Educational Data Analysis using Generative AI

Abdul Berr[1], Sukrit Leelaluk[2], Cheng Tang[2], Li Chen[2], Fumiya Okubo[2], and Atsushi Shimada[2]

[1] *Department of Electrical Engineering and Computer Science, Kyushu University, Japan*
[2] *Graduate School of Information Science and Electrical Engineering, Kyushu University, Japan*

Abstract

With the advent of generative artificial intelligence (AI), the scope of data analysis, prediction of performances, real-time feedback, etc. in learning analytics has widened. The purpose of this study is to explore the possibility of using generative AI to analyze educational data. Moreover, the performances of two large language models (LLMs): GPT-4 and text-davinci-003, are compared with respect to different types of analyses. Additionally, a framework, LangChain, is integrated with the LLM in order to achieve deeper insights into the analysis, which can be beneficial for beginner data scientists. LangChain has a component called an agent, which can help study the analysis being performed step-by-step. Furthermore, the impact of the OpenLA library, which pre-processes the data by calculating the number of reading seconds of students, counting the number of operations performed by students, and making page-wise behavior of each student, is also studied. Besides, factors with the most significant impact on students' performances were also discovered in this analysis. The results show that GPT-4, when using the data pre-processed by OpenLA, provides the best analysis in terms of both, the accuracy of the final answer, and the step-by-step insights provided by LangChain's agent. Also, we learn the significance of reading time and interactions used (Add marker, bookmark, memo) by students in predicting grades.

Keywords

Generative AI, Large Language Models, LangChain, Data Analysis, OpenLA

1. Introduction

Over the past few years, the rise of distance learning has helped e-learning environments grow further. This has made the collection of different types of student data easier. Since the start of student data collection in the 1960s, the research in this field has progressed from database management systems to advanced database systems and then finally to advanced data analysis, which incorporates different data mining techniques [1]. Such developments have led to an increase in leveraging data mining techniques to predict and analyze students' academic performances. For example, intelligent systems have also been made using data mining algorithms like neural networks, decision trees, etc. to analyze and uncover the relationship between different data inputs and performances of students [2]. Moreover, different machine learning algorithms have been compared and implemented to find out the attributes with the greatest impact on the students' performances [3]. Similarly, studies have been done on students' behaviors and interactions acquired by log functionality of e-books, and the interpretable model, the Dempster-Shafer classifier (DS classifier), was implemented and compared with other models in order to predict students' academic performances [4].

With the recent advancements in the field of Artificial Intelligence (AI), numerous opportunities are opened in education and research. A few of the significant benefits include personalized learning experiences, adaptive learning materials, real-time feedback, assessments, etc. [5]. The advent of generative AI, in particular, has provided education with a completely new dynamic. With the help of generative AI, we can generate new content in contrast to the traditional AI which works on predefined rules. As one of the classes of AI models, generative AI can produce content like text, figures, and other media based on the learning of existing data. These models are composed of different deep learning techniques and neural networks that help in analyzing and generating human-like content. Recently, generative AI has been used in many different fields, be it business, marketing, education, or research [6]. Teaching and learning have been specially improved using generative AI. For instance, many instructors had very positive

feedback about generative AI's range of applications for language teaching [7]. Similarly, teaching assistant chatbots have also been pioneered which provide personalized learning and encourage student inquiry and learning [8]. However, in order to expand the prowess of generative AI in the field of education, it is imperative to analyze students' data and infer different relationships for the development of applications that can be helpful to enhance the learning experiences of both students and instructors.

Generative AI has been mainly popularized by the development of ChatGPT by OpenAI. The OpenAI API has a diverse set of models with varying capabilities. The purpose of this study is to explore further applications of generative AI in the field of learning analytics by using OpenAI's GPT-4 model [9] and text-davinci-003 model [10] for the tabular data analysis of students' interactions with the e-book and provide deeper insights into the analysis with the help of integration of langchain framework [11] and OpenLA library.

2. Methodology

The data has been acquired from the e-book environment in the form of log data (Event Stream) [12]. Afterwards, two sets of data are made. One is pre-processed with the OpenLA library, and the other one is without pre-processing. Further, we integrate the LangChain framework with GPT-4 and text-davinci-003 models to study different insights during the analysis and also the difference between the performance of models with respect to different types of analyses.

