Empowering Linux Experimental Teaching with Cursor: Ecological Connotations and Practical Approaches

Pengcheng Lin

College of continuing Education, Xiamen university of technology, Xiamen, Fujian, China

linpc@xmut.edu.cn

Abstract. Large models are driving transformative changes in computer experimental teaching. Their efficiency in handling computational tasks perfectly addresses foundational needs in computer course instruction. This paper examines the ecological impact of large models on experimental teaching and explores practical pathways for integrating large models throughout experimental teaching processes. Using the application of Cursor in Linux system courses as a case study, it illustrates the instructional practices involved. Experimental results indicate that this approach effectively enhances student learning efficiency and assists them in tackling complex problems.

Keywords: Large Models; Linux Experimental Teaching; Cursor

1 Introduction

Recent advances in Generative AI and LLMs, exemplified by Copilot [1] and ChatGPT, highlight their promising ability to address complex problems with human-like expertise. While efficient in handling programming tasks, they raise concerns about misuse affecting teaching integrity. Researchers are exploring their pros and cons in education.

Internationally, Ramazan [2] used ChatGPT in the programming course and found that students obtained reasonably accurate answers quickly, thus saving time searching for answers. Students appreciated its convenience but feared laziness and cognitive decline from over-reliance. Similarly, Quershi [3] found the ChatGPT group scored higher and faster than the textbook group in programming challenges. Domestically, Zhang Jin [4] proposed reshaping teaching goals to enhance computer systems skills. Zhai Jie [5] experimented with incorporating large language models into educational assessment.

As technology evolves, multimodal large models are trending beyond language processing. Research on their integration into computer teaching is limited. This paper explores ecological construction and practical pathways for experimental teaching using the Cursor, focusing on Linux systems and current large model applications.

M. F. A. Ghani et al. (eds.), *Proceedings of the 2024 4th International Conference on Education, Language and Art (ICELA 2024)*, Advances in Social Science, Education and Humanities Research 907,
https://doi.org/10.2991/978-2-38476-364-1_32

2 Ecological Connotations of Large Model Empowerment in Experimental Teaching

The empowerment of computer experimental teaching through large models involves four stages: ecological environment, input, transformation process, and output.[6]

Ecological Environment: This includes students, instructors, settings, and resources. Traditionally, instructors bridge students to resources. Large models add a service layer, enhancing access to knowledge and tailored learning, and providing resources for instructors to improve teaching quality.

Input Diversification: Large models offer diverse information pathways, aligning with differentiated instruction to meet students' unique learning needs.

Transformation Process: They aid in knowledge transformation by providing solutions and insights, facilitating information interpretation. Multimodal models, using audio, images, and virtual experiments, offer practical learning experiences, helping students build comprehensive knowledge frameworks.

Output and Dynamic Evaluation: Large models shift assessments to a holistic system, with continuous learning documentation as feedback. They also provide personalized career advice based on assessments.

Overall, large models transform all elements and stages of experimental teaching, presenting new challenges and opportunities.

3 Empowering Linux System Practical Courses with Cursor

The Linux course, a foundational computer science subject, is tool-intensive, broad, and time-sensitive. Teaching focuses on imparting knowledge and teaching students how to learn the Linux OS to adapt to evolving versions. Cursor, an IDE based on VS Code, features the advanced Claude Sonnect multimodal large model for programming tasks. This paper leverages large models' potential, using Cursor to implement open experimental teaching in the Linux course.

3.1 Objectives of Experimental Teaching

This experimental course consists of four sessions, totaling 8 hours, covering environment setup, shell command practice, web server setup, and programming under Linux. The objective is to restructure the experimental teaching through a project-driven approach, as outlined in Table 1.

Table 1. Objectives of Linux Operating System Experimental Teaching

Project	Specific Content	Experimental Teaching Objectives
Environment Setup	1.Install VMware and the Linux operating system, and learn the basics of Cursor usage.	1.Master virtual machine setup and Linux installation, and understand basic Cursor usage.

Shell Command Pratice	1.Learn to use common Shell commands and engage in advanced Shell programming.	1.Master basic Linux commands and advanced Shell script programming, using Cursor to assist in learning commands and writing advanced scripts, thereby enhancing scripting and automation skills.
Web Server Setup	1.Setting up an Nginx-based server 2.Configuring Nginx	1.Use Cursor to master Nginx setup and configuration, understand web server principles, and enhance server deployment and network optimization skills.
Programming in Linux Environment	1.Set up a programming environment and implement large model deployment in Linux.	1.Set up the Python environment in Linux and use Cursor to deploy large models, mastering tools like Ollama and Gradio.

Compared to traditional methods, Linux teaching with large models cultivates self-directed learning, enhances practical skills, and improves innovation. Cursor's large model analyzes learning records and feedback, helping students identify progress and bottlenecks, enabling teachers to adjust plans and improve teaching quality.

3.2 Application of Cursor in Experimental Teaching Scenarios

Four experimental projects with increasing difficulty levels are set according to the experimental objectives.

Experiment 1: Experimental Environment Setup

Set up the Linux environment using VMware and deploy Cursor, the large model carrier. Learn Cursor's interactive interface, including intelligent completion, syntax completion, and error prompts, to complete system configurations effectively.

Experiment 2: Shell Command Practice

Adopting a problem-oriented approach, guide students to inquire about relevant commands and their corresponding parameter usage from Cursor based on specific needs, and introduce advanced Shell programming through general questions. For example, to create a file in the "/home/" directory, guide students to obtain the usage of the "touch" command and other parameters through Cursor, as shown in Figure 1.

