techniques such as the RICTEF prompt template, chained prompt strategy, and tree-based decomposition method. These approaches reduce issues of excessive generalization or divergence in generated questions while clearly defining output requirements. However, prompt engineering alone cannot completely eliminate the instability of generated questions, a limitation consistent with existing research findings on the variability of LLM outputs under fixed prompts [115]. Additionally, we observed that LLMs, in rare cases, may over-rely on template examples, which can constrain content diversity. To mitigate this, we suggest using dynamic templates and random example selection strategies to encourage more varied and personalized output.

Implication: The variability of LLM outputs and their 'black-box' nature indicate that prompt design and its impact on output results are highly complex. For general-purpose LLMs, the key to transforming outputs into content that meets specific domain requirements lies in the construction of precise, well-structured prompt templates. These templates must establish a clear relationship between the input data, the model's processing method, and the output. In addition, it is essential to incorporate relevant domain-specific constraints (such as the requirements for various types of pedagogical question) to ensure the quality and stability of the output conversion. To balance stability and diversity in the generated content, future LLM-based educational tools may need to integrate more flexible, context-sensitive mechanisms to accommodate evolving input and output requirements. Meanwhile, post-processing techniques (such as secondary accuracy checks) can help ensure the stability of the output.

6.2 Balancing AI assistance and user autonomy in pedagogical questions creation

LLMs have greatly enhanced the efficiency of creating pedagogical questions while significantly reducing teachers' workloads. However, we found that participants often feel that these tools limit their direct control over summarizing knowledge points and creating questions, thereby hindering their ability to fully exercise autonomy. Although teachers can modify or regenerate questions through interactive editing, these options do not fully alleviate their concerns. Furthermore, we observed that participants frequently overlook the fact that they can manually adjust the pedagogical questions generated by LLMs during use. This issue is closely linked to their self-perception as assessors or users rather than collaborators in the creative process, which deepens their reliance on the creation tools and further diminishes their sense of autonomy.

In the process of creating pedagogical questions, teachers' reliance manifests in dependence on both the LLM assistance (creative tools) and overly rigid solutions to questions. The impact of reliance on LLM assistance is relatively minor, as the questions generated by LLM function similarly to traditional purchased question sets or exam papers, while also encouraging teachers to actively check and verify the correctness of the questions. However, dependence on AI-generated overly rigid solutions may have a more profound negative impact on the pedagogical process. This reliance may lead to a more uniform teaching approach, reducing both the flexibility and diversity of instructional methods, thereby limiting students' ability to innovate.

Implication: As stated by Passi and Vorvoreanu [78], offering personalized adjustments, real-time feedback, and modified interaction strategies can enhance user autonomy and reduce overreliance on LLM assistance (e.g., modifying question attributes and making manual adjustments). Additionally, gradual guidance or offering diverse content creation options, such as providing multiple solution steps or guiding users to add hints or scaffolding, can further prevent dependency. Beyond reliance on LLM assistance, our primary concern is users' overreliance on AI-generated overly rigid solutions (e.g., question-solving steps). To address this, we recommend providing users with multiple alternative solutions and introducing a reflection mechanism (such as annotations and comments on generated content) to encourage critical thinking and reduce dependency.

6.3 Teacher-centered interactive design for LLM-based pedagogical support tools

For teachers of non-computer science, directly using LLM or designing appropriate prompts for it is a highly challenging task[111]. To address this, TutorCraftEase simplifies the interaction process with the LLMs by only requiring teachers to provide the reference materials for the pedagogical questions or modify the attributes of the generated questions that need to be regenerated. However, apart from modifying the attributes to regenerate new questions, TutorCraftEase does not provide corresponding interaction technologies for the details of the pedagogical questions (such as undo, redo, etc.), and only allows teachers to manually modify those questions. This somewhat limits the flexibility and ease of operation for teachers during the question-editing process. Moreover, TutorCraftEase's generation of large amounts of content (for example, generating multiple pedagogical questions at once, each with more than 500 words) results in longer generation times, which affects the user experience. To improve this, progress feedback for each question could be provided in real-time, or even elements within a single question, based on the 'title-question-solution steps' framework, could be displayed gradually to alleviate the user's demand for real-time responsiveness.

Furthermore, tools like TutorCraftEase face the complexity of personalization because their users and content recipients are not in the same group. The tool must meet the personalized needs of creators in content creation, while also accommodating the personalized experiences of the recipients. However, creators often struggle to obtain specific information about the recipients, which limits the design space for interaction technologies in content creation and affects the creator's freedom during the creative process. Additionally, we have found that custom prompts can effectively support the generation of personalized content (as reflected in feedback from some users of the LLM Toolbox), but helping teachers who are unfamiliar with prompt design to quickly and accurately customize prompts remains a challenge that needs to be addressed in future work.

Implication: Providing intuitive interfaces and interaction methods is essential to facilitate collaboration between humans and LLMs, especially for users unfamiliar with LLMs[47]. This includes showcasing the LLMs' strengths, such as their ability to quickly summarize and generate content in pedagogical questions creating tasks, and providing appropriate intervention strategies at key stages to reduce the learning curve [95] for new technologies, enable personalized changes (e.g., prompts and content), and support flexible review. Furthermore, generating multiple pieces of content at once may reduce response time expectations, so displaying content (e.g., elements within content) sequentially can help alleviate pressure. Another approach is to simplify prompts and limit the output size of LLM to improve the LLM's understanding. Equally important is enabling creators to access audience-specific information. Future LLM-based pedagogical support tools could collect or integrate user and contextual data (e.g., students' responses to pedagogical questions), thereby facilitating more effective content customization while preserving creative freedom.

6.4 The impact of LLM-based technology on teachers' role and educational development

As educational methods continue to evolve—from manual crafting of pedagogical content to the use of productivity tools [75], and now to the widespread adoption of LLM-based technologies and the emerging potential of AI-generated content (AIGC), future development of education prompts deep reflection[21, 49]. In this process, improving teaching quality and advancing the education sector remain central goals. These advancements are evident in various ways, such as the innovation of supplementary teaching methods, the realization of personalized learning, and the removal of geographic barriers to broaden access to education. However, as technology progresses and educational models transform, the demands on teachers increase. Teachers must not only master new technologies, but also reconsider their roles in this evolving educational landscape. Despite these changes, the core of education remains unchanged: the profound communication and emotional resonance between teachers and students continue to be the essence of teaching. Although LLM-based technologies and AI can offer powerful support, they cannot replace the guiding and inspiring roles of teachers in the educational process.