2.1. Data Collection

The dataset contains different types of data from 163 students. For instance, student's id, content id, reading time of each page in the form of event stream data, interactions done by students on the e-book for example, adding a marker or bookmark, moving onto the next page, returning to the previous page, adding memo, and so on. Besides, the grades of the students are also stored in a separate file ranging from A to F. A is the highest grade which can be obtained to pass, and D is the lowest grade to pass, and F is the failing grade. Figure 1 below shows an example of the data. The focus of this particular study is the analysis of students' reading behaviors and their interactions, which can be used to deduce significant correlations.

userid	contentsid	devicecode	marker	eventtime	operationn	pageno
B-2019_U1	B-2019_C1	pc		09/04/2019 14:59	OPEN	1
B-2019_U1	B-2019_C1	pc		09/04/2019 15:00	NEXT	1
B-2019_U1	B-2019_C1	pc		09/04/2019 15:00	NEXT	2
B-2019_U1	B-2019_C1	pc		09/04/2019 15:00	NEXT	3

Figure 1: Characteristics of the data

2.2. Integrated Analysis System

LangChain is an open-source framework that allows software developers or data scientists to work with artificial intelligence. The purpose is to integrate LLMs like OpenAI's text-davinci-003 and GPT-4 with external sources. In this study, the comparison of table data analysis of GPT-3.5 and GPT-4 was tried to be drawn. To choose the best model for comparison with GPT-4, all the GPT-3.5 models were tried for the analysis at first. text-davinci-003's ability to understand the prompt seemed to surpass the other models and hence it was chosen for the comparison. LangChain incorporates many components that allow this convenient linking of LLMs with it [13]. However, in this research the component that is used most is agent.

Agents use LLMs and help to choose a sequence of actions to take. More importantly, agents have access to many tools, and they decide which tool to use according to the user's input, as the language model takes the prompt constructed by the prompt template to return some output.
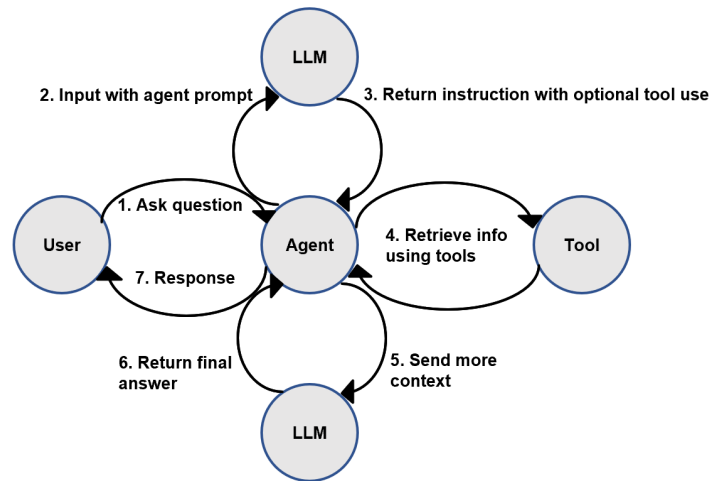
Figure 2: Flowchart of Agent's mechanism

Figure 2 illustrates how the agent handles the chain calls and gives suitable output to the user with the help of tools. After the LLM receives the input in step 2, it gives agent instructions regarding different tools which agent can use. Using those tools, for example in figure 3 python_repl_ast is the tool used, which is a tool for running Python code in a Read-Eval-Print Loop (REPL), it sends the LLM with more context about what action to take. One of the most significant features of agent in this study and analysis is the AgentExecutor. This is what makes the call, and implements the actions chosen [14].

```
> Entering new AgentExecutor chain...
Thought: I need to combine the data from all the dataframes
Action: python_repl_ast
Action Input: df1['ADD MARKER'] + df2['ADD MARKER'] + df3['ADD MARKER'] + df4['ADD MARKER']
Observation: KeyError: 'ADD MARKER'
Thought: I need to look at the dataframes to see which ones have the 'ADD MARKER' column
Action: python_repl_ast
Action Input: df1.columns.contains('ADD MARKER')
Observation: AttributeError: 'Index' object has no attribute 'contains'
Thought: I need to use the 'in' operator
Action: python_repl_ast
Action Input: 'ADD MARKER' in df1.columns
Observation: True
Thought: I need to sum the 'ADD MARKER' columns from the dataframes that have it
Action: python_repl_ast
Action Input: df1['ADD MARKER'].sum() + df2['ADD MARKER'].sum() + df3['ADD MARKER'].sum()
Observation: 63519.0
Thought: I now know the final answer
Final Answer: 63519 student interactions ('ADD MARKER', 'ADD MEMO', 'ADD BOOKMARK') are stored in the logs.
```

