AI-Powered Personalised Learning Plans for Intelligent Tutoring Systems

Tanvi Shah[1], Milind Bhushan[2], Naman Vyas[3], Rajat Shingate[4]
Department of Computer Engineering, Sinhgad Institute of Technology, Lonavala

Abstract: Intelligent Tutoring Systems (ITS) are a promising way to personalize and adapt learning. They use educational theory and artificial intelligence to give students individualized instruction and support. This study examines the practical aspects of ITS implementation, emphasizing important factors, difficulties, and recommended procedures. The paper begins by outlining the design concepts that guide the deployment of ITS, highlighting the significance of personalized learning, proactive participation, prompt feedback, and flexibility. The first section of the article outlines the design principles that direct the implementation of ITS, emphasizing the value of proactive engagement, timely feedback, individualized learning, and adaptability. The paper also discusses how to include instructional methodologies and domain-specific expertise into the design of ITS, emphasizing the need of matching pedagogical approaches with educational goals.
Keywords: Intelligent Tutoring Systems (ITS), Personalization, Educational Theory, Artificial Intelligence (AI), Prompt Feedback, Instructional Methodologies, Domain Specific Expertise, pedagogical approaches.

I. INTRODUCTION

One possible method for customizing and adjusting learning is through Intelligent Tutoring Systems (ITS). They provide pupils with tailored training and support by utilizing artificial intelligence and educational philosophy. This study looks at the practical aspects of ITS installation, highlighting key elements, challenges, and suggested practices. The first section of the article outlines the design principles that direct the implementation of ITS, emphasizing the value of proactive engagement, timely feedback, individualized learning, and adaptability. The article's first section describes the design concepts that guide the application of ITS, highlighting the importance of prompt feedback, individualized learning, adaptability, and proactive engagement. The article also covers the integration of domain-specific knowledge and instructional methodologies into ITS design, stressing the importance of aligning pedagogical strategies with learning objectives.

II. SYSTEM IMPLEMENTATION PLAN

Intelligent Tutoring System, as the name implies, is a system centered around AI and its capabilities. AI-based intelligent tutoring systems enable the creation of personalized learning experiences that offer immediate instruction and feedback to learners, typically without the need for human intervention.

A. Simplified Structure

Intelligent tutoring systems (ITSs) comprise four fundamental components, established through consensus. The essential components of an ITS include:

1) *The Domain Model:* Also referred to as the cognitive model or expert knowledge model, is constructed based on a learning theory, such as the ACT-R theory, which aims to encompass all potential steps necessary to solve a problem. This model encompasses the concepts, rules, and problem-solving strategies within the domain targeted for learning. It serves multiple functions, acting as a wellspring of expert knowledge, a benchmark for assessing a student's performance, or for identifying errors, among other roles.

2) *The Student Model:* Can be visualized as an additional layer atop the domain model, and it serves as the fundamental element of an ITS, with a strong focus on the cognitive and affective states of the student and how they evolve as the learning process unfolds. During the student's progression through their problem-solving journey, an ITS employs a process known as model tracing. Whenever the student model diverges from the domain model, the system detects or highlights an error occurrence. In contrast, within constraint-based tutors, the student model is presented as an overlay on the constraint set. Constraint-based tutors assess the student's solution against the constraint set, distinguishing between satisfied and violated constraints. If any constraints are violated, the student's solution is deemed incorrect, and the ITS offers feedback concerning those constraints. Constraint based tutors provide not only negative feedback (addressing errors) but also positive feedback.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue V May 2024- Available at www.ijraset.com

3) *The Tutoring Model:* receives input from both the domain and student models 3and makes decisions regarding tutoring strategies and actions. Throughout the problem-solving process, the learner can request guidance on their next steps, considering their current position within the model. Furthermore, the system identifies deviations from the production rules of the model and offers timely feedback to the learner. This results in a quicker path to proficiency in the targeted skills. The tutoring model may encompass numerous production rules, each existing in one of two states: learned or unlearned. Whenever a student successfully applies a rule to a problem, the system updates a probability estimate that the student has acquired the rule. The system continues to provide students with exercises that necessitate the effective application of a rule until the probability of having learned the rule reaches at least 95%. Knowledge tracing monitors the learner's advancement as they move from one problem to another, constructing a profile that highlights strengths and weaknesses in relation to the production rules. In the context of the cognitive tutoring system, this information is presented as a skillometer, a graphical representation of the learner's proficiency in the various skills associated with solving algebra problems. Whenever a learner seeks a hint or an error is detected, the knowledge tracing data and the skillometer are dynamically updated in real-time.