While acknowledging the positive impact of LLM-based technologies on the development of education, it remains essential to critically evaluate their actual capabilities in terms of accuracy, diversity, and ability to meet personalized teaching needs[1]. Some educators argue that tools such as TutorCraftEase are gradually shifting the focus of teaching quality from the teacher's expertise to the quality of content generated by AI or LLMs. This shift raises important questions about how to measure the participation of teachers and LLMs in the teaching process and how this involvement relates to teaching quality. Furthermore, this change in the degree of participation could exacerbate disparities in teacher capabilities, potentially undermining the effectiveness of teaching.

Moreover, compared to applications in healthcare and transportation that require zero tolerance for errors [22, 85], the education field demonstrates a greater tolerance for inaccuracies in generated content. Educators may even welcome such errors to some extent, viewing them as opportunities to stimulate student thinking and assess their understanding of knowledge. This tolerant attitude underscores the importance of fostering critical thinking and independent learning skills, further emphasizing the supportive role of LLM-based technologies in education.

Implication: With the introduction of LLMs in education, the role of teachers may shift from being knowledge transmitters to more active roles as knowledge reviewers, motivators of student learning and emotional supporters. In response to this shift, the design of pedagogical support tools should prioritize fostering efficient collaboration between teachers and AI, with a clear delineation of their respective roles. For example, teachers should retain responsibility for classroom management and personalized guidance, while AI focuses on tasks such as automated content generation and data analysis. Future research should also examine the specific impact of varying levels of collaboration between teachers and AI on teaching quality, providing insights to optimize the design and application of teaching tools. AI-based tools should be viewed as partners in education, not just as impersonal machines. Therefore, these tools should be designed with a teacher- and student-centered approach, prioritizing on emotional and personalized support while allowing for minor non-common-sense errors.

7 Limitations and Future Work

We did not fully adopt validated tools such as SUS or NASA-TLX [2, 10] for user experience evaluation. Instead, we selected metrics from them and supplemented them with assessments of collaboration and autonomy demonstrated by TutorCraftEase during the creation of pedagogical questions. Future research will employ a mixed-methods approach, combining standardized tools with self-designed questionnaires to improve measurement accuracy and generalizability. Although statistical analysis shows that TutorCraftEase generates pedagogical questions of similar quality to those created by experienced teachers, the current process does not fully account for students' abilities. We plan to adapt the tool to better align with students' learning needs and test it in real-world teaching contexts.

We also found that LLM's generative capabilities may affect teachers' autonomy in question creation. Although our interaction methods can partially mitigate this, further exploration of the influencing factors is needed. We also discovered that the dependency on the overly rigid solution generated by AI in question solving could influence the teachers' pedagogical thinking. Future research should examine this further and propose appropriate solutions. Furthermore, although we have compared the effects of not using, selectively using, and fully relying on the capabilities of LLM to create pedagogical questions, the methods and extent of LLM integration in education will have broad implications for teachers and teaching practices. Future studies should extend this discussion to other areas, such as lesson planning, student learning assessments, and more.

Finally, while this study focuses on TutorCraftEase's performance in generating teaching questions for OATutor, we have not yet examined its application within OATutor or other intelligent tutoring systems. Future work will explore its role in the ITS content creation ecosystem and conduct comparative studies to assess its effectiveness and usability.

8 Conclusion

In this work, we introduced TutorCraftEase, an innovative interactive tool designed to help authors simplify the creation of complex pedagogical questions. Comparative studies with traditional spreadsheet-based authoring methods and basic LLM support tools show that the pedagogical questions generated by TutorCraftEase are comparable in quality to those created by experienced teachers, while significantly reducing the time and effort required. Meanwhile, most of the participants expressed a strong preference for TutorCraftEase and were willing to recommend it to others.

Acknowledgments

We thank anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Natural Science Foundation of China (Grant no. 62332017, 62172397), the Key Project of the Institute of Software Chinese Academy of Sciences (Grant no. ISCAS-ZD-202401) and the Innovation Team 2024 ISCAS (Grant no. 2024-66).

References

[1] Alaa Abd-Alrazaq, Rawan AlSaad, Dari Alhuwail, Arfan Ahmed, Padraig Mark Healy, Syed Latifi, Sarah Aziz, Rafat Damseh, Sadam Alabed Alrazak, Javaid Sheikh, et al. 2023. Large language models in medical education: opportunities, challenges, and future directions. *JMIR Medical Education* 9, 1 (2023), e48291. https://doi.org/10.2196/48291

[2] Naser Al Madi, Siyuan Peng, and Tamsin Rogers. 2022. Assessing Workload Perception in Introductory Computer Science Projects using NASA-TLX. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1* (Providence, RI, USA) *(SIGCSE 2022)*. Association for Computing Machinery, New York, NY, USA, 668–674. https://doi.org/10.1145/3478431.3499406

[3] Vincent Aleven, Bruce M. McLaren, Jonathan Sewall, and Kenneth R. Koedinger. 2006. The Cognitive Tutor Authoring Tools (CTAT): Preliminary Evaluation of Efficiency Gains. In *Intelligent Tutoring Systems (Lecture Notes in Computer Science)*, Mitsuru Ikeda, Kevin D. Ashley, and Tak-Wai Chan (Eds.). Springer, Berlin, Heidelberg, 61–70. https://doi.org/10.1007/11774303_7

[4] Vincent Aleven, Bruce M. McLaren, Jonathan Sewall, Martin Van Velsen, Octav Popescu, Sandy Demi, Michael A. Ringenberg, and K. Koedinger. 2016. Example-Tracing Tutors: Intelligent Tutor Development for Non-programmers. *International Journal of Artificial Intelligence in Education* 26 (2016), 224 – 269. https://doi.org/10.1007/s40593-015-0088-2

[5] Bashaer Alsafari, Eric Atwell, Aisha Walker, and Martin Callaghan. 2024. Towards effective teaching assistants: From intent-based chatbots to LLM-powered teaching assistants. *Natural Language Processing Journal* (2024), 100101. https://doi.org/10.1016/j.nlp.2024.100101

[6] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, and Eric Horvitz. 2019. Guidelines for Human-AI Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300233

[7] John Anderson, Albert Corbett, Kenneth Koedinger, and Ray Pelletier. 1995. Cognitive Tutors: Lessons Learned. *Journal of the Learning Sciences* 4 (04 1995), 167–207. https://doi.org/10.1207/s15327809jls0402_2

[8] John Annett. 2003. Hierarchical task analysis. In *Handbook of cognitive task design*. CRC Press, 17–36.

