Creating Dynamic Learning Environments:

A Case for Adaptive E-Learning

Pallavit Aggawral[1]

[1] Trinity College Dublin, Dublin, Ireland
MSc Intelligent Systems 2022-2023

Abstract. Adaptive e-learning systems using learning styles have been shown to help learners more effectively by accommodating their individual needs and preferences. With the advent of new technologies, these systems are becoming more widely available on various platforms and applications. This paper proposes an adaptive e-learning system that models a student's learning style and prior knowledge to tailor the learning experience. The system also incorporates user behavior and motivation into its adaptation process to better predict the user's expectations of the kind of questions from the system. To achieve this, the system give tests to the user with questions based on their level of difficulty and updates the skill schema based on the user's performance. The proposed system combines learning theories and pedagogical theories to be used more efficiently at all levels of education. In addition, the system prompts GPT engine to generate an ever-growing domain of question base that is tagged with the right ontological tags, allowing for dynamic question adaptation based on student performance. The proposed adaptive e-learning system effectively models a student's learning style and prior knowledge, while also taking into account their behavior and motivation to create a more personalized and effective learning experience. Future work includes incorporating user scrutability and meta-cognition aspects into the system to provide more flexibility and to introduce additional tuning mechanisms for better question adaptation.

Keywords: Adaptive e-learning, Adaptive Applications, Learning styles, Student modeling, Ontology-based frameworks, Prompt Engineering

1 Introduction

Education has been structured with a “one size fits all” approach where learners have to conform to a standardized curriculum without considering their individual learning styles. However, this approach has led to learners struggling to acquire knowledge, leading to a different direction towards personalized learning. Personalized learning offers a learning system that is adapted to each student’s needs and learning style, which increases motivation and potential based on their skills. Digital platforms and educational systems that provide access to personalized learning have gained popularity due to advances in technology and the wide usage of networks. Learning Styles have become one of the most popular methods for personalized learning as they consider learners’ learning preferences. However, traditional methodologies have limitations, and applications that capitalize on technological platforms have restrictions in classroom activities. This paper discusses the development of an adaptive e-learning application that utilizes learning styles and machine learning techniques to offer personalized learning. The application models the user’s motivation and interaction metrics and uses GPT to generate a growing domain of questions. The results show that the combination of learning styles and prior knowledge gave better results in adaptation, and user behavior plays a vital role in adapting the model, making the system more accurate in determining the next set and type of questions. The application can be used to provide personalized and adapted lessons to students based on their preferences and needs.

2 Research Questions

A few of the research questions that naturally emerged during the runtime of this course were:

1) Which learning style model and prediction techniques should be used in adaptive e-learning systems?
2) What different factors can be used to effectively model a user?
3) How does a system adapt, what is adaptiveness, and what techniques do we use to build this feature for our e-learning system?
4) How do we define an effective domain? How do we create a knowledge base that is scalable and adaptive?

Through implementing an adaptive eLearning system, we developed a sense of what gives an adaptive application its ability to transform according to user behavior and what are the different techniques to think about adaptability in terms of system design.

3 Literature Review & Background

3.1 Adaptive e-learning systems

The World Wide Web and information technologies have been instrumental in diversifying the educational process, particularly within universities where the focus has shifted towards more personalized learning [1]. These platforms customize content, presentation methods, and difficulty levels based on each learner's needs, resulting in uniquely tailored lessons for every student [2]. Learning platforms typically organize their content based on Learning Styles, as research from educational psychologists indicates that these styles significantly impact the learning process [3].

Advancements in technology have facilitated the development of systems that integrate Learning Styles, subsequently guiding and supporting learners. Some of these systems include Learning Management Systems (LMS), Adaptive Hypermedia Systems (AHS), and Intelligent Tutoring Systems (ITS). In recent years, Learning Style Based Adaptive Educational Systems (LSAES) have emerged as well. These systems differ from their predecessors in that they create personalized learning environments based on learning styles, recommending the most appropriate methods and techniques for each student [4].

Incorporating Learning Styles into the educational process has become increasingly vital and practical. There are three primary domains within the Personalized E-Learning Architecture: (1) User module: Storing and managing user profiles; (2) Adaptation module: Analyzing the best learning objects; (3) Domain module: Storing and managing learning objects.

Studies indicate that Adaptive E-learning Systems based on Learning Styles enhance efficiency, reduce learning time, increase subject completion rates, and improve academic performance. New methods have emerged to offer learners a more reliable and high-performing personalized learning experience. Some of the most notable systems include:

- **AHS (Adaptive Hypermedia Systems):** These systems provide a personalized user experience through the use of hypertexts and hypermedia.