4) *The user Interface:* component harmoniously incorporates three essential types of information crucial for conducting a dialogue: knowledge about patterns of interpretation (for understanding a speaker) and action (for generating utterances) within dialogues; domain-specific knowledge required for conveying content effectively; and the knowledge necessary for conveying intent.
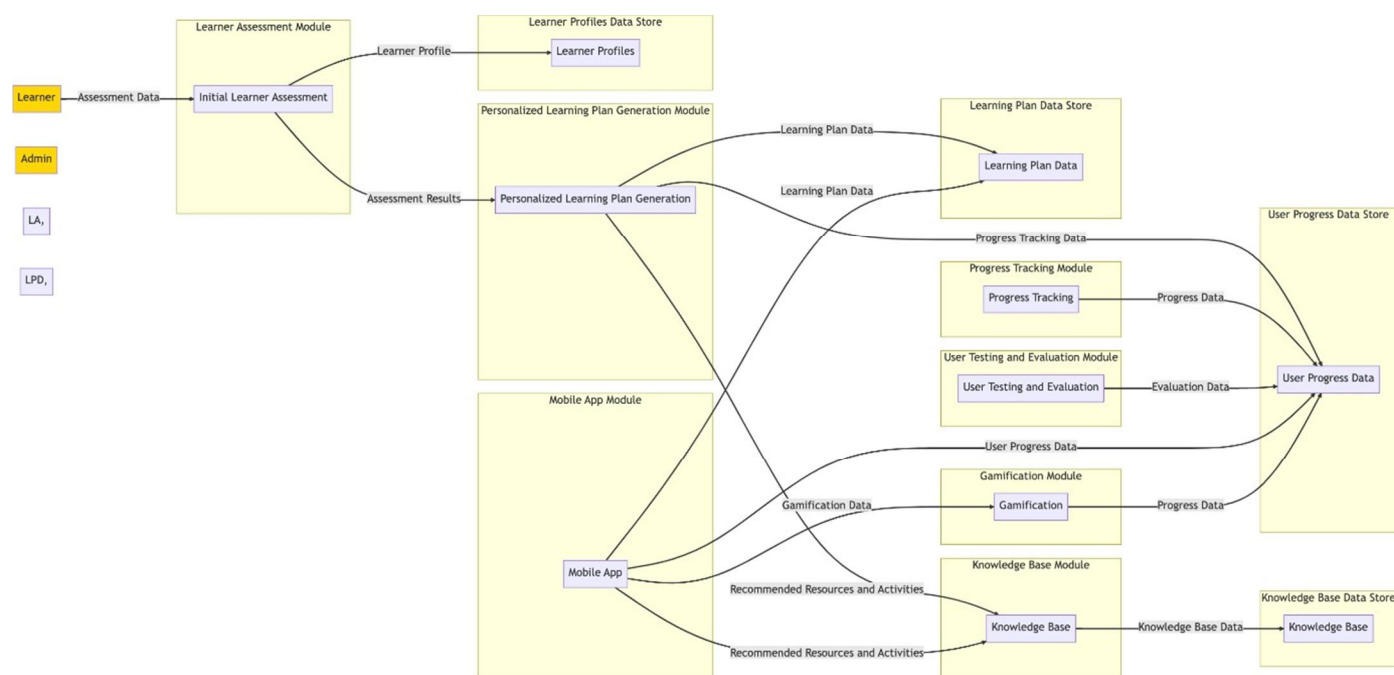

Fig 1: System Implementation Plan

Fig 2: System Architecture

III. SYSTEM WORKING

The implementation of ITS for personalized learning fosters a student-centric approach to education, emphasizing the importance of continuous progress monitoring, customized support, and data-driven insights for teachers. Through the utilization of data analytics made possible by ITS, teachers can identify learning gaps, make well-informed instructional decisions, and provide targeted interventions to enhance student achievement.