[9] Igor Bascandziev, Patrick Shafto, and Elizabeth Bonawitz. 2021. The sound of pedagogical questions. In *Proceedings of the annual meeting of the cognitive science society*, Vol. 43.

[10] Juergen Baumgartner, Naomi Frei, Mascha Kleinke, Juergen Sauer, and Andreas Sonderegger. 2019. Pictorial System Usability Scale (P-SUS): Developing an Instrument for Measuring Perceived Usability. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3290605.3300299

[11] Seth Bernstein, Paul Denny, Juho Leinonen, Matt Littlefield, Arto Hellas, and Stephen MacNeil. 2024. Analyzing Students' Preferences for LLM-Generated Analogies. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 2* (Milan, Italy) *(ITiCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 812. https://doi.org/10.1145/3649405.3659504

[12] Sue Black. 2001. Ask Me a Question: How Teachers Use Inquiry in a Classroom. *The American school board journal* 188 (2001), 43–45. https://api.semanticscholar.org/CorpusID:150493445

[13] Matthew L Bolton, Elliot Biltekoff, and Laura Humphrey. 2023. The mathematical meaninglessness of the NASA task load index: A level of measurement analysis. *IEEE Transactions on Human-Machine Systems* 53, 3 (2023), 590–599.

[14] Stephen Brade, Bryan Wang, Mauricio Sousa, Sageev Oore, and Tovi Grossman. 2023. Promptify: Text-to-Image Generation through Interactive Prompt Exploration with Large Language Models. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3586183.3606725

[15] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative research in psychology* 3, 2 (2006), 77–101.

[16] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. https://doi.org/10.48550/arXiv.2005.14165 arXiv:2005.14165 [cs].

[17] William Cain. 2024. Prompting change: exploring prompt engineering in large language model AI and its potential to transform education. *TechTrends* 68, 1 (2024), 47–57.

[18] Tommaso Calo and Christopher Maclellan. 2024. Towards Educator-Driven Tutor Authoring: Generative AI Approaches for Creating Intelligent Tutor Interfaces. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale* (Atlanta, GA, USA) *(L@S '24)*. Association for Computing Machinery, New York, NY, USA, 305–309. https://doi.org/10.1145/3657604.3664694

[19] Matúš Čavojský, Gabriel Bugár, Tomáš Kormaník, and Martin Hasin. 2023. Exploring the Capabilities and Possible Applications of Large Language Models for Education. In *2023 21st International Conference on Emerging eLearning Technologies and Applications (ICETA)*. IEEE, 91–98.

[20] Balakrishnan Chandrasekaran. 1990. Design problem solving: A task analysis. *AI magazine* 11, 4 (1990), 59–59.

[21] Xiaojiao Chen, Zhebing Hu, and Chengliang Wang. 2024. Empowering education development through AIGC: A systematic literature review. *Education and Information Technologies* (2024), 1–53.

[22] Michelle Chua, Doyun Kim, Jongmun Choi, Nahyoung G Lee, Vikram Deshpande, Joseph Schwab, Michael H Lev, Ramon G Gonzalez, Michael S Gee, and Synho Do. 2023. Tackling prediction uncertainty in machine learning for healthcare. *Nature Biomedical Engineering* 7, 6 (2023), 711–718.

[23] Adam Coscia and Alex Endert. 2023. Knowledgevis: Interpreting language models by comparing fill-in-the-blank prompts. *IEEE Transactions on Visualization and Computer Graphics* (2023).

[24] Hai Dang, Karim Benharrak, Florian Lehmann, and Daniel Buschek. 2022. Beyond Text Generation: Supporting Writers with Continuous Automatic Text Summaries. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3526113.3545672

[25] Emily N Daubert, Yue Yu, Milagros Grados, Patrick Shafto, and Elizabeth Bonawitz. 2020. Pedagogical questions promote causal learning in preschoolers. *Scientific reports* 10, 1 (2020), 20700.

[26] Sheila Degotardi, Jane Torr, and Feifei Han. 2018. Infant educators' use of pedagogical questioning: Relationships with the context of interaction and educators' qualifications. *Early Education and Development* 29, 8 (2018), 1004–1018.

[27] Yang Deng, An Zhang, Yankai Lin, Xu Chen, Ji-Rong Wen, and Tat-Seng Chua. 2024. Large Language Model Powered Agents in the Web. In *Companion Proceedings of the ACM Web Conference 2024* (Singapore, Singapore) *(WWW '24)*. Association for Computing Machinery, New York, NY, USA, 1242–1245. https://doi.org/10.1145/3589335.3641240

[28] Diego Dermeval, Italo Lima, Matheus Castro, Henrique Couto, Daniel Gomes, Aristoteles Peixoto, and Ig Ibert Bittencourt. 2019. Helping Teachers Design Gamified Intelligent Tutoring Systems. *2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)* 2161-377X (2019), 60–62. https://api.semanticscholar.org/CorpusID:201833281

[29] Federico Farini and Angela Scollan. 2021. Meanings and methods of pedagogical innovation. 84450.

[30] Shi Feng, Alejandra J. Magana, and Dominic Kao. 2021. A Systematic Review of Literature on the Effectiveness of Intelligent Tutoring Systems in STEM. In *2021 IEEE Frontiers in Education Conference (FIE)*. 1–9. https://doi.org/10.1109/FIE49875.2021.9637240

[31] Sylvain Fleury and Noémie Chaniaud. 2024. Multi-user centered design: acceptance, user experience, user research and user testing. *Theoretical Issues in Ergonomics Science* 25, 2 (2024), 209–224.

