Applying real-time dynamic scaffolding techniques during tutoring sessions using intelligent tutoring systems

Anwar, Aamir, Haq, Ijaz Ul, Mian, Imdad Ahmad, Shah, Fadia, Alroobaea, Roobaea, Hussain, Saddam, Ullah, Syed Sajid and Umar, Fazlullah (2022) Applying real-time dynamic scaffolding techniques during tutoring sessions using intelligent tutoring systems. Mobile Information Systems, 2022. pp. 1-9. ISSN 1574-017X

http://dx.doi.org/10.1155/2022/6006467

Hindawi
Mobile Information Systems
Volume 2022, Article ID 6006467, 9 pages
https://doi.org/10.1155/2022/6006467

Research Article

Applying Real-Time Dynamic Scaffolding Techniques during Tutoring Sessions Using Intelligent Tutoring Systems

Aamir Anwar ,[1] Ijaz Ul Haq,[2] Imdad Ahmad Mian,[3] Fadia Shah,[4] Roobaea Alroobaea,[5] Saddam Hussain,[6] Syed Sajid Ullah,[7,8] and Fazlullah Umar [9]

[1]*School of Computing and Engineering, The University of West London, London W55RF, UK*
[2]*Faculty of Education and Psychology, University de Lleida, Lleida 25002, Spain*
[3]*Department of Computer Science, Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Islamabad,44000, Pakistan*
[4]*Department of Computer Science, SZABIST University, Islamabad Campus, Islamabad, Pakistan*
[5]*College of Computers and Information Technology, Taif University, Taif 21974, Saudi Arabia*
[6]*School of Computer Science and Engineering, Central South University, Changsha 410083, China*
[7]*Department of Information and Communication Technology, University of Agder (UiA), N-4898, Grimstad, Norway*
[8]*Department of Electrical and Computer Engineering, Villanova University, Villanova, PA, USA*
[9]*Department of Information Technology, Khana-e-Noor University, Pol-e-Mahmood Khan, Shashdarak 1001, Kabul, Afghanistan*

Correspondence should be addressed to Fazlullah Umar; fazlullahumer@gmail.com

Received 12 April 2022; Accepted 23 May 2022; Published 22 June 2022

Academic Editor: Zhongguo Yang

An intelligent tutoring system (ITS) is a computer system or software application that is built to replicate human tutors by supporting the theory of "learning by doing." Even though ITSs have been proven to be successful in academic studies, they still have not found large adoption by the industry due to the complexities of building such systems due to the high technical expertise and domain knowledge requirements. Attempts have been made to build authoring tools that can provide assistance in building tutoring systems; however, most of these tools are targeted toward authors that have considerable programming experience. This research proposes an authoring tool for ITS, which is targeted at novice authors with minimum technical/programming experience and provides real-time scaffolding to learner's incomplete/incorrect answers using the best scaffolding techniques. Two evaluation techniques were applied for the evaluation of the performance of the proposed authoring tool, e.g., paired *t*-test analysis and postexperiment survey. The learning gains obtained from paired *t*-test contend a significant learning gain and improvement in the learning process with enhanced learning performance with multiple scaffolding techniques as compared to single scaffolding technique experience. The postexperiment survey has a notable result that shows the effectiveness of the tutor model that ensures a very user-friendly interface, deploying scaffolding techniques and adequate control of selecting and deploying scaffolding techniques and making the authoring process easy.

1. Introduction

An intelligent tutoring system (ITS) is a software application that provides direct customized and adaptive instructions or feedback to learners, i.e., performing a task without human intervention. ITS implements the theory of "learning by doing." The ITS is used in a broad range of domains from traditional education to distance learning and training. It responds to open-ended student responses and queries. ITS's main architecture comprises of three modules, i.e., student module, tutoring module, and domain module. Figure 1 shows the architecture of a typical ITS. The domain model contains domain knowledge and information relating to the specific topic/domain. The ITS uses the domain information to provide adaptive content, select examination material, and answer questions postured by learners. Diverse information representation of a similar domain of learning is consolidated to support instructing procedures [1]. The

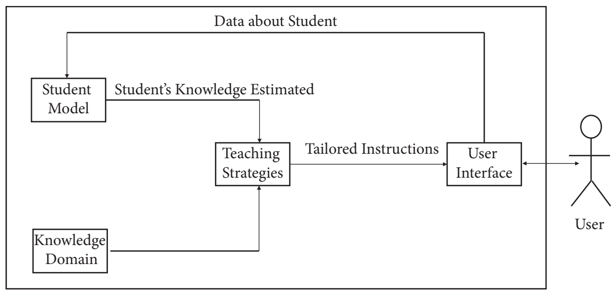

Figure 1: ITS architecture [3].

student model represents the learner's learning data and aptitudes. Information consisting of gaining knowledge of choices, beyond getting to know reviews and development, will also be applicable in adapting the teaching technique [1]. The tutor model gives the information expected to accomplish instructing objectives. The tutor model contains tutoring strategies used during tutoring a subject/course and provides knowledge and strategies that help in achieving teaching goals [2].