While there are numerous advantages to employing ITS for personalized learning, it is crucial to acknowledge the potential drawbacks and challenges that may accompany its implementation. These challenges encompass technical issues, the necessity for adequate educator training and support, as well as concerns related to security and privacy.

An Intelligent Tutoring System (ITS) is a computer-based learning tool that gives students individualised and flexible training. Artificial intelligence (AI) and machine learning algorithms are used by ITS to determine each student's unique learning needs, give individualised education, and offer feedback. The general operation of an intelligent tutoring system is as follows:

1) *Evaluation of Students:* ITS starts with a knowledge, skill, and learning preference assessment of the student. Pre-tests, quizzes, or diagnostic tasks may be used in this assessment to determine the student's present level of subject-matter competency.

2) *Modelling Users:* The evaluation findings are used by the system to generate a user model. The learner's progress, learning style, skills, and limitations are all included in this user model. Personalised teaching is based on the user model.

3) *Selection of Content:* The uses the user model to guide the selection of educational materials. The system determines which ideas or abilities the student needs to concentrate on and adjusts the lesson plan appropriately. Text, multimedia, simulations, and interactive activities are examples of content.

4) *Personalised Education:* ITS provides adaptive training, which modifies the learning activities' pacing and complexity in response to the learner's progress. The system could graduate to more difficult subjects if a learner shows that they have mastered a particular idea. On the other hand, the system can offer extra help and remediation if a kid needs it.

5) *Comments and Evaluation:* The student receives instant, tailored feedback from the system. This feedback may consist of explanations, corrections, and recommendations for enhancements. The system updates the user model and improves the teaching strategy with the aid of ongoing assessment.

6) *Engagement and Interactivity:* To keep students interested, ITS frequently includes interactive components. This can include games, virtual laboratories, simulations, and other interactive exercises meant to enhance learning and add interest to the learning process.

7) *Observation and Progress Tracking:* The learner's progress is tracked over time by the system. It records accomplishments in learning, performance indicators, and completed tasks. Instruction is modified and the user model is further improved with the usage of this data.

8) *NLP Natural Language Processing:* Certain ITS systems make use of natural language processing to comprehend student input and reply more naturally. More organic communication between the learner and the system is made possible by this capacity.

9) *Machine Learning and Data Analysis:* To increase its efficacy, the ITS examines a lot of data produced by student interactions. Algorithms for machine learning may be able to spot trends in student performance and eventually assist in improving teaching methods.

10) *Combining Instruction in the Classroom: I*n certain instances, ITS is combined with conventional classroom instruction, giving teachers the ability to track students' progress, gain understanding, and adjust their training according to data provided by ITS.

These 10 components are the main factors which help the Intelligent Tutoring System to operate and function according to the user's necessity.

IV. ALGORITHM USED

Large Language Models: With training from enormous volumes of textual data, Large Language Models (LLMs) are robust artificial intelligence systems that can comprehend and produce language like that of humans. These models, which use deep learning approaches to process and generate text, include OpenAI's GPT (Generative Pre-trained Transformer) series. They demonstrate impressive abilities in tasks including summarizing, translation, natural language interpretation, text production, and more. The capacity of LLMs to comprehend intricate linguistic patterns, context, and semantics enables them to produce logical and contextually appropriate answers to input material. They are widely used in many different fields, such as information retrieval, language translation, conversational agents, and content creation. LLMs offer a breakthrough in natural language processing technology as they develop further, and they can completely transform information processing, education, and communication.

Large Language Models (LLMs) offer sophisticated natural language creation and interpretation capabilities, which greatly improve Intelligent Tutoring Systems (ITS). With the use of these models—like OpenAI's GPT (Generative Pre-trained Transformer) series—ITS can understand text and provide responses that are human-like, enabling more organic and interesting interactions with students. LLMs give ITS the capacity to provide individualized feedback, respond to inquiries, clarify ideas, and hold dialogue tutoring sessions, all of which increase the system's overall efficacy and flexibility. Additionally, LLMs may help create educational resources, create interactive exercises, and evaluate student feedback, all of which can help create ITS that are stronger and more adaptable. Teachers can use natural language processing to create immersive, individualized learning experiences that meet the different requirements of their students by incorporating LLMs into ITS.