[32] Luis Francisco and Srini Arikati. 2024. LLM Based Physical Verification Runset Generator. In *Proceedings of the 2024 ACM/IEEE International Symposium on Machine Learning for CAD* (Salt Lake City, UT, USA) *(MLCAD '24)*. Association for Computing Machinery, New York, NY, USA, Article 35, 7 pages. https://doi.org/10.1145/3670474.3685976

[33] Meredith Damien Gall. 1970. The Use of Questions in Teaching. *Review of Educational Research* 40 (1970), 707 – 721. https://api.semanticscholar.org/CorpusID:67793244

[34] Hannah Gonzalez, Liam Dugan, Eleni Miltsakaki, Zhiqi Cui, Jiaxuan Ren, Bryan Li, Shriyash Upadhyay, Etan Ginsberg, and Chris Callison-Burch. 2023. Enhancing Human Summaries for Question-Answer Generation in Education. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*. Association for Computational Linguistics, Toronto, Canada, 108–118. https://doi.org/10.18653/v1/2023.bea-1.9

[35] Virginia Grande, Natalie Kiesler, and María Andreína Francisco R. 2024. Student Perspectives on Using a Large Language Model (LLM) for an Assignment on Professional Ethics. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1* (Milan, Italy) *(ITiCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 478–484. https://doi.org/10.1145/3649217.3653624

[36] Michael A. Hedderich, Natalie N. Bazarova, Wenting Zou, Ryun Shim, Xinda Ma, and Qian Yang. 2024. A Piece of Theatre: Investigating How Teachers Design LLM Chatbots to Assist Adolescent Cyberbullying Education. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 668, 17 pages. https://doi.org/10.1145/3613904.3642379

[37] Xinying Hou, Zihan Wu, Xu Wang, and Barbara J. Ericson. 2024. CodeTailor: LLM-Powered Personalized Parsons Puzzles for Engaging Support While Learning Programming. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale* (Atlanta, GA, USA) *(L@S '24)*. Association for Computing Machinery, New York, NY, USA, 51–62. https://doi.org/10.1145/3657604.3662032

[38] Lina Fouad Jawad, Ban Hassan Majeed, and Haider Th Salim ALRikabi. 2021. The Impact of Teaching by Using STEM Approach in The Development of Creative Thinking and Mathematical Achievement Among the Students of The Fourth Scientific Class. *International Journal of Interactive Mobile Technologies* 15, 13 (2021).

[39] Breanna Jury, Angela Lorusso, Juho Leinonen, Paul Denny, and Andrew Luxton-Reilly. 2024. Evaluating LLM-generated Worked Examples in an Introductory Programming Course. In *Proceedings of the 26th Australasian Computing Education Conference* (Sydney, NSW, Australia) *(ACE '24)*. Association for Computing Machinery, New York, NY, USA, 77–86. https://doi.org/10.1145/3636243.3636252

[40] Wenhui Kang, Jin Huang, Feng Tian, Xiangmin Fan, Jie Liu, and Guozhong Dai. 2021. Human-in-the-Loop Based Online Handwriting Mathematical Expressions Recognition. *Journal of Computer-Aided Design & Computer Graphics* (2021). https://api.semanticscholar.org/CorpusID:245959429

[41] Wenhui Kang, Jin Huang, Qingshan Tong, Qiang Fu, Feng Tian, and Guozhong Dai. 2024. MathAssist: A Handwritten Mathematical Expression Autocomplete Technique. In *Proceedings of the 29th International Conference on Intelligent User Interfaces* (Greenville, SC, USA) *(IUI '24)*. Association for Computing Machinery, New York, NY, USA, 566–581. https://doi.org/10.1145/3640543.3645149

[42] Wenhui Kang, Jin Huang, Qingshan Tong, Qiang Fu, Feng Tian, and Guozhong Dai. 2025. HchMER: A Handwritten Mathematical Expression Recognition Method Hybridized Human-machine Intelligence. *Ruan Jian Xue Bao/Journal of Software* 36, 2 (02 2025), 915. https://doi.org/10.13328/j.cnki.jos.007169

[43] Majeed Kazemitabaar, Runlong Ye, Xiaoning Wang, Austin Zachary Henley, Paul Denny, Michelle Craig, and Tovi Grossman. 2024. CodeAid: Evaluating a Classroom Deployment of an LLM-based Programming Assistant that Balances Student and Educator Needs. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 650, 20 pages. https://doi.org/10.1145/3613904.3642773

[44] Jinhee Kim, Hyunkyung Lee, and Young Hoan Cho. 2022. Learning design to support student-AI collaboration: Perspectives of leading teachers for AI in education. *Education and Information Technologies* 27, 5 (2022), 6069–6104.

[45] Tae Soo Kim, Yoonjoo Lee, Jamin Shin, Young-Ho Kim, and Juho Kim. 2024. EvalLM: Interactive Evaluation of Large Language Model Prompts on User-Defined Criteria. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 306, 21 pages. https://doi.org/10.1145/3613904.3642216

[46] Vassilka D. Kirova, Cyril S. Ku, Joseph R. Laracy, and Thomas J. Marlowe. 2024. Software Engineering Education Must Adapt and Evolve for an LLM Environment. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 666–672. https://doi.org/10.1145/3626252.3630927

[47] Harsh Kumar, Ilya Musabirov, Mohi Reza, Jiakai Shi, Xinyuan Wang, Joseph Jay Williams, Anastasia Kuzminykh, and Michael Liut. 2023. Impact of guidance and interaction strategies for LLM use on Learner Performance and perception. *arXiv preprint arXiv:2310.13712* (2023).

[48] Vivian Lai, Chacha Chen, Alison Smith-Renner, Q Vera Liao, and Chenhao Tan. 2023. Towards a science of human-AI decision making: An overview of design space in empirical human-subject studies. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. 1369–1385.

[49] Jinsook Lee, Yann Hicke, Renzhe Yu, Christopher Brooks, and René F Kizilcec. 2024. The life cycle of large language models in education: A framework for understanding sources of bias. *British Journal of Educational Technology* 55, 5 (2024), 1982–2002.

[50] Christopher Davin Leoputra, Dicky Prima Satya, and Muhammad Romadhon Al-Ghazali. 2023. Application of User-Centered Design Approach in Developing Interaction Design of In-Kind Donation Feature on a Crowdfunding Platform. In *2023 International Conference on Electrical Engineering and Informatics (ICEEI)*. IEEE, 1–6.

[51] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

[52] Jiayang Li, Jiale Li, and Yunsheng Su. 2024. A Map of Exploring Human Interaction Patterns with LLM: Insights into Collaboration and Creativity. In *International Conference on Human-Computer Interaction*. Springer, 60–85.