- **AEHS (Adaptive Educational Hypermedia Systems):** Created to support learners in their education, these systems focus on individual needs and adapt learning materials based on prior knowledge and learning objectives set by a tutor. This approach allows the system to deliver a learning experience that best suits each learner's needs.

- **AES (Adaptive Educational Systems):** Distinguished from the systems mentioned above, these systems incorporate the Learning Style factor. Based on educational theories, these systems adapt the navigation and materials provided to learners according to both their needs and learning preferences. This results in a more engaging and interactive learning experience, ultimately promoting active student participation and increased satisfaction with the learning process.

- **ITS (Intelligent Tutoring System):** Developed to leverage user-generated data, these systems utilize information such as mouse movements, time spent, resulting grades, and types of errors made by users to guide them in the right direction. By analyzing this data, the system can create a profile to deliver suitable content and provide an adaptive learning environment tailored to each use.

3.2 Learning Styles

A crucial initial step in constructing a student model is selecting the appropriate student characteristics to consider. When building a new student model for a specific intelligent tutoring system, the question "What aspects of the student should we model?" must be answered. According to [5], to carry out personalization effectively, the student model needs to consider both domain-dependent and domain-independent characteristics. Some of these characteristics are static, while others are dynamic.

In [7], authors reviewed user models of existing adaptive web-based systems concerning the sources of adaptation and the techniques for user modeling. Their sources of adaptation regarding user individual characteristics include user knowledge, interests, goals and tasks, background, and individual traits. This categorization's individual traits encompass cognitive styles and learning styles, while other individual traits, particularly cognitive abilities and personality, are addressed marginally.

The challenge lies in defining the dynamic student characteristics that form the basis for the system's adaptation to each student's individual needs. These characteristics include knowledge and skills, errors and misconceptions, learning styles and preferences, affective and cognitive factors, and meta-cognitive factors. Knowledge refers to a student's prior understanding of the domain and their current knowledge level. This is typically measured through questionnaires and tests that the student completes during the learning process. Additionally, by administering these tests and observing the student's actions, the system can identify misconceptions.

Many ITSs adopt the Felder–Silverman learning style model (FSLSM) as a proposal for modeling learning styles. FSLSM classifies students into four dimensions: active/reflective, sensing/intuitive, visual/verbal, and sequential/global. Another method for modeling learning styles is the Myers-Briggs Type Indicator (MBTI), which identifies eight categories of learning styles: extrovert, introvert, sensing, intuitive, thinking, feeling, judging, and perceiving.

The initiative to provide adaptivity to learning styles stems from the assumption that matching instructional strategies to learners' learning styles results in improved learning performance. Several studies confirm this hypothesis.

A key observation is that learning systems often adapt to students' cognitive abilities, gender, and meta-cognitive abilities but less frequently to factors such as age, working memory, personality, previous experience, and anxiety levels. This observation suggests a new avenue for adaptiveness. A more comprehensive consideration of individual differences, both theoretical and practical, can be found in the work of [7]. They concluded that cognitive style, gender, working memory, knowledge, and anxiety significantly impact web-based learning. Given the evidence that experienced human tutors monitor and react to students' emotional states to motivate them and enhance their learning process, a tutoring system should interpret students' emotional states and adapt its behavior to their needs, providing appropriate responses to those emotions.

3.3 Student Modelling

Overlay Modelling: One of the most popular and widely used student models is the overlay model, first introduced by Stansfield, Carr, and Goldstein in 1976 and subsequently employed in numerous systems [8]. The core assumption of the overlay model is that a student may possess incomplete but accurate knowledge of a domain. As a result, according to overlay modeling, the student model is a subset of the domain model, which reflects expert-level knowledge of the subject. The discrepancies between the student's and expert's knowledge sets are considered to result from the student's lack of skills and knowledge, with the instructional goal being to minimize these differences as much as possible. In its original form, the pure overlay model assigns a Boolean value, yes or no, to each element, indicating whether the student knows or does not know that element. In its contemporary iteration, the overlay model represents the extent to which a user knows a domain element using either a qualitative measure (good–average–poor) or a quantitative measure, such as the probability that the student is familiar with the concept.

Stereotype Modelling: Stereotyping, introduced by E. Rich in 1979 through the GRUNDY system [9], involves clustering users into groups based on shared characteristics. These groups, or stereotypes, contain common knowledge about a user group. New users are assigned to a stereotype when their characteristics match those in the group. Stereotypes have been used for student modeling in many adaptive and/or personalized tutoring systems, often combined with other user modeling methods. Some examples include INSPIRE, Web-PTV, WELSA, and Wayang Outpost [10], a software tutor that helps students learn to solve standardized-test type questions, such as the Scholastic Aptitude Test and other state-based exams taken at the end of high school in the USA, to discern factors that affect student behavior beyond cognition.