V. WORKING OF THE ALGORITHM

One of the most important strategies for improving machine language intelligence is language modelling (LM). LM often entails modelling word sequence probability to forecast future likelihood. [1]
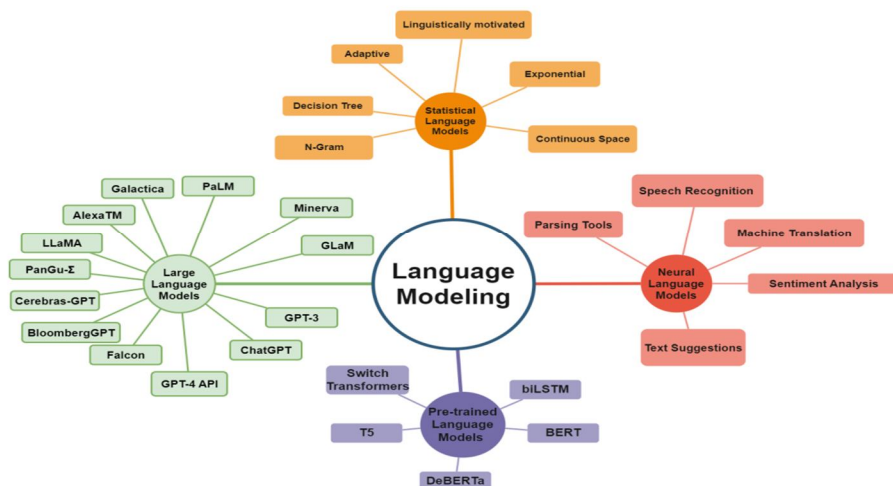

Fig 3: Types of language Modelling

Sophisticated artificial intelligence systems known as Large Language Models (LLMs) are constructed using Transformer architectures and trained on enormous text datasets from the internet. They pick up on the subtleties of language structure, semantics, and context through pre-training, which teaches them to anticipate the next word in a sequence given the previous context. Their talents are further refined by honing in on activities. Upon input, text is tokenized and encoded into numerical representations, which are then handled by feed-forward neural networks and layers of self-attention processes. This makes it possible for the model to accurately represent long-range relationships and assess the significance of each token. Text that is logical and pertinent to the situation is produced by decoding. Natural language production and comprehension skills have been shown by LLMs in a variety of applications, including as chatbots, language translation, and content creation.
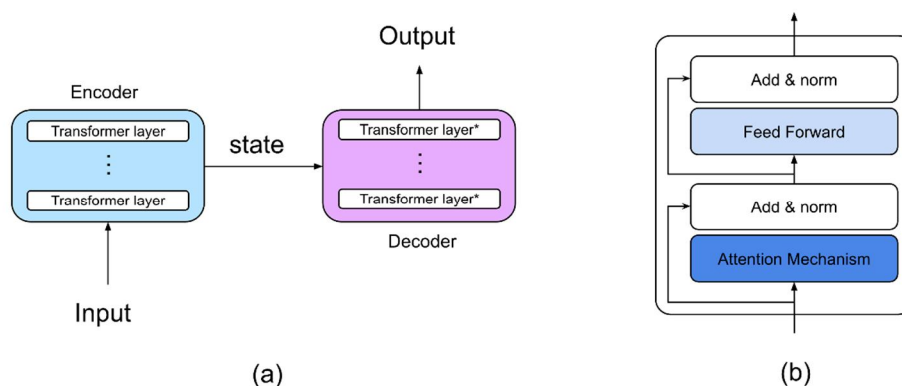
Fig 4: Simplified Architecture of LLM

Usually, LLMs are constructed on top of the Transformer neural network architecture. Text and other sequential data types are especially well-suited for the Transformer design. It is comprised of feed-forward neural networks and many layers of self-attention processes. Large volumes of textual data gathered from the internet are used to train LLMs. Books, essays, webpages, and other textual materials can be included in this data. The vast amount and variety of this data are necessary for the model to pick up on language's subtleties. Large-scale corpora of text data are utilised for pre-training LLMs before they are applied to tasks. The model gains the ability to anticipate the following word in a phrase based on the previous context during pre-training. Through this procedure, the model is better able to extract syntactic, semantic, and contextual data from the text.