[53] Lei Li, Yongfeng Zhang, and Li Chen. 2023. Prompt Distillation for Efficient LLM-based Recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (Birmingham, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 1348–1357. https://doi.org/10.1145/3583780.3615017

[54] Q. Vera Liao, Daniel Gruen, and Sarah Miller. 2020. Questioning the AI: Informing Design Practices for Explainable AI User Experiences. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3313831.3376590

[55] Bang Liu, Haojie Wei, Di Niu, Haolan Chen, and Yancheng He. 2020. Asking Questions the Human Way: Scalable Question-Answer Generation from Text Corpus. In *Proceedings of The Web Conference 2020* (Taipei, Taiwan) *(WWW '20)*. Association for Computing Machinery, New York, NY, USA, 2032–2043. https://doi.org/10.1145/3366423.3380270

[56] Yiren Liu, Si Chen, Haocong Cheng, Mengxia Yu, Xiao Ran, Andrew Mo, Yiliu Tang, and Yun Huang. 2024. How AI Processing Delays Foster Creativity: Exploring Research Question Co-Creation with an LLM-based Agent. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 17, 25 pages. https://doi.org/10.1145/3613904.3642698

[57] Zhe Liu, Chunyang Chen, Junjie Wang, Xing Che, Yuekai Huang, Jun Hu, and Qing Wang. 2023. Fill in the Blank: Context-Aware Automated Text Input Generation for Mobile GUI Testing. In *Proceedings of the 45th International Conference on Software Engineering* (Melbourne, Victoria, Australia) *(ICSE '23)*. IEEE Press, 1355–1367. https://doi.org/10.1109/ICSE48619.2023.00119

[58] Leo S Lo. 2023. The CLEAR path: A framework for enhancing information literacy through prompt engineering. *The Journal of Academic Librarianship* 49, 4 (2023), 102720.

[59] Ryan Louie, Andy Coenen, Cheng Zhi Huang, Michael Terry, and Carrie J. Cai. 2020. Novice-AI Music Co-Creation via AI-Steering Tools for Deep Generative Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376739

[60] Xinyi Lu, Simin Fan, Jessica Houghton, Lu Wang, and Xu Wang. 2023. ReadingQuizMaker: A Human-NLP Collaborative System that Supports Instructors to Design High-Quality Reading Quiz Questions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–18. https://doi.org/10.1145/3544548.3580957

[61] Wenhan Lyu, Yimeng Wang, Tingting (Rachel) Chung, Yifan Sun, and Yixuan Zhang. 2024. Evaluating the Effectiveness of LLMs in Introductory Computer Science Education: A Semester-Long Field Study. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale* (Atlanta, GA, USA) *(L@S '24)*. Association for Computing Machinery, New York, NY, USA, 63–74. https://doi.org/10.1145/3657604.3662036

[62] Wenting Ma, Olusola O. Adesope, John C. Nesbit, and Qing Liu. 2014. Intelligent tutoring systems and learning outcomes: A meta-analysis. *Journal of Educational Psychology* 106, 4 (2014), 901–918. https://doi.org/10.1037/a0037123 Place: US Publisher: American Psychological Association.

[63] Christopher J. MacLellan and Kenneth R. Koedinger. 2022. Domain-General Tutor Authoring with Apprentice Learner Models. *International Journal of Artificial Intelligence in Education* 32, 1 (March 2022), 76–117. https://doi.org/10.1007/s40593-020-00214-2

[64] Noboru Matsuda, William W. Cohen, and K. Koedinger. 2014. Teaching the Teacher: Tutoring SimStudent Leads to More Effective Cognitive Tutor Authoring. *International Journal of Artificial Intelligence in Education* 25 (2014), 1 – 34. https://api.semanticscholar.org/CorpusID:18202628

[65] Hunter McNichols, Mengxue Zhang, and Andrew Lan. 2023. Algebra Error Classification with Large Language Models. In *Artificial Intelligence in Education*, Ning Wang, Genaro Rebolledo-Mendez, Noboru Matsuda, Olga C. Santos, and Vania Dimitrova (Eds.). Springer Nature Switzerland, Cham, 365–376.

[66] Marina Milner-Bolotin, Davor Egersdorfer, and Murugan Vinayagam. 2016. Investigating the effect of question-driven pedagogy on the development of physics teacher candidates' pedagogical content knowledge. *Physical Review Physics Education Research* 12, 2 (2016), 020128.

[67] Antonija Mitrovic, Brent Martin, Pramuditha Suraweera, Konstantin Zakharov, Nancy Milik, Jay Holland, and Nicholas Mcguigan. 2009. ASPIRE: An Authoring System and Deployment Environment for Constraint-Based Tutors. *http://iaied.org/pub/1150/* 19 (01 2009).

[68] Antonija Mitrovic, Brent Martin, Pramuditha Suraweera, Konstantin Zakharov, Nancy Milik, Jay Holland, and Nicholas Mcguigan. 2009. ASPIRE: An Authoring System and Deployment Environment for Constraint-Based Tutors. *Int. J. Artif. Intell. Ed.* 19, 2 (apr 2009), 155–188.

[69] Steven Moore, Huy A Nguyen, Norman Bier, Tanvi Domadia, and John Stamper. 2022. Assessing the quality of student-generated short answer questions using GPT-3. In *European conference on technology enhanced learning*. Springer, 243–257.

[70] Steven Moore, Huy A Nguyen, Tianying Chen, and John Stamper. 2023. Assessing the quality of multiple-choice questions using gpt-4 and rule-based methods. In *European Conference on Technology Enhanced Learning*. Springer, 229–245.

[71] Elham Mousavinasab, Nahid Zarifsanaiey, Sharareh R. Niakan Kalhori, Mahnaz Rakhshan, Leila Keikha, and Marjan Ghazi Saeedi. 2021. Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods. *Interactive Learning Environments* 29, 1 (2021), 142–163.

[72] Tom Murray. 2003. *An Overview of Intelligent Tutoring System Authoring Tools: Updated Analysis of the State of the Art.* Springer Netherlands, Dordrecht, 491–544. https://doi.org/10.1007/978-94-017-0819-7_17

[73] Maddisen Neuman, Callan Hundl, Aimee Grimaldi, Donna Eudaley, Darrell Stein, and Peter Stout. 2022. Blind testing in firearms: Preliminary results from a blind quality control program. *Journal of forensic sciences* 67, 3 (2022), 964–974.

[74] Changhoon Oh, Jungwoo Song, Jinhan Choi, Seonghyeon Kim, Sungwoo Lee, and Bongwon Suh. 2018. I Lead, You Help but Only with Enough Details: Understanding User Experience of Co-Creation with Artificial Intelligence. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3174223

[75] Ganiyat K Oloyede and Goodness J Ogunwale. 2022. Digital productivity tools as a necessity in education, research and career in the 21st century. In *Proceedings of the 31st Accra Bespoke Multidisciplinary Innovations Conference. Accra, Ghana: University of Ghana/Academic City University College.* 1–6.

[76] OpenAI. 2024. OpenAI API. https://platform.openai.com/.