Perturbation: The perturbation student model extends the overlay model by representing the student's knowledge as a combination of both the expert's knowledge subset and misconceptions. Like the overlay model, it considers the learner's subset of expert knowledge but also incorporates their mal-knowledge. The perturbation model replaces correct rules with incorrect ones, leading to the student's answers. Since multiple reasons can contribute to a wrong answer, the system generates discriminating problems to pinpoint the student's incorrect rules. This collection of mistakes, known as a bug library, can be built through empirical analysis of errors (enumeration) or by generating mistakes based on common misconceptions (generative technique).

Machine Learning Techniques: Student modeling involves inferring a student's behavior based on their knowledge level, cognitive abilities, preferences, skills, and aptitudes. Automating the processes of observation and induction in adaptive and personalized tutoring systems can be achieved through machine learning, which focuses on forming models from observations.
Web-EasyMath [11] combines stereotypes with the distance-weighted k-nearest neighbor algorithm for initializing new student models. The student is first assigned to a stereotype category based on their knowledge level. Then, the system initializes all aspects of the student model using the distance-weighted k-nearest neighbor algorithm among students belonging to the same stereotype category as the new student.

Ontology-based student modelling: Recent research has explored the intersection of user modeling and web ontologies, as both disciplines aim to qualitatively model real-world phenomena.
OPAL is an ontology-based framework that offers content presentation and navigation assistance tailored to individual users, adapting specifically to a learner's knowledge and interests in the subject [12].
MAEVIF utilizes a student model based on ontologies and diagnosis rules to make informed tutoring decisions and provide the most suitable feedback to the student at each moment [13].

4 Implementation of eLearning System

4.1 Overview

Inspired by the different adaptive eLearning systems and ITS mentioned above and more, and going through multiple different concepts of user modelling and meta-cognition approaches, we implemented a unique web-based application during our course that models a user by explicitly getting the user's background information, analyzing his/her learning style and calculating user's motivation based on some predefined questions. The system also takes in user performance as an explicit information by doing a diagnostic test, but later uses user's performance and interaction or activity metadata implicitly to fine tune the system. A decision tree classifier model is used to classify the user's difficulty level based on which he/she would be presented with an appropriate question for the test preparation. The information from learning style is used to provide the input with the right kind of questions (for e.g. a word-based problem, over a direct problem), and also the right kind of answers and explanations based on whether the user is more visually receptive or not or other kind of FSLSM attributes of the user. Using these a prompt is created for the GPT engine (OpenAI's chatgpt) which tailors a question a provides it to the interface. We also ask for 5 ontological tags on the questions, and store it in the database for offline use later.

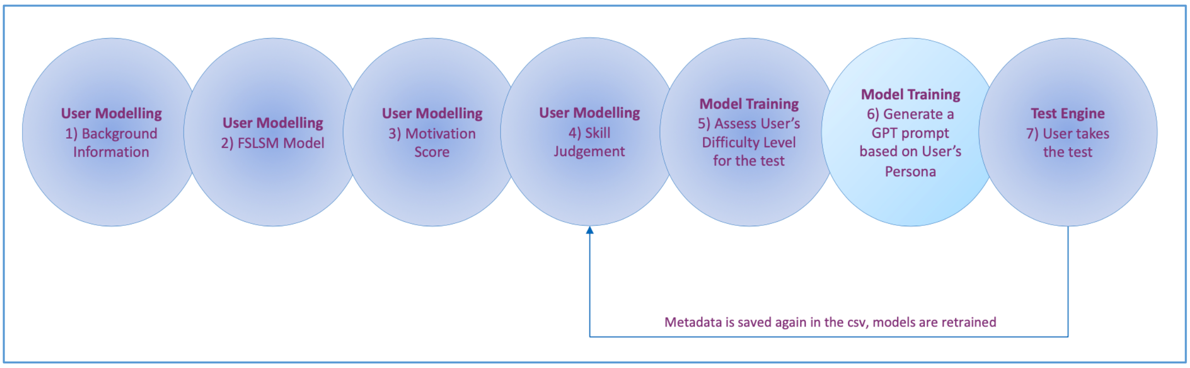

Fig. 1. High level overview of the functioning of the implemented test based system

4.2 Architecture

We used Django for our backend, which enabled us to use python and also provide a web interface and API MVC architecture for our web app. The backend was connected with a MongoDB cluster. To facilitate communication with the frontend we used GraphQL, which acted as a message relay / middleware. Our frontend was in Next.js (React.js). We defined 3 csv files Learning_style.csv, Motivation_Level.csv, Skill_level.csv which had schema definitions to store learning style configurations of the user, his motivation scores, and his/her skill and performance metric standardized out of 10 for each topic supported by the application against the user's system id.