Following pre-training, LLMs can be improved for activities or domains. To fine-tune the model to execute a specific task, such text synthesis, translation, summarization, etc., it must be trained on a smaller, task-specific dataset. The Input of the LLM first undergoes tokenization. In tokenization, the text is divided into smaller pieces known as tokens, which might be single words, groups of words, or characters. Every token has a special number identification allocated to it. Following tokenization, an embedding layer is used to transform the tokens into numerical representations. Each token is mapped by this procedure to a high-dimensional vector that the model may use. The contextual and semantic information of the tokens is captured by these vectors.

The self-attention mechanism is a crucial part of the Transformer design. By using this approach, the model can determine each token's relative value to the other tokens in the input sequence. It facilitates the model's efficient acquisition of contextual data and long-range relationships. Transformers have feed-forward neural networks in addition to self-attention layers. To produce the desired result, these networks process the data that is gathered by the self-attention mechanism and apply non-linear adjustments to it. A decoding technique is used to produce the output once the input text has been processed via the model's layers. Using the context given by the input text and the previously created tokens, this method creates one token at a time. Until an end-of-sequence token is created or the maximum length is achieved, the procedure keeps going. Depending on the job, the model may provide different outputs. The result of text generation jobs is usually a list of words that combine to create sentences that make sense and are relevant to the situation. The output for other tasks, such as translation or summarization, may seem different, but it still tries to represent the sense of the original text.
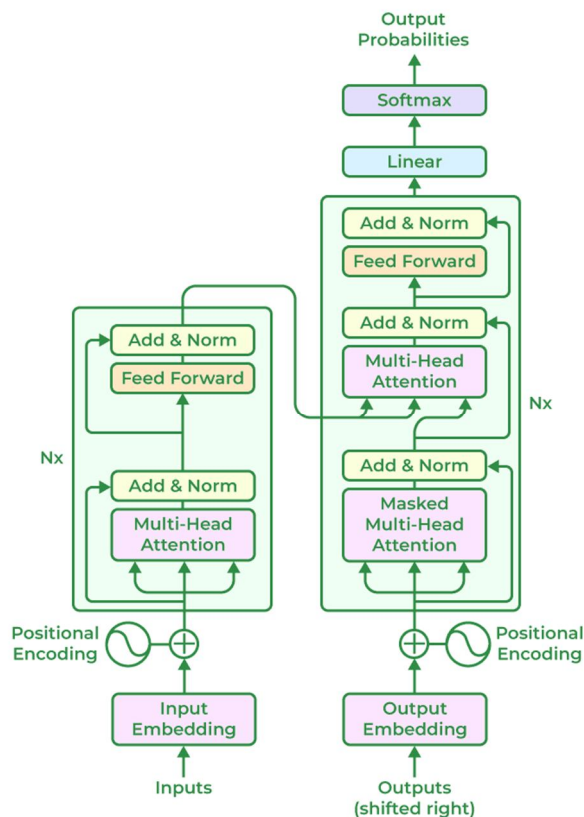
Fig 5: Transformer-based Architecture of LLM

VI. IMPLEMENTATION AND RESULTS

Suppose a user enters the system. To him, the system would look like this:
The system would ask his name, followed by age and gender thereafter.
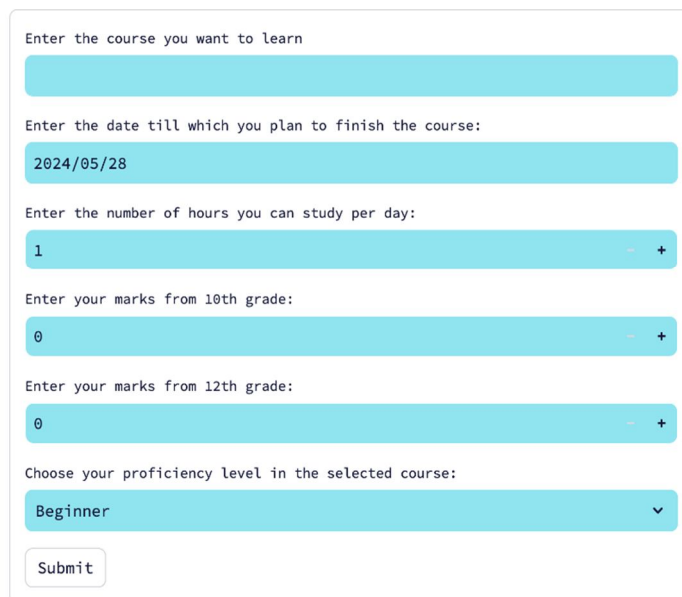