Figure 3: AgentExecutor Chain

Data and the LLM model need to be provided to the agent. The agent created for this study is the Pandas Dataframe agent [15]. In this case, the data provided is in the form of dataframes. This is a powerful component that allows the handling of large datasets and makes applications capable of question-answering over Pandas Dataframes. Another agent, csv agent, is also created in order to study the performance of LLM on data that is not pre-processed by OpenLA. The data provided in this case is in csv file format.

2.3. OpenLA Library

As discussed earlier, the OpenLA library is used in the pre-processing of the log data [11]. Here, we only discuss how it reshapes the data before the analysis. In this analysis, we use OpenLA to convert data into three different types: operation count in each content, behavior in each page, and behavior in each page with consideration page transition, e.g., going back and jumping to a page. The pre-processing with OpenLA allows the extraction of logs with the required information. In this experiment, we extract the total number of each operation performed by students in each content. Also, the average number of reading seconds and each operation's total count for each page are acquired. Finally, the time of entry and exit from each page along with the

total number of reading seconds of each page were tabulated as well through pre-processing by OpenLA. This helps in easy retrieval of precise information.

2.4. Prompt Categorization with Respect to Analysis

In order to study the performance of the two LLMs: GPT-4 and text-davinci-003, three different levels of analyses are performed, and prompts are categorized accordingly. Each level contains 20 prompts. Table 1 shows each type of analysis performed. For example, in level 1, Reading time analysis refers to calculation of number of reading seconds of different parts of the e-book, Students and interaction use determines the type and number of interactions used on different parts of the e-book, Grade distribution trend tells how the grades are spread out in the whole dataset, and Reading time relationships illustrates how different factors, like student interactions, are related with the number of seconds students read. In level 2, Factors affecting grades draws relationships between different factors, like reading seconds and interactions, and students' grades and Content and reading analysis covers reading patterns across different content ids. Finally, in level 3, Predictive modelling prompts suggests different algorithms to predict grades, Optimal learning strategies discovers how students can improve learning experiences to achieve better grades, Personalized interventions help generate models giving personalized summaries of e-book based on different factors like reading patterns, and finally, Predictive models consists of prompts which generate models to predict performances of future students.

Table 1: This is the categorization of each level.

Level 1 (Reading interaction analysis)	Level 2 (Factors influencing students' performance)	Level 3 (Prediction analysis)
1. Reading time analysis	**1.** Factors affecting grade	**1.** Predictive modelling for grades of students
2. Students and interaction use	**2.** Content and reading analysis	**2.** Optimal learning strategies
3. Grade distribution trend		**3.** Personalized intervention
4. Reading time relationships		**4.** Predictive models

Table 2: Prompt example of each level of analysis.

Level 1 (Reading interaction analysis)	Level 2 (Factors influencing students' performance)	Level 3 (Prediction analysis)
Identify any significant differences in reading times between different content ids.	Give a decision tree model to determine the most influential factors in predicting grades.	Develop a predictive model to estimate the grade of a student based on their reading behavior and interaction patterns

To sum up the whole procedure of making the integrated analysis system, firstly, we collect data from the e-book environment, then it is pre-processed through the OpenLA library. Afterward, we create two agents, pandas dataframe and csv, for data with OpenLA and without OpenLA respectively. Finally, we give the prompts and evaluate the results based on two

parameters, task-specific performance, and the agent's thought/observation/output from the AgentExecutor chain. Task-specific performance refers to the accuracy of the final answer provided, in other words, we confirm if the final output is a logical answer or not. In the case of not being able to provide a correct answer, the AgentExecutor chain is referred to study the thoughts and observations during the analysis. This further insight is helpful on many occasions regardless of the accuracy of the final output.