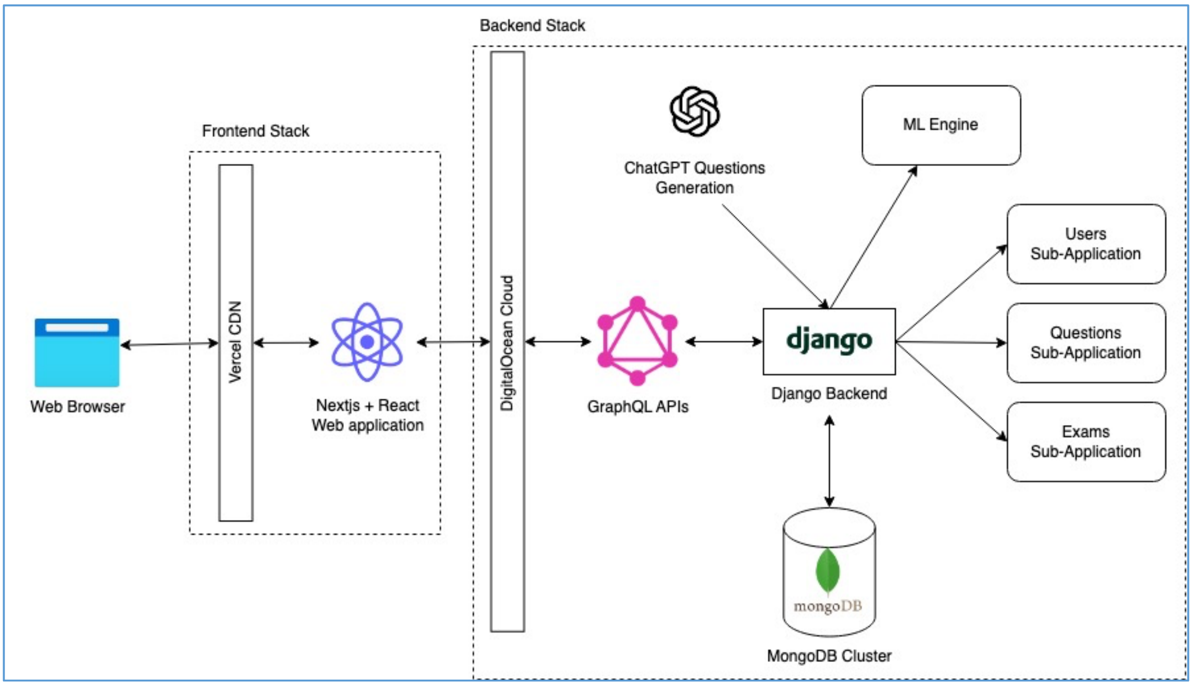

Fig. 2. Architecture overview

4.3 User Modelling

We carry out the user modelling, by first determining the user's FSLSM type. This is determined by asking user the standard Felderman survey questions like: When starting a new project, do you prefer to dive right in or take some time to plan and gather information? , Do you rely on your senses to understand information or do you tend to rely more on intuition and abstract concepts? , Do you find it helpful to use diagrams or other visual aids when learning new information? , Do you find it helpful to discuss and debate ideas with others when learning new information? Etc. The user is also asked questions, which help determine his initial motivations for e.g.: Has user taken GRE before? , If yes, What has the user scored? , When is the user planning to take GRE?, How many times has the user taken it overall? Etc. The motivation score is not calculated based on the user's inputs and a formula normalizes the motivation score from 0-1. Finally a diagnostic test with 10 questions is done, to gauge the user's skills in different topics. The skill is calculated out from 0-10, and the higher the number the better the user is in that topic. It is calculated as score = (topic_weight + (time_weight / time) – (diff_weight * difficulty) + correct_weight * (1 if correct else 0)). The time_weight is divided by the time taken to answer a question to incorporate the time efficiency of the user in the score calculation. By dividing the time_weight by the time taken, we are rewarding users who can answer questions faster while maintaining accuracy. The faster a user answers a question, the more significant the contribution of the time_weight will be to the overall score. The diff_weight is multiplied by the difficulty to account for the level of challenge a question presents in the score calculation. By multiplying the diff_weight by the difficulty, the formula takes into consideration the complexity of the question, giving a higher penalty to the score for more difficult questions.