Fig 6: Welcome Page of ITS

Then, coming to the part where the user needs to choose the preferred course, the duration followed by the hours investing during the course, the marks of 10th, 12th and the proficiency level of the respective user. The following GUI would be visible as:
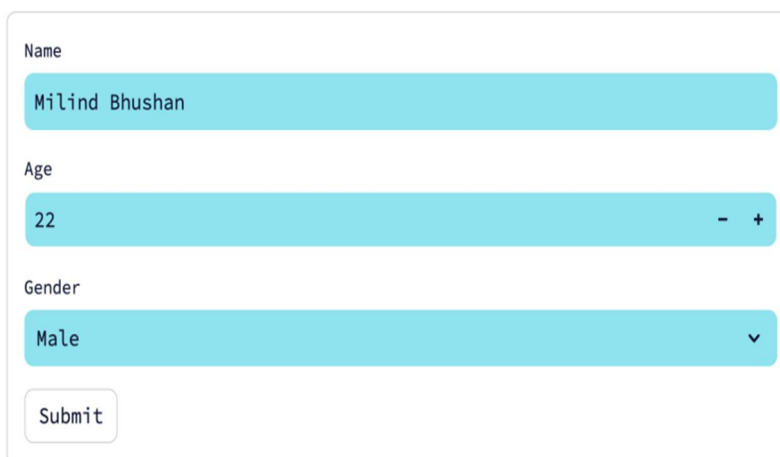
Fig 7: ITS GUI for user credentials

The successful implementation is shown below:

Suppose a user, named Milind Bhushan wants to learn the courses in his own personalised way. The user enters his details and furthermore, chooses the course.
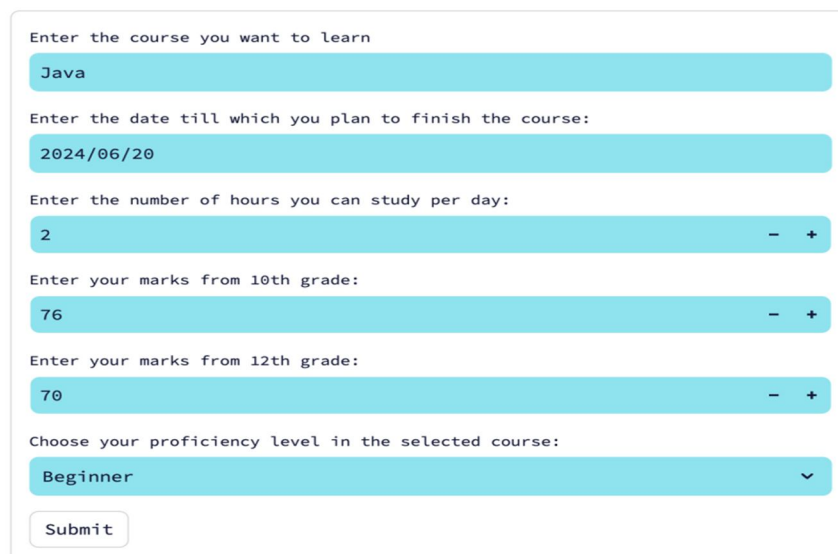

Fig 8: Welcome page for user

Similarly, as mentioned – the user needs to fill in the details for the course he will be choosing.

Fig 9: The GUI after the user enters his details in ITS

The system then tries to generate the learning plan for the user.

Fig 10: Personalized plan being generated

Here is the plan the Intelligent Tutoring System has generated for user.