Figure 4: Summary of whole method

3. Results

In this study, we evaluate the performance of two models, GPT-4 and text-davinci-003, with regard to the analysis of students' performances. In addition, we also compare both models' efficiency with and without the use of the OpenLA library. Moreover, we also study the analysis provided by these models. Finally, we will conclude the best generative AI model for the educational table data analysis.

3.1. Comparison of Models

In the experiment, 20 prompts were used for each type of analysis. In order to understand the results more clearly, each category's label has been defined in detail in the section 2.4. The results were evaluated after a comparison of the final analysis done by models and manual, human analysis, done personally, of the raw, log data. Further, each level was subdivided into further categories to make the comparison more precise. For each prompt, the correct result was tried to be drawn manually from the csv files and then compared with the result generated by the AI model. In the case of level 3's predictive modelling analysis, the results were evaluated after human analysis of the AgentExecutor chain, an example of which is shown in figure 3. For any model or algorithm given by the AI model, the actual practicality of that algorithm was tested manually, separately. The following results demonstrate the percentage of prompts the models were able to accurately answer.

Table 3: Level 1 results of each category of analysis.

Type of analysis	GPT-4 with OpenLA	GPT-4 without OpenLA	text-davinci-003 with OpenLA	text-davinci-003 without OpenLA
Reading time analysis	80%	0%	80%	15%
Students and interaction use	100%	75%	100%	75%
Grade distribution trend	65%	65%	50%	30%
Reading time relationships	65%	35%	65%	0%

From Table 3, we can deduce that the use of the OpenLA library is integral to this table data analysis. Furthermore, we can also see a slight difference in the accuracy of being able to answer correctly between GPT-4 and text-davinci-003. In this case, only one of the analyses: Grade distribution and trends, shows the superiority of GPT-4 over text-davinci-003 when used with OpenLA. On the contrary, all the models underperform when analysing reading time relationships. Similarly, both the models are not able to give accurate answers when analysing the data without

being pre-processed by OpenLA, especially in the case of reading time analysis GPT-4 was not able to answer any question, and text-davinci-003 answered only 15% of the questions. In the case of reading time relationships, text-davinci-003 without OpenLA could not answer any question while GPT-4 without OpenLA could answer only 35% of the time. Overall, students and interaction use was the easiest to analyse by both the models and reading time relationships were the hardest.

Table 4: Level 2 results of each category of analysis.

Type of analysis	GPT-4 with OpenLA	GPT-4 without OpenLA	text-davinci-003 with OpenLA	text-davinci-003 without OpenLA
Factors affecting grade	65%	40%	45%	15%
Content and reading analysis	65%	15%	65%	35%

From Table 4, we again see how the use of OpenLA library has a positive impact on the analysis. In the case where OpenLA is used, GPT-4 has better accuracy than text-davinci-003. Overall, it is notable that both models are not as effective for level 2 as for level 1.

Table 5: Level 3 results of each category of analysis.

Type of analysis	GPT-4 with OpenLA	GPT-4 without OpenLA	text-davinci-003 with OpenLA	text-davinci-003 without OpenLA
Predictive modelling for grades of students	65%	35%	50%	0%
Optimal learning strategies	50%	0%	50%	50%
Personalized intervention	75%	50%	50%	0%
Predictive models	65%	35%	15%	15%

Similar to before, table 5 also infers that using OpenLA significantly improves the analysis regardless of whichever type of language model was used. Further, we can also see a considerably better performance of GPT-4 in the level 3 category of analysis as compared to text-davinci-003. Moreover, it can be observed that the performance of text-davinci-003 with or without OpenLA for level 3 has dropped compared to previous levels except in the case of optimal learning strategies. All in all, both models perform best in level 1 analysis of students and interaction use, while generating predictive models, in level 3 analysis, seemed to be harder for all the integrated analysis systems. It is important to note that the performance of GPT-4 with OpenLA is consistent in most of the analyses.