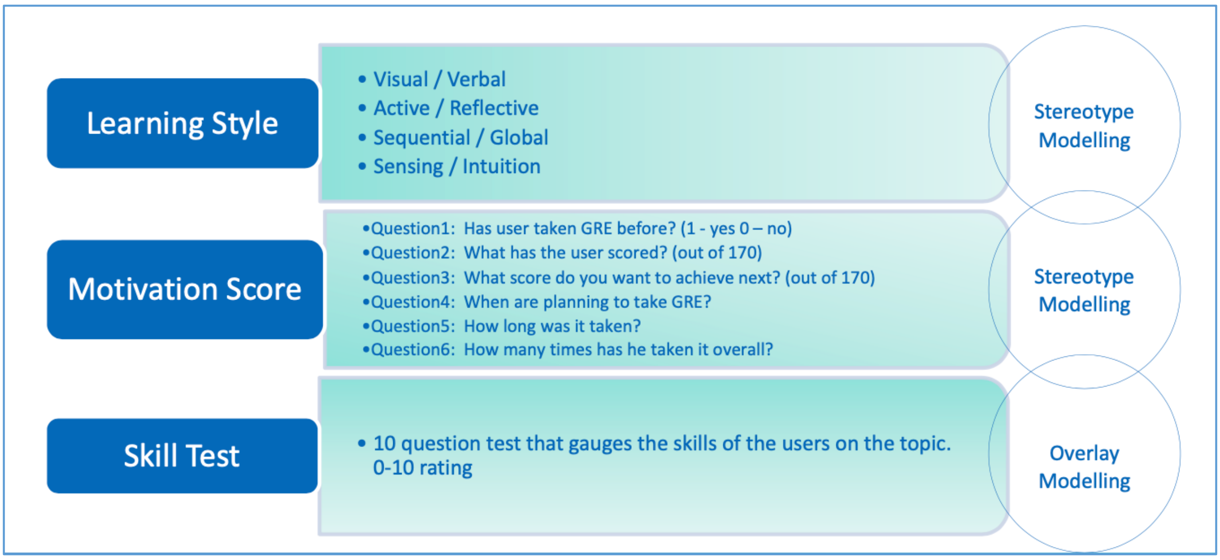

Fig. 3. User Modelling Approach

4.4 Machine Learning Model

1000 randomly generated user data is used to generate the data for the learning_style.csv, motivation_level.csv and skill_level.csv. The csv files are merged together and the FSLSM groups are canonicalized. Now using the skill level scores across 18 topics, motivation scores we map the value to a 3-value difficulty system of "Easy", "Medium" and "Hard". These labels would ultimately serve as the output label for the model that we train it on. A decision tree classifier model is used, with currently no hyperparameter tuning, to take the 1000 rows and learn to output the difficulty level based on the skills and motivation scores. Our model yields a 90% accuracy in predicting the Difficulty label for the user.

4.5 Open-AI

In 2017, the Google Brain research team introduced the "self-attention" mechanism and the Transformer network architecture based on it. The architecture uses an encoder-decoder structure, but instead of focusing on relationships between sequences, self-attention captures relationships within a single sequence through key-value pairs of vectors. OpenAI's Generative Pre-training Transformers (GPT) is the first model pretrained on a large corpus, and it is a decoder-only model. The combination of pre-training and fine-tuning proved to greatly enhance GPT's performance, making it adaptable to multiple tasks while addressing the issue of insufficient labeled data.

The GPT-3 playground is a website designed to help users start using the model. Users can choose from 4 models and 3 modes based on their needs, with "complete" mode being the most frequently used. The "temperature" parameter is often used to control randomness during generation, while other parameters such as Maximum-Length, Top P, presence penalty, and frequency penalty can be used to fine-tune responses from OpenAI's API.