Day 1

- Topic: Introduction to Java, Variables, and Data Types
- Activities:
- Read Chapter 1 of textbook
- Complete exercises on variables and data types

Day 2

- Topic: Operators and Control Flow
- Activities:
- Read Chapter 2 of textbook
- Practice using operators and conditional statements

Day 3

- Topic: Arrays and Strings

- Activities:

- Read Chapter 3 of textbook

- Code examples using arrays and strings

Day 4

- Topic: Object-Oriented Programming (OOP) Concepts

- Activities:

- Read Chapter 4 of textbook

- Understand OOP principles like classes, objects, and inheritance

Day 5

- Topic: Classes and Objects

- Activities:

- Code examples using classes and objects

- Practice instantiating and manipulating objects

Day 6

- Topic: Inheritance and Polymorphism

- Activities:

- Read Chapter 5 of textbook

- Understand inheritance and polymorphism

- Code examples using inheritance

Day 7

- **Topic:** Exceptions and Error Handling
- **Activities:**
- Read Chapter 6 of textbook
- Learn about exceptions and how to handle them

Day 8

- **Topic:** Data Structures: Lists, Stacks, and Queues
- **Activities:**
- Read Chapter 7 of textbook
- Implement data structures using Java

Day 9

- **Topic:** Algorithms: Searching and Sorting
- **Activities:**
- Read Chapter 8 of textbook
- Understand searching and sorting algorithms
- Code examples using algorithms

Day 10

- **Topic:** Input/Output and File Handling
- **Activities:**
- Read Chapter 9 of textbook
- Learn how to read and write to files

Day 11

- **Topic:** Java Collections Framework
- **Activities:**
- Read Chapter 10 of textbook
- Understand Java collections and use them in code

Day 12

- **Topic:** Advanced OOP Concepts
- **Activities:**
- Read Chapter 11 of textbook
- Learn about advanced OOP topics like generics and design patterns

Day 13

- **Topic:** Java Development Tools
- **Activities:**
- Learn about Java development tools like IDEs (Eclipse, IntelliJ)
- Set up a development environment

Day 14

- **Topic:** Java Projects
- **Activities:**
- Work on a small Java project
- Implement Java concepts in a practical setting

Day 15

- **Topic:** Review and Assessment
- **Activities:**
- Review key concepts covered so far
- Take a practice quiz or exam

Day 16 onwards:

- Repeat the above topics with increasing complexity and difficulty, as the student progresses.

- Provide additional support and guidance as needed in areas where the student struggles.

- Encourage the student to explore advanced topics and resources to enhance their learning.

Additional Resources:

- Java Tutorials: https://docs.oracle.com/javase/tutorial/

- Java for Beginners: https://www.coursera.org/learn/java-programming

- Java Programming (edx): https://www.edx.org/course/java-programming

Strategies for Extra Attention:

- Conduct regular assessments to identify areas where the student needs additional support.

- Provide individualized tutoring sessions to clarify concepts and address specific challenges.

- Assign additional exercises and projects to reinforce understanding.

- Create a study group or connect with peers for collaborative learning.

Here is the successful implementation of Intelligent Tutoring System, for a user. The plan may vary according the user and his inputs in the ITS.

VII. CONCLUSION

In summary, the proposed AI-based intelligent web tutoring system has the potential to revolutionize how students engage with and access instructional materials. This system can provide personalized learning objectives, recommend resources and activities, and assess progress in a way tailored to each student's needs, addressing the limitations of existing tutoring systems. This level of customization will not only motivate students to pursue further studies but also enhance the system's effectiveness.

Furthermore, there is an opportunity that the proposed approach can benefit students, teachers, and society. This approach can help bridge the gap between those who have access to high-quality education and those who do not by offering individualized and accessible instruction. Moreover, the implementation of ITS for personalized learning fosters a student-centric approach to education, emphasizing the importance of continuous progress monitoring, customized support, and data-driven insights for teachers. Through the utilization of data analytics made possible by ITS, teachers can identify learning gaps, make well-informed instructional decisions, and provide targeted interventions to enhance student achievement.