3.2. Study of Students' Data analysis

Many key results and relationships were found with the analysis provided by generative AI. The most important one out of those are the relationships between students' performances and their reading times and interaction use.
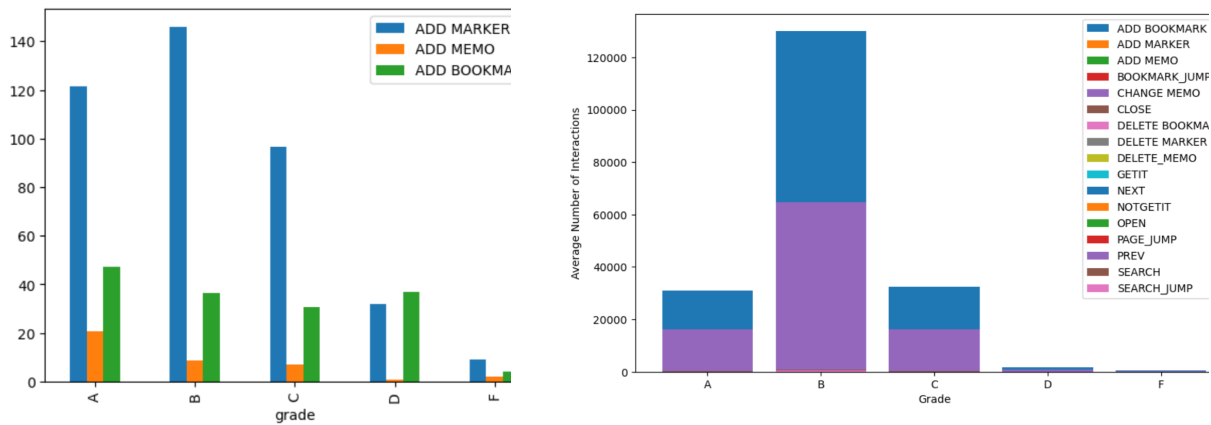
Figure 5: Comparison of graphs generated by GPT-4 with and without OpenLA for average number of different interactions (marker, memo, bookmark) for each grade, level 1 analysis.

Figure 5 illustrates the comparison of graphs generated by GPT-4 with and without OpenLA when prompted with: "plot graph of an average number of different interactions (marker, memo, bookmark) for each grade". On the left is the graph generated by GPT-4 with OpenLA and when compared with the log data, the plot turned out to be a correct representation of the average number of interactions for each grade i.e., students with grade B interacted with e-book the most. On the other hand, the graph on the right is also generated by GPT-4 but with data without pre-processed by OpenLA. In this case, since the original log data contained so many interactions, the plot was not able to correctly represent all of the interactions. In addition, students with grade C used slightly more interactions than students with grade A which also contradicts the original log data.
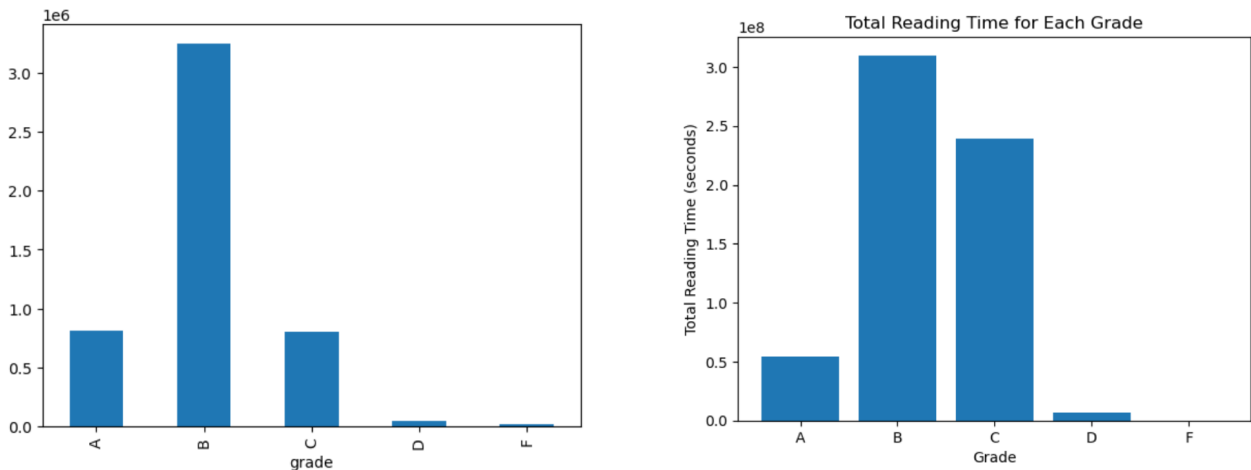