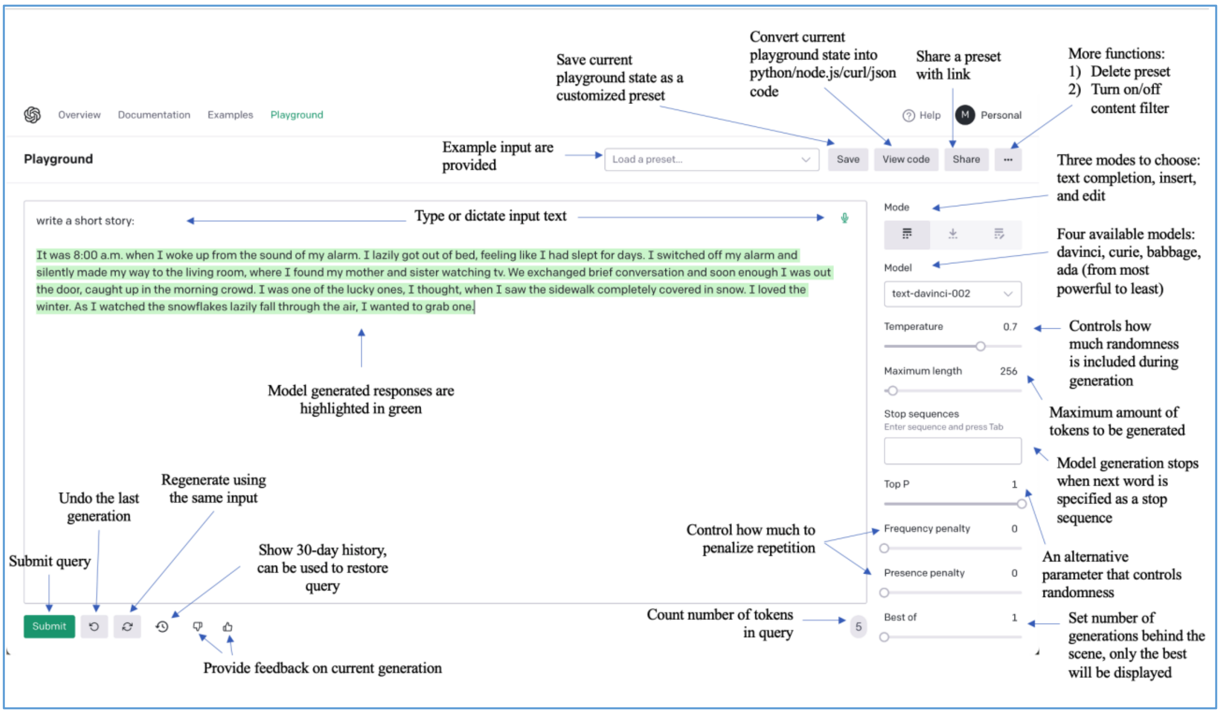

Fig. 4. Open-AI's API web interface

We had to use the following prompt to get the question back in a fixed format. "Structure your response in the following format:

Question: [Your question here]
A) [Option A]
B) [Option B]
C) [Option C]
D) [Option D]

Answer: [Correct answer option, e.g., 'A', 'B', 'C', or 'D']
Ontology Tag: [Choose one from the list: 'Properties of integers, Fractions, decimals, and percents, Ratio, proportion, and variation, Exponents and roots, Descriptive statistics, Operations with algebraic expressions, Equations and inequalities, Functions and graphs, Quadratic equations and functions, Sequences and series, Lines and angles, Triangles and polygons, Circles, Three-dimensional geometry, Geometric transformations, Probability, Counting methods and combinatorics, Data interpretation']

Explanation:
1. [Step 1 of the explanation]

2. [Step 2 of the explanation]
...". We also had to set the temperature value to 0.1, really low, to avoid GPT model from giving creative or unexpected responses. The answers provided by GPT engine were sometimes wrong, and this created another set of challenge.

5 Discussion

Adaptive e-learning systems using Learning Styles have been supported by numerous studies, highlighting their ability to effectively cater to individual learners' preferences, needs, and learning approaches. The increasing use of new technologies and smart devices has further propelled the popularity of adaptive e-learning platforms and applications.

The Felder Silverman model is prevalent in existing implementations, primarily due to its wide acceptance and the availability of an effective, easy-to-use questionnaire for adaptive environments. This questionnaire is crucial in accurately determining a user's learning style and shaping the system's adaptation accordingly.

Future research should explore other models or combinations of models to cover a broader range of learning systems. Integrating user scrutability and meta-cognition aspects could provide users with more flexibility to fine-tune the system rather than relying on default or modeled settings.

In our study, we modeled user motivation and interaction metrics to predict user expectations. The learning style and motivation determined the level of difficulty, which in turn influenced the questions generated by the GPT model. A feedback mechanism, based on user performance on each test and question, updated the skill schema and allowed the system to adjust the level of difficulty in a subject. Additionally, motivation scores, calculated on various metrics like user's time spent on each question and accuracy, enhanced the user modeling, making the system more accurate in determining the next set and type of questions.