While there are numerous advantages to employing ITS for personalized learning, it's crucial to acknowledge the potential drawbacks and challenges that may accompany its implementation. These challenges encompass technical issues, the necessity for adequate educator training and support, as well as concerns related to security and privacy.

Considering these factors, educators, administrators, and technology specialists should collaborate to ensure the effective utilization and sustainable deployment of these systems as they continually enhance and integrate personalized learning via ITS. By addressing these challenges and optimizing personalized learning, ITS can play a pivotal role in revolutionizing the field of education and equipping students with the essential skills and knowledge needed to thrive in a dynamic global environment.

REFERENCES

[1] Hadi, Muhammad Usman, Rizwan Qureshi, Abbas Shah, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, and Seyedali Mirjalili. "A survey on large language models: Applications, challenges, limitations, and practical usage." Authorea Preprints (2023).

[2] Butz, Cory J., Shan Hua, and R. Brien Maguire. "A web-based intelligent tutoring system for computer programming." In IEEE/WIC/ACM International Conference on Web Intelligence (WI'04), pp. 159-165. IEEE, 2004.

[3] Paladines, José, and Jaime Ramirez. "A systematic literature review of intelligent tutoring systems with dialogue in natural language." IEEE Access 8 (2020): 164246-164267.

[4] Lin, Chien-Chang, Anna YQ Huang, and Owen HT Lu. "Artificial intelligence in intelligent tutoring systems toward sustainable education: a systematic review." Smart Learning Environments 10, no. 1 (2023): 41.

[5] Bewersdorff, Arne, Kathrin Seßler, Armin Baur, Enkelejda Kasneci, and Claudia Nerdel. "Assessing student errors in experimentation using artificial intelligence and large language models: A comparative study with human raters." Computers and Education: Artificial Intelligence 5 (2023): 100177.

[6] Kochmar, Ekaterina, Dung Do Vu, Robert Belfer, Varun Gupta, Iulian Vlad Serban, and Joelle Pineau. "Automated personalized feedback improves learning gains in an intelligent tutoring system." In Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21, pp. 140-146. Springer International Publishing, 2020.

[7] Yadav, Gautam, Ying-Jui Tseng, and Xiaolin Ni. "Contextualizing Problems to Student Interests at Scale in Intelligent Tutoring System Using Large Language Models." arXiv preprint arXiv:2306.00190 (2023).

[8] Cavanagh, Thomas, Baiyun Chen, Rachid Ait Maalem Lahcen, and James R. Paradiso. "Constructing a design framework and pedagogical approach for adaptive learning in higher education: A practitioner's perspective." International review of research in open and distributed learning 21, no. 1 (2020): 173-197.

[9] Lopes, Arcanjo Miguel Mota, and José Francisco De Magalhães Netto. "Designing Pedagogical Agents Toward the Recommendation and Intervention Based on Students' Actions in an ITS." In 2021 IEEE Frontiers in Education Conference (FIE), pp. 1-8. IEEE, 2021.

[10] Woods, Kathryn. "The development and design of an interactive digital training resource for personal tutors." In Frontiers in Education, vol. 5, p. 100. Frontiers Media SA, 2020.

[11] Chen, Yulin, Ning Ding, Hai-Tao Zheng, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. "Empowering Private Tutoring by Chaining Large Language Models." arXiv preprint arXiv:2309.08112 (2023)

[12] Nye, B., Dillon Mee, and Mark G. Core. "Generative large language models for dialog-based tutoring: An early consideration of opportunities and concerns." In AIED Workshops. 2023.

[13] Ramesh, Vyshnavi Malathi, N. J. Rao, and Chandrashekar Ramanathan. "Implementation of an intelligent tutoring system using Moodle." In 2015 IEEE Frontiers in Education Conference (FIE), pp. 1-9. IEEE, 2015.

[14] Choi, HeeSun, Cindy Crump, Christian Duriez, Asher Elmquist, Gregory Hager, David Han, Frank Hearl et al. "On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward." Proceedings of the National Academy of Sciences 118, no. 1 (2021): e1907856118.

[15] Lai, Jennifer WM, and Matt Bower. "How is the use of technology in education evaluated? A systematic review." Computers & Education 133 (2019): 27-42.

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 ◯ (24*7 Support on Whatsapp)