Figure 6: Comparison of graphs generated by GPT-4 with OpenLA and text-davinci-003 without OpenLA and GPT-4 without OpenLA for number of reading seconds for each grade, level 1 analysis.

Figure 6 shows the plot, on the left, given by GPT-4 with OpenLA and text-davinci-003 without OpenLA when prompted with: "plot graph of number of reading seconds for each grade." The main result i.e., students with grade B read the most is correct depiction of the actual data as well. However, the difference in reading time between A and C is not as plotted. In other words, students with grade A read considerably more than students with grade C. Also, text-davinci-003 with OpenLA failed to generate any plot at all. On the right, we get the plot for the analysis done by GPT-4 without OpenLA. Although it correctly shows that students with grade B read the most, the plot for students with grade C and A do not match the log data.

4. Discussion

Having obtained the analyses, we can now note the differences between the two models. From the experiment, we can say that out of all four ways we discussed, GPT-4 with OpenLA gives the

best results for all types of analyses performed, and the best performance for every integrated analysis system was achieved in level 1. To understand the reasoning behind the better performance of GPT-4 in general, we should be aware of difference between text-davinci-003's and GPT-4's level of instruction comprehension. Prompt engineering is a vital factor for text-davinci-003 for clear comprehension of instructions [10]. On the other hand, GPT-4 has been further fine-tuned in accordance with human feedback and infrastructure for better predictability has also been implemented [9]. In order to discuss the performance of OpenLA it is important to look at the features of data before and after pre-processing. The importance of data pre-processing has previously been addressed in different areas, for example in web usage mining process [16]. An example of data before pre-processing can already be seen in Figure 1. Figure 7 below shows one of the examples of the pre-processed data. Comparing Figure 1 and Figure 7, we can see that the nature of the information we want to know is more precisely broken down by OpenLA and this leads to the better performance of the models used with the pre-processed data by OpenLA.

Another important inference that can be made is about the insights provided by the AgentExecutor chain. As shown in Figure 3, we are provided with a step-by-step process of the analysis done by the LLMs. The quality of insight also varies depending on the model used. From the experiment, we get to know that insights provided by GPT-4's, are clearer and deeper to understand the procedure of analysis. Finally, we also studied the impact of reading behaviour and interaction with e-book on the grades of students. These inferences are important for the prediction of grades in future courses as well.

userid	contentsid	pageno	num_visits	average_reading_seconds	reading_seconds	OPEN	CLOSE	ADD MARK	ADD MEMO	ADD BOOKMARK
B-2019_U1	B-2019_C1	1	4	20.25	81	2	0	1	0	0
B-2019_U1	B-2019_C1	2	3	59	177	0	0	3	1	0
B-2019_U1	B-2019_C1	3	2	16.5	33	0	0	0	0	0
B-2019_U1	B-2019_C1	4	3	13.66666667	41	0	0	0	0	0
B-2019_U1	B-2019_C1	5	3	20.33333333	61	0	0	1	0	0
B-2019_U1	B-2019_C1	6	5	23.4	117	0	0	3	1	0