[77] Zachary A. Pardos, Matthew Tang, Ioannis Anastasopoulos, Shreya K. Sheel, and Ethan Zhang. 2023. OATutor: An Open-source Adaptive Tutoring System and Curated Content Library for Learning Sciences Research. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–17. https://doi.org/10.1145/3544548.3581574

[78] Samir Passi and Mihaela Vorvoreanu. 2022. *Overreliance on AI: Literature Review.* Technical Report MSR-TR-2022-12. Microsoft. https://www.microsoft.com/en-us/research/publication/overreliance-on-ai-literature-review/

[79] Suraj Patil. 2020. T5 for multi-task QA and QG. https://huggingface.co/valhalla/t5-base-qa-qg-hl. https://openstax.org/details/books/physics.

[80] Yingtao Peng, Chen Gao, Yu Zhang, Tangpeng Dan, Xiaoyi Du, Hengliang Luo, Yong Li, and Xiaofeng Meng. 2024. Denoising Alignment with Large Language Model for Recommendation. *ACM Trans. Inf. Syst.* (Sept. 2024). https://doi.org/10.1145/3696662 Just Accepted.

[81] Savvas Petridis, Nicholas Diakopoulos, Kevin Crowston, Mark Hansen, Keren Henderson, Stan Jastrzebski, Jeffrey V Nickerson, and Lydia B Chilton. 2023. AngleKindling: Supporting Journalistic Angle Ideation with Large Language Models. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–16. https://doi.org/10.1145/3544548.3580907

[82] Kishore Prakash, Shashwat Rao, Rayan Hamza, Jack Lukich, Vatsal Chaudhari, and Arnab Nandi. 2024. Integrating LLMs into Database Systems Education. In *Proceedings of the 3rd International Workshop on Data Systems Education: Bridging Education Practice with Education Research* (Santiago, AA, Chile) *(DataEd '24)*. Association for Computing Machinery, New York, NY, USA, 33–39. https://doi.org/10.1145/3663649.3664371

[83] Hua Xuan Qin, Shan Jin, Ze Gao, Mingming Fan, and Pan Hui. 2024. CharacterMeet: Supporting Creative Writers' Entire Story Character Construction Processes Through Conversation with LLM-Powered Chatbot Avatars. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 1051, 19 pages. https://doi.org/10.1145/3613904.3642105

[84] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683 [cs.LG]

[85] NL Rane, SK Mallick, O Kaya, and J Rane. 2024. Applications of deep learning in healthcare, finance, agriculture, retail, energy, manufacturing, and transportation: A review. *Applied Machine Learning and Deep Learning: Architectures and Techniques* (2024), 132–152.

[86] Leena Razzaq, Jozsef Patvarczki, Shane F. Almeida, Manasi Vartak, Mingyu Feng, Neil T. Heffernan, and Kenneth R. Koedinger. 2009. The ASSISTment Builder: Supporting the Life Cycle of Tutoring System Content Creation. *IEEE Transactions on Learning Technologies* 2, 2 (April 2009), 157–166. https://doi.org/10.1109/TLT.2009.23 Conference Name: IEEE Transactions on Learning Technologies.

[87] Quentin Roy, Futian Zhang, and Daniel Vogel. 2019. Automation Accuracy Is Good, but High Controllability May Be Better. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3290605.3300750

[88] Asuman Şımşek and Safiye İpek Kuru Gönen. 2020. Raising awareness of EFL teachers on question types and pedagogical goals: An analysis through classroom modes. *Language Teaching and Educational Research* 3, 1 (2020), 56–75.

[89] Ninni Singh, Vinit Kumar Gunjan, Amit Kumar Mishra, Ram Krishn Mishra, and Nishad Nawaz. 2022. SeisTutor: a custom-tailored intelligent tutoring system and sustainable education. *Sustainability* 14, 7 (2022), 4167.

[90] Hari Subramonyam, Roy Pea, Christopher Pondoc, Maneesh Agrawala, and Colleen Seifert. 2024. Bridging the Gulf of Envisioning: Cognitive Challenges in Prompt Based Interactions with LLMs. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 1039, 19 pages. https://doi.org/10.1145/3613904.3642754

[91] Malcolm Thorburn and Katrina Seatter. 2015. Asking better questions! A review of the pedagogical strategies used in one senior level award in Scotland. *Journal of Pedagogy* 6, 1 (2015), 123–149. https://api.semanticscholar.org/CorpusID:143208362

[92] Computational Approaches to Human Learning (CAHL) Research. 2024. Open Source Intelligent Tutoring System w/ BKT (ReactJS and Firebase). https://github.com/CAHLR/OATutor/.

[93] Rice University. 2024. Physics - OpenStax. https://openstax.org/details/books/physics.

[94] KURT VanLEHN. 2011. The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems. *Educational Psychologist* 46, 4 (2011), 197–221. https://doi.org/10.1080/00461520.2011.611369 arXiv:https://doi.org/10.1080/00461520.2011.611369

[95] Tom Viering and Marco Loog. 2022. The shape of learning curves: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 6 (2022), 7799–7819.

[96] Prokopia Vlachogianni and Nikolaos Tselios. 2022. Perceived usability evaluation of educational technology using the System Usability Scale (SUS): A systematic review. *Journal of Research on Technology in Education* 54, 3 (2022), 392–409.

[97] Ines Šarić Grgić, Ani Grubišić, Slavomir Stankov, and Maja Štula. 2019. An agent-based intelligent tutoring systems review. *Int. J. Learn. Technol.* 14, 2 (jan 2019), 125–140. https://doi.org/10.1504/ijlt.2019.101847

[98] Sinh Trong Vu, Huong Thu Truong, Oanh Tien Do, Tu Anh Le, and Tai Tan Mai. 2024. A ChatGPT-based approach for questions generation in higher education. In *Proceedings of the 1st ACM Workshop on AI-Powered Q&A Systems for Multimedia*. 13–18.