Future work in this area could focus on providing users the ability to change predefined settings for question generation, enabling user scrutability and introducing a meta-cognition aspect to the system. Furthermore, prompts could be made more dynamic and fluid based on additional variables derived from user interactions.

For creating an adequately adapted Learner model, the combination of Learning Styles and prior knowledge yields better results in adaptation. Additionally, user behavior within the system plays a crucial role in adapting the model, helping the system dynamically adjust its outputs to learners. Modern adaptive systems should integrate both factors while incorporating learning and pedagogical theories to enhance efficiency in lower education levels.

We also established an ever-growing domain of question base by prompting ChatGPT and tagging questions with relevant ontological tags. Future work could involve utilizing the question base by assessing how different users answer the questions, and re-classifying or re-adapting questions based on multiple student performances. This would make the question base more robust and usable overall.

Moreover, the prompt engineering was kept mostly static, and in future work, it could be made more dynamic and fluid based on more variables received from the user's interaction. This would provide user scrutability and an ability to introduce some meta-cognition aspect to the system.

Furthermore, we found that the combination of learning styles and prior knowledge provides better results in the adaptation of the learner model. Additionally, user behavior within the system plays a crucial role in adapting the model and can help the system dynamically adjust its outputs to learners. Therefore, modern adaptive systems should combine these two factors and be combined with learning theories and pedagogical theories to be used more efficiently at the lowest levels of education.

We also defined an effective and ever-growing domain of question base by prompting ChatGPT and tagging the questions with the right ontological tags. In future works, we could utilize the question base based on assessing how different users provide answers to them. Questions could be reclassified and readapted based on multiple students' performance, making the question base more usable overall.

We demonstrated the effectiveness of our proposed approach that combines learning styles, prior knowledge, and user behavior within the system to provide personalized learning experiences. Our findings suggest that future research should focus on providing users with more control over the predefined settings to generate questions, providing user scrutability and introducing meta-cognition aspects to the system. Additionally, combining adaptive systems with learning theories and pedagogical theories could result in more efficient learning experiences at all levels of education.

6 Conclusion

The research paper has thoroughly explored the importance of adaptive e-learning systems and the potential advantages they offer when learning styles are incorporated into their design. By examining various learning style models, such as Felder-Silverman and Myers-Briggs Type Indicator, the paper demonstrates how understanding individual learning preferences can contribute to the development of personalized and effective e-learning systems. The implementation of systems like AHS, AEHS, AES, and ITS showcases the wide range of possibilities in this field.

The paper also delves into student modeling techniques, including overlay modeling, stereotype modeling, perturbation, machine learning techniques, and ontology-based student modeling. These methods provide a foundation for creating adaptive systems that can effectively adjust to individual learners' needs and preferences. The implemented eLearning system serves as an example of how these concepts can be applied in practice, integrating FSLSM learning styles, user motivation, interaction metrics, and GPT-based question generation.

Throughout the research, the paper highlights opportunities for future work. This includes exploring additional learning style models, incorporating user scrutability and meta-cognition, and refining question generation through student performance assessment. The paper also emphasizes the value of combining learning styles with prior knowledge and user behavior within the system to achieve effective adaptation.

Moreover, the paper showcases the development of an ever-growing domain of question base by leveraging chatgpt and using ontological tags. The potential for utilizing this question base in the future, based on assessing student responses, demonstrates another avenue for improving adaptive e-learning systems.

This research paper presents a comprehensive exploration of adaptive e-learning systems and the integration of learning styles. By examining various models, techniques, and implementations, the paper provides valuable insights and contributes to the ongoing development of effective and personalized e-learning experiences. The potential for future work in this field underscores the significance of adaptive e-learning systems and the promise they hold for improving educational outcomes for diverse learners.

References

1. Hurtado, C., Licea, G., & Garcia-Valdez, M. (2018). Integrating Learning Styles in an Adaptive Hypermedia System with Adaptive Resources. Studies in Systems, Decision and Control, 143, 49-67. https://doi.org/10.1007/978-3-319-74060-7 3
2. Thaker, K., Huang, Y., Brusilovsky, P., & Daqing, H. (2018). Dynamic Knowledge Modeling with Heterogeneous Activities for Adaptive Textbooks. 11th International Conference on Educational Data Mining, September 2019, 592-595. http://d-scholarship.pitt.edu/34939