Intelligent tutoring systems have been delineated to enhance the performance of a student in a broad range of domains. Beal et al. performed a control evaluation for high school mathematics using an interactive online tutoring system [1]. They concluded that students who experienced online tutoring showed effective progress in learning, while students who took routine classroom instructions showed no pretest to posttest enhancement. AutoTutor, an ITS, achieved learning gains of approximately 0.8 sigma by simulating a human tutor having a conversation with the students in natural language [2]. In another study, considering a large-scale experiment, 470 students took classes on an experimental basis using an algebra tutor and showed improvement in contrast to classes by 15% on standardized tests [4].

Authoring an effective ITS is a nontrivial task that requires a certain level of understanding of teaching strategies and technical expertise. Modifying any feature leads toward the involvement of highly skilled programmers to modify/change any feature. Modification becomes necessary when a teacher/domain expert tries to modify course contents and teaching strategies or incorporate any new materials. Authoring ITS requires high technical and programming expertise. It has been noticed that it requires 100 hours of development time to make 1 hour of instructions [5]. Building ITS requires ability in many fields, including education, software engineering, artificial intelligence, psychological research, and interface designing. This necessity makes building ITSs, starting with no outside help, a tedious and challenging procedure and expands the expenses of making the ITS, constraining the quantity of ITSs. The principle purpose of an ITS authoring tool is to make the process of constructing an ITS less difficult. This ease interprets into decreased price and ability threshold for novice authors. The ITS authoring tools come across many challenges. In the past decade of research, the community enormously worked on developing effective and adaptive authoring tools for intelligent tutoring systems.

Authoring systems are divided into two main categories: authoring tools for novice and nontechnical users and shells for technical and domain experts [6]. Shell is a generalized framework/building block for building ITS [7]. Shell requires programming experts to develop ITS. Developing a shell is a costly job, which requires technical expertise and modification to shells that become costly in time and budget. Many studies have been published on different shells and their usage in different environments [8–11].

An authoring tool is an ITS shell having a user interface that facilitates modification without programming expertise. Authoring tools do not require programming expertise to modify and develop an ITS. Due to the cost-efficient and natural-driven approach, IT experts prefer authoring tools over shells to develop ITS for various domains [7, 12].

2. Related Work

The improvement of intelligent tutoring has been given special attention. Over time, authoring systems have been improved, supported by multiple features and providing advanced functionalities. This section discusses such systems from very basic authoring systems, based on limited features to a more advanced level.

The idea of providing more adaptive scaffolding depends on students' online behavior with the intelligent tutoring system. The study uses an AC-ware Tutor that examines students' online behavior. Based on the online behavior of the students, the tutor divided the students into different groups using clustering techniques. The results of the students' clusters are constructed using a decision tree to get a human-readable description [13]. This study emphasizes on analyzing students' behavior in order to provide more adaptive scaffolding by the intelligent tutoring system.

A learner model of an intelligent tutoring system constructed from learner behavior provides adaptive tutoring. It is effective for ITS to apply multiple scaffolding techniques in ITS based on students' learning proficiency. The study resulted in a developed learner model that can reduce the hints and feedback [14].

In 1996, such a tool was created that enabled its users to create adaptive books on www [15]. This tool was based on concepts not only drawn from the intelligent tutoring system but also on adaptive hypermedia as well. The basic idea that was used behind making the educational material flexible and intelligent was concept-based indexing. Concept-based indexing provided the relevant details of the unit by indexing it along with relevant concepts. [15].

In 1997, RIDES provided such a software environment that enabled users to create tutorials that were based on models and graphic simulations [16]. Such tutorials were extremely useful for students in grasping the relevant concepts. The authors simply had to create the interactive model of the system from which interactive lessons were created. [16]. In the same year, CALAT was also proposed [17]. The CALAT was such an authoring system that was courseware. It possessed such a feature that enabled it to convert according to the capacity of its learner. It selected and presented its learner with such relevant material that was according to their level. The students interacted with the updated learning strategies, which were created using the relevant content of the course [17].

Merrill et al. investigated step-based tutoring and providing vocabulary in LISP programming courses [18]. Anderson et al. so studied step-based tutoring and example-based tutoring in a geometry course [19]. Step-based tutoring is used in arithmetic, electrical troubleshooting [20], practical algebra tutor [4], fundamental skill training [21, 22], personal assistance for learning, and websoc individualized tutoring [23].

Later, more advances were seen in the field of intelligent tutoring making it more specified, realistic, efficient, and effective. These systems benefited a wide range of users apart from only instructors and students. In 2001, REDEEM, such a tool that enabled users to create ITS from computer-based training, was proposed [24]. The REDEEM provided different authoring stages with the help of which required shells of ITS were organized. The existing material was supplemented with more questions. The whole course was then specialized in a way that the needs of specific learners were fulfilled. [24].

In 2004, such a system had been proposed that taught students in the form of conversation [25]. AutoTutor, which was able to teach by communication in natural language, had been designed after inspiring learning theories, empirical studies, and intelligent tutoring systems. With the help of this tool, the problem to be answered was presented and then the user was engaged in a way that ultimately directed the user in creating an answer. These answers were then further improved based on the feedback given by AutoTutor [25]. In 2006, it is based on a collection of authoring tools with the help of which lessons can be efficiently and easily created. Cognitive tutors require AI programming for their creation and were created without programming. The tutors created with the help of CTAT not only helped in instruction but were also useful in empirical studies. [26].

In 2014, GIFT was such an ITS framework that was composed of multiple modules that were proposed [27]. Unlike other ITS systems, which were only limited to some specific domain, GIFT was