[99] Bryan Wang, Yuliang Li, Zhaoyang Lv, Haijun Xia, Yan Xu, and Raj Sodhi. 2024. LAVE: LLM-Powered Agent Assistance and Language Augmentation for Video Editing. In *Proceedings of the 29th International Conference on Intelligent User Interfaces* (Greenville, SC, USA) *(IUI '24)*. Association for Computing Machinery, New York, NY, USA, 699–714. https://doi.org/10.1145/3640543.3645143

[100] Dongqing Wang, Hou Han, Zehui Zhan, Jun Xu, Quanbo Liu, and Guangjie Ren. 2015. A problem solving oriented intelligent tutoring system to improve students' acquisition of basic computer skills. *Computers & Education* 81 (Feb. 2015), 102–112. https://doi.org/10.1016/j.compedu.2014.10.003

[101] Jiyao Wang, Haolong Hu, Zuyuan Wang, Song Yan, Youyu Sheng, and Dengbo He. 2024. Evaluating Large Language Models on Academic Literature Understanding and Review: An Empirical Study among Early-stage Scholars. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 12, 18 pages. https://doi.org/10.1145/3613904.3641917

[102] Liuping Wang, Hongxin Li, Tong Wu, Fan Yang, Yang Yang, Jin Huang, and Feng Tian. 2024. Extrovert Increases Consensus? Exploring the Effects of Conversational Agent Personality for Group Decision Support. In *Proceedings of the Eleventh International Symposium of Chinese CHI* (Denpasar, Bali, Indonesia) *(CHCHI '23)*. Association for Computing Machinery, New York, NY, USA, 127–138. https://doi.org/10.1145/3629606.3629619

[103] Azmine Toushik Wasi, Mst Rafia Islam, and Raima Islam. 2024. LLMs as Writing Assistants: Exploring Perspectives on Sense of Ownership and Reasoning. In *Proceedings of the Third Workshop on Intelligent and Interactive Writing Assistants* (Honolulu, HI, USA) *(In2Writing '24)*. Association for Computing Machinery, New York, NY, USA, 38–42. https://doi.org/10.1145/3690712.3690723

[104] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. https://doi.org/10.48550/arXiv.2201.11903 arXiv:2201.11903 [cs].

[105] Daniel Weitekamp, Erik Harpstead, and Ken R. Koedinger. 2020. An Interaction Design for Machine Teaching to Develop AI Tutors. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3313831.3376226

[106] Tongshuang Wu, Ellen Jiang, Aaron Donsbach, Jeff Gray, Alejandra Molina, Michael Terry, and Carrie J Cai. 2022. PromptChainer: Chaining Large Language Model Prompts through Visual Programming. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems (CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, Article 359, 10 pages. https://doi.org/10.1145/3491101.3519729

[107] Changrong Xiao, Sean Xin Xu, Kunpeng Zhang, Yufang Wang, and Lei Xia. 2023. Evaluating Reading Comprehension Exercises Generated by LLMs: A Showcase of ChatGPT in Education Applications. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*. Association for Computational Linguistics, Toronto, Canada, 610–625. https://doi.org/10.18653/v1/2023.bea-1.52

[108] Qian Yang, Aaron Steinfeld, Carolyn Rosé, and John Zimmerman. 2020. Re-examining Whether, Why, and How Human-AI Interaction Is Uniquely Difficult to Design. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376301

[109] Yue Yu, Elizabeth Bonawitz, and Patrick Shafto. 2019. Pedagogical questions in parent–child conversations. *Child development* 90, 1 (2019), 147–161.

[110] Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story Writing With Large Language Models. In *27th International Conference on Intelligent User Interfaces (IUI '22)*. Association for Computing Machinery, New York, NY, USA, 841–852. https://doi.org/10.1145/3490099.3511105

[111] J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 437, 21 pages. https://doi.org/10.1145/3544548.3581388

[112] J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–21. https://doi.org/10.1145/3544548.3581388

[113] Gangyan Zeng, Yuan Zhang, Yu Zhou, Bo Fang, Guoqing Zhao, Xin Wei, and Weiping Wang. 2023. Filling in the Blank: Rationale-Augmented Prompt Tuning for TextVQA. In *Proceedings of the 31st ACM International Conference on Multimedia* (Ottawa ON, Canada) *(MM '23)*. Association for Computing Machinery, New York, NY, USA, 1261–1272. https://doi.org/10.1145/3581783.3612520

[114] An Zhang, Yang Deng, Yankai Lin, Xu Chen, Ji-Rong Wen, and Tat-Seng Chua. 2024. Large Language Model Powered Agents for Information Retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 2989–2992. https://doi.org/10.1145/3626772.3661375

[115] Chao Zhang, Xuechen Liu, Katherine Ziska, Soobin Jeon, Chi-Lin Yu, and Ying Xu. 2024. Mathemyths: Leveraging Large Language Models to Teach Mathematical Language through Child-AI Co-Creative Storytelling. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 274, 23 pages. https://doi.org/10.1145/3613904.3642647

[116] Zheng Zhang, Jie Gao, Ranjodh Singh Dhaliwal, and Toby Jia-Jun Li. 2023. VISAR: A Human-AI Argumentative Writing Assistant with Visual Programming and Rapid Draft Prototyping. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 5, 30 pages. https://doi.org/10.1145/3586183.3606800

[117] Yuanhang Zheng, Zhixing Tan, Peng Li, and Yang Liu. 2024. Black-box prompt tuning with subspace learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2024).

A The structure of pedagogical question

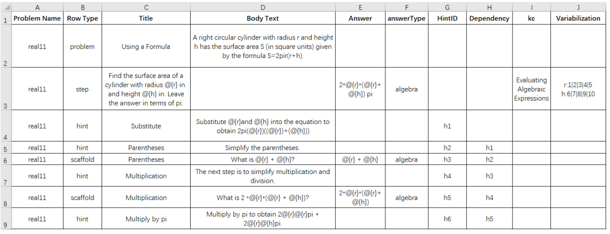
The template for creating pedagogical questions using Spreadsheet in the OAtutor system is shown in Figure 10. To simplify the process of creating pedagogical questions, we organize each pedagogical question into a title, body, and solution steps. Each step includes a title, body, answer, and hints, which can be divided into hints and scaffolding.

B RICTEF template and LLM output format

B.1 RICTEF prompt template

Based on several existing methods[19, 98], including RTF (Role, Task, Format), RISE (Rose, Input, Steps, Expectation), RTCF (Role, Task, Context, Format) and RTCEF (Role, Task, Context, Example, Format), we developed a RICTEF (Role, Input, Constraint, Task, Example, Format) prompt template to enhance the generation of pedagogical questions. This template supports the creation of various pedagogical questions using reference materials selected by the author.

RICTEF emphasizes two key factors: input and constraints. The input, derived from user-selected content, provides the contextual semantics and scenarios necessary for generating pedagogical questions. Constraints limit the scope of the LLM's output, preventing irrelevant content and ensuring that the generated questions align with teaching requirements, such as specified grade level and difficulty. As shown in Table 7, we provide a detailed explanation of the purpose of each factor in the RICTEF template, along with corresponding examples to support the explanation.