Figure 7: Pre-processed data by OpenLA

5. Conclusion and Future Work

In this study, we proposed a new mechanism to perform educational table data analysis using generative AI, GPT-4, and text-davinci-003. We compared the performances of both models and also highlighted the impact of the OpenLA library on the analysis. We learned that GPT-4, when provided with data pre-processed by OpenLA, demonstrates the best results for the analysis. It was also discovered that all the systems work best with level 1 analysis of students and interactions use and worst with level 3 analysis of generating predictive models. A potential reason for this result could be the difference between the approach taken by the agent for these two analyses. Since level 1 is associated more with simpler calculations and statistics, the agent does not run into errors frequently as compared to level 3 analysis. In the case of generating predictive models in level 3, many algorithms and functions are applied, resulting in more errors and stoppage of agent due to time or iteration limit. Furthermore, we drew many significant relationships from students' data, which are helpful for studying the impact of factors on students' performances. Finally, we were also able to uncover the analysis step-by-step with the help of LangChain's agent. The study of this analysis can be very important for the development of educational applications in the future. The chronological procedure of data analysis provided by the agent can be very helpful for the learning of beginner data scientists. Furthermore, many applications can be developed to improve students' learning experiences by giving them insights into their daily learning routines. The analysis can be further enhanced by integrating the memory chain into the LLM using the LangChain framework. This will allow further room for prompt engineering in the cases where LLM does not understand the prompt at first.

6. Acknowledgements

References

[1] Han, J., Kamber, M., & Computer, P. (2012). Data mining: concepts and techniques. Elsevier/Morgan Kaufmann.

[2] bin Mat, U., Buniyamin, N., Arsad, P. M., & Kassim, R. (2013, December 1). An overview of using academic analytics to predict and improve students' achievement: A proposed proactive intelligent intervention. IEEE Xplore.

[3] Nedeva, V., & Pehlivanova, T. (2021). Students' Performance Analyses Using Machine Learning Algorithms in WEKA. IOP Conference Series: Materials Science and Engineering, 1031(1), 012061.

[4] Baloian, N., Cobaise, J., Peñafiel, B., Majumdar, R., & Ogata, H. (2023). Applying an interpretable and accurate model to Learning analytics. DC@LAK23 Workshop. (In press).

[5] Eman Alasadi, & Baiz, C. R. (2023). Generative AI in Education and Research: Opportunities, Concerns, and Solutions. Journal of Chemical Education, 100(8), 2965–2971.

[6] Gozalo-Brizuela, R., & Garrido-Merchán, E. C. (2023, June 14). A survey of Generative AI Applications. ArXiv.org.

[7] Ulla, M. B., Perales, W. F., & Stephenie Ong Busbus. (2023). "To generate or stop generating response': Exploring EFL teachers" perspectives on ChatGPT in English language teaching in Thailand. Learning, 1–15.

[8] Ali, F., Choy, D., Divaharan, S., Hui Yong Tay, & Chen, W. (2023). Supporting self-directed learning and self-assessment using TeacherGAIA, a generative AI chatbot application: Learning approaches and prompt engineering. Learning, 1–13.

[9] OpenAI. (2023). GPT-4 Technical Report. ArXiv (Cornell University).

[10] Ye, J., Chen, X., Xu, N., Zu, C., Shao, Z., Liu, S., Cui, Y., Zhou, Z., Gong, C., Shen, Y., Zhou, J., Chen, S., Gui, T., Zhang, Q., & Huang, X. (2023). A Comprehensive Capability Analysis of GPT-3 and GPT-3.5 Series Models. ArXiv (Cornell University).

[11] Murata, R., Minematsu, T., Shimada, A. (2020). OpenLA: Library for Efficient E-book Log Analysis and Accelerating Learning Analytics. In International Conference on Computer in Education (ICCE 2020), 301-306.

[12] Ogata, H., Oi, M., Kousuke Mohri, Okubo, F., Shimada, A., Yamada, M., Wang, J., & Hirokawa, S. (2017). Learning Analytics for E-Book-Based Educational Big Data in Higher Education. 327–350.

[13] Oguzhan Topsakal, & Tahir Cetin Akinci. (2023). Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast. 1(1), 1050–1056.

[14] Agents | 🦜🔗 Langchain. (n.d.). Python.langchain.com. https://python.langchain.com/docs/modules/agents/

[15] Pandas Dataframe | 🦜🔗 Langchain. (n.d.). Python.langchain.com. Retrieved December 11, 2023, from https://python.langchain.com/docs/integrations/toolkits/pandas

[16] Dwivedi, S. K., & Rawat, B. (2015, October 1). A review paper on data preprocessing: A critical phase in web usage mining process. IEEE Xplore.