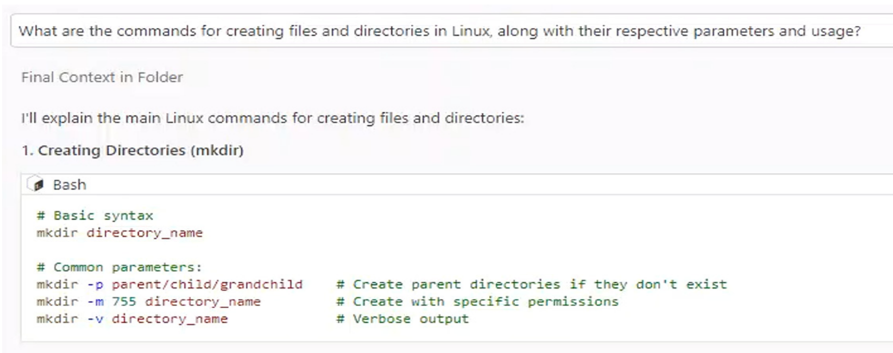

Fig. 1. Command Usage Inquiry

Experiment 3: Web Server Setup

Using the most common web server scenario, introduce Nginx. Utilize Cursor to explore the structure and specific purposes of Nginx files.Next, based on the application scenario, use Cursor to complete the modification of configuration files.

Experiment 4: Programming in Linux Environment

Take Python programming under Linux as an example, and explore the process of deploying large models on Linux through Cursor. With the support of Cursor, use tools like Ollama and Gradio to deploy large models. As shown in Figure 2, Cursor provides configuration steps and optimization suggestions, helping students understand how to set up large model applications.

1. Run a large model using Ollama and enable the corresponding web interface; 2. Use Gradio to build the corresponding web front-end page.

I'll help you create a solution that combines Ollama with a Gradio web interface. Here's how we can implement this:

1. First, let's create a Python script that sets up both Ollama and Gradio:

app.py

```python
import gradio as gr
from langchain.llms import Ollama

def generate_response(prompt, model_name="llama2"):
    """Generate response using Ollama model"""
    try:
        # Initialize Ollama
        llm = Ollama(model=model_name)

        # Generate response
        response = llm(prompt)
        return response
    except Exception as e:
        return f"Error: {str(e)}"

# Create Gradio interface
def create_interface():
    # Define the interface
    iface = gr.Interface(
        fn=generate_response,
        inputs=[
            gr.Textbox(label="Enter your prompt", lines=4),
            gr.Dropdown(
                choices=["llama2", "mistral", "codellama"],
                label="Select Model",
                value="llama2"
            )
        ],
        outputs=gr.Textbox(label="Response", lines=8),
        title="Ollama Chat Interface",
        description="Chat with different LLMs using Ollama",
    )
    return iface

if __name__ == "__main__":
    # Create and launch the interface
    interface = create_interface()
    interface.launch(server_name="0.0.0.0", server_port=7860, share=True)
```

Fig. 2. Cursor-Assisted Local Large Model Setup

3.3 Implementation Effectiveness

During the 2023-2024 academic year at Xiamen University of Technology, an open-ended experiment was conducted twice across different majors. One course used traditional methods, the other employed a new approach with Cursor. The results of the grades are shown in Table 2. Cursor improved efficiency, enhancing students' response speed, experiment completion, and report quality, especially in high-difficulty tasks.

Table 2. The average score of courses and the median score

Semester	Average Score	Median
Fall 2023-2024	80.6	80
Spring 2023-2024	97	97

Through the implementation of the new approach, students quickly mastered relevant knowledge of the Linux operating system, and those trained achieved excellent results in computer-related competitions. Practice has proven that the method of empowering experimental teaching with large models is effective.

4 Conclusion

In the reform of computer experimental course teaching, large models can serve both theoretical teaching and enhance students' practical skills and innovative awareness. This article proposes a teaching reform approach empowered by large models and uses Cursor as a tool carrier to elaborate on the implementation plan of open-ended experimental teaching for this course. From the implementation effects, the experimental teaching reform based on Cursor effectively improved students' operational skills and self-directed learning abilities, promoting a deeper understanding of Linux system operations and applications.

Acknowledgement

Funding Project: Xiamen University of Technology 2024 University-Level Education and Teaching Research Project, Project Number: JYSY202405.

References

1. Phung, T., Pădurean, V. A., Cambronero, J., Gulwani, S., Kohn, T., Majumdar, R., ... & Soares, G. (2023, August). Generative ai for programming education: Benchmarking chatgpt, gpt-4, and human tutors. In Proceedings of the 2023 ACM Conference on International Computing Education Research-Volume 2 (pp. 41-42).
2. Yilmaz, R., & Yilmaz, F. G. K. (2023). Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning. Computers in Human Behavior: Artificial Humans, 1(2), 100005.
3. Zhang, Jin, Gong, Xiaoli, Gao, Xiaopeng. (2024) Intelligent collaborative teaching experiment design for cultivating computer system capabilities in the era of large models. Laboratory Science, 27(2): 21-23.
4. Zhai, Jie, Li, Yanhao, Meng, Tianxin, et al. (2023) Exploration and Practice of Personalized Computer Experiment Teaching Based on Decision Trees and Large Models. Experimental Technology and Management, 40(12): 8-15.
5. Xu, Xiaofei, Mi, Jie, Chen, Wenbai. (2019) Engineering Training Platform for Collaborative Filtering Recommendation Systems in the Perspective of Artificial Intelligence. Experimental Technology and Management, 36(4): 109-113.

6. Lin, Pengcheng. (2024) Ecological Connotation, Application Model, and Practical Approach of ChatGPT Empowering Continuing Education. Adult Education, 44(5): 19-23. DOI: 10.3969/j.issn.1001-8794.2024.05.004