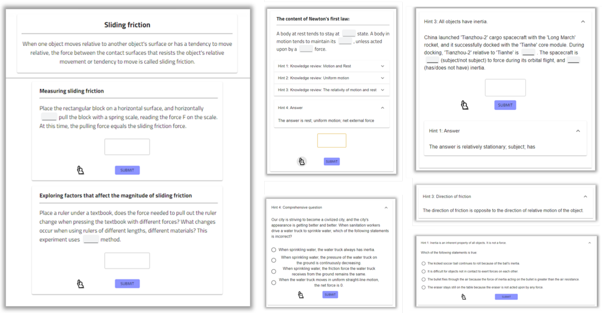


	A	B	C	D	E	F	G	H	I	J
1	Problem Name	Row Type	Title	Body Text	Answer	answerType	HintID	Dependency	kc	Variabilization
2	real11	problem	Using a Formula	A right circular cylinder with radius r and height h has the surface area S (in square units) given by the formula S=2pir(r+h).						
3	real11	step	Find the surface area of a cylinder with radius @{r} in. and height @{h} in. Leave the answer in terms of pi.		2*@{r}*(@{r}+@{h}) pi	algebra			Evaluating Algebraic Expressions	r:1|2|3|4|5 h:6|7|8|9|10
4	real11	hint	Substitute	Substitute @{r}and @{h} into the equation to obtain 2pi(@{r})((@{r})+(@{h}))			h1			
5	real11	hint	Parentheses	Simplify the parentheses.			h2	h1		
6	real11	scaffold	Parentheses	What is @{r} + @{h}?	@{r} + @{h}	algebra	h3	h2		
7	real11	hint	Multiplication	The next step is to simplify multiplication and division.			h4	h3		
8	real11	scaffold	Multiplication	What is 2 *@{r}*(@{r} + @{h})?	2*@{r}*(@{r}+@{h})	algebra	h5	h4		
9	real11	hint	Multiply by pi	Multiply by pi to obtain 2@{r}@{r}pi + 2@{r}@{h}pi.			h6	h5		

Figure 10: Pedagogical question in OATutor with with Spreadsheet creation.

B.2 Formatting for LLM Outputs

To streamline the parsing of LLM outputs, we instruct the LLM to structure its responses in JSON format. For instance, in the final output of the question generation chain, we prompt the LLM to



Sliding friction

When one object moves relative to another object's surface or has a tendency to move relative, the force between the contact surfaces that resists the object's relative movement or tendency to move is called sliding friction.

Measuring sliding friction

Place the rectangular block on a horizontal surface, and horizontally ____ pull the block with a spring scale, reading the force F on the scale. At this time, the pulling force equals the sliding friction force.

Exploring factors that affect the magnitude of sliding friction

Place a ruler under a textbook, does the force needed to pull out the ruler change when pressing the textbook with different forces? What changes occur when using rulers of different lengths, different materials? This experiment uses ____ method.

The content of Newton's first law:

A body at rest tends to stay at ____ state. A body in motion tends to maintain its ____ , unless acted upon by a ____ force.

Hint 1: Knowledge review: Motion and Rest

Hint 2: Knowledge review: Uniform motion

Hint 3: Knowledge review: The relativity of motion and rest

Hint 4: Answer

The answer is rest; uniform motion; net external force

Hint 3: All objects have inertia.

China launched 'Tianzhou-2' cargo spacecraft with the 'Long March' rocket, and it successfully docked with the 'Tianhe' core module. During docking, 'Tianzhou-2' relative to 'Tianhe' is ____ . The spacecraft is ____ (subject/not subject) to force during its orbital flight, and ____ (has/does not have) inertia.

Hint 1: Answer

The answer is relatively stationary; subject; has

Hint 4: Comprehensive question

Our city is striving to become a civilized city, and the city's appearance is getting better and better. When sanitation workers drive a water truck to sprinkle water, which of the following statements is incorrect?

○ When sprinkling water, the water truck always has inertia.
○ When sprinkling water, the pressure of the water truck on the ground is continuously decreasing.
○ When sprinkling water, the friction force the water truck receives from the ground remains the same.
○ When the water truck moves in uniform straight-line motion, the net force is 0.

Hint 3: Direction of friction

The direction of friction is opposite to the direction of relative motion of the object.

Hint 1: Inertia is an inherent property of all objects. It is not a force.

Which of the following statements is true:

○ The kicked soccer ball continues to roll because of the ball's inertia.
○ It is difficult for objects not in contact to exert forces on each other.
○ The bullet flies through the air because the force of inertia acting on the bullet is greater than the air resistance.
○ The eraser stays still on the table because the eraser is not acted upon by any force.

Figure 11: Examples of pedagogical questions created by participants using TutorCraftEase (translated into English), including solution step, hints and scaffolding.

output a JSON array, where each element is an object containing keys for ’title’, ’body’, ’question’, ’answer’, and ’instructions’. Within ’instructions’, each element is further defined as an object with keys for ’type’, ’title’, ’text’, and ’answer’, where ’type’ can be either ’hint’ or ’scaffolding’, and ’answer’ is required only for scaffolding. Upon receiving the output from the LLM, we employ regular expressions to correct any formatting errors, ensuring the output adheres to valid JSON syntax. This includes detecting and escaping unescaped special characters by prefixing them with a backslash, thereby facilitating smooth JSON parsing and integration into the system.

C Examples of generated pedagogical questions

The pedagogical questions generated by TutorCraftEase are shown in Figure 11, including fill-in-the-blank questions, multiple-choice questions, and others. The generated pedagogical questions are displayed in the preview panel, where users can verify and check their answers by entering them in the designated input area. Additionally, users can access hints and scaffolding through the assistive button (the icon: A person with their hand raised).

Table 7: RICTEF prompt elements and their uses.

Factor	Purpose	Example
Role	Define the perspective or assumed persona to guide the tone and intent of question creation.	Act as a physics teacher
Input	Provide the selected text by user.	The information selected by the user from the reference panel, such as "What is temperature? It's one of those concepts..."

Constraint	Constraints specify custom parameters, including grade level, question type, knowledge points, and sub-questions, to tailor the generated question.	Grade: 8th-grade;
		question type: true/false question;
		course: physical;
		knowledge points: boiling point

Task	Describe the task you want the LLM to perform, combined with the specified constraints.	Create a pedagogical question based on the constraint
Example	Provide specific example instructions to guide the LLM's response.	"question": "The boiling point of water is typically 100°C at standard atmospheric pressure"; "answer":"True"
Format	Define the desired output structure, including question, options, answer, type, etc.	the format of true/false question