3. Popescu, E., Badic ˇ a, C., & Moraret, L. (2009). WELSA: An intelligent and adaptive web-based educational system. Studies in Computational Intelligence, 237, 175-185. https://doi.org/10.1007/978-3-642-03214-1 17
4. Halawa, M. S., Hamed, E. M. R., & Shehab, M. E. (2016). Personalized E-learning recommendation model based on psychological type and learning style models. 2015 IEEE 7th International Conference on Intelligent Computing and Information Systems, ICICIS 2015, 578-584. https://doi.org/10.1109/IntelCIS.2015.7397281
5. Yang, G., Kinshuk, K., & Graf, S. (2010). A practical student model for a location-aware and context-sensitive personalized adaptive learning system. In Proceedings of the IEEE technology for education conference (pp. 130–133). Bombay, India.
6. Brusilovsky, P., & Milan, E. (2007). User models for adaptive hypermedia and adaptive educational systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), The adaptive web. Methods and strategies of web personalization. LNCS 4321, (pp. 3-53). Berlin Heidelberg: Springer-Verlag.
7. Grimley, M., & Riding, R. (2009). Individual Differences and Web-Based Learning. In Mourlas, C., Tsianos, N., & Germanakos, P. (Eds.), Cognitive and Emotional Processes in Web-Based Education: Integrating Human Factors and Personalization. (pp. 209-228). Hershey, PA: IGI Global.
8. Stansfield, J. C., Carr, B., & Goldstein, I. P. (1976). Wumpus advisor I: A first implementation of a program that tutors logical and probabilistic reasoning skills. At Lab Memo 381, Massachusetts Institute of Technology, Cambridge, Massachusetts.
9. Rich, Elaine. "User modeling via stereotypes." Cognitive science 3.4 (1979): 329-354.
10. Arroyo, I., Meheranian, H., & Woolf, B. P. (2010). Effort-based tutoring: An empirical approach to intelligent tutoring. In Proceedings of the 3rd international conference on educational data mining (pp. 1–10). Pittsburgh, PA, USA.
11. Tsiriga, V., & Virvou, M. (2002). Initializing the student model using stereotypes and machine learning. In Proceedings of the 2002 IEEE international conference on system, man and cybernetics (pp. 404–409).
12. Cheung, R., Wan, C., & Cheng, C. (2010). An ontology-based framework for personalized adaptive learning. In Proceedings of the 9th international conference on web-based learning (ICWL 2010) (pp. 52–61). Shanghai, China.
13. J. Clemente, J. Ramírez and A. de Antonio Expert Systems with Applications, 38 (7) (2011), pp. 8066-8078

SURVEYS:

14. Alameen, Abdalla, and Bhawna Dhupia. "Implementing adaptive e-learning conceptual Model: a survey and comparison with open source LMS." International Journal of Emerging Technologies in Learning (iJET) 14.21 (2019): 28-45.
15. Martin, Florence, et al. "Systematic review of adaptive learning research designs, context, strategies, and technologies from 2009 to 2018." Educational Technology Research and Development 68 (2020): 1903-1929.
16. Bower, Matt. "A framework for adaptive learning design in a web-conferencing environment." (2016): 223.
17. Nakic, Jelena, Andrina Granic, and Vlado Glavinic. "Anatomy of student models in adaptive learning systems: A systematic literature review of individual differences from 2001 to 2013." Journal of Educational Computing Research 51.4 (2015): 459-489.
18. Ennouamani, Soukaina, and Zouhir Mahani. "An overview of adaptive e-learning systems." 2017 eighth international conference on intelligent computing and information systems (ICICIS). IEEE, 2017.
19. Popescu, Elvira. "A Unified Learning Style Model for Technology-Enhanced Learning: What, Why and How?." IJDET vol.8, no.3 2010: pp.65-81. http://doi.org/10.4018/jdet.2010070105
20. Truong, Huong May. "Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities." Computers in human behavior 55 (2016): 1185-1193.
21. Kardan, Ahmad A., Mohammadreza Fani Sani, and Somayeh Modaberi. "Implicit learner assessment based on semantic relevance of tags." Computers in Human Behavior 55 (2016): 743-749.
22. Katsaris, Iraklis, and Nikolas Vidakis. "Adaptive e-learning systems through learning styles: A review of the literature." Advances in Mobile Learning Educational Research 1.2 (2021): 124-145.
23. Brusilovsky, P. L., et al. "Dynamic Knowledge Modeling with Heterogeneous Activities for Adaptive Textbooks." http://educationaldatamining. org/files/conferences/EDM2018/papers/EDM2018_paper_199. pdf. EDM Society, 2018.
24. Chrysafiadi, Konstantina, and Maria Virvou. "Student modeling approaches: A literature review for the last decade." Expert Systems with Applications 40.11 (2013): 4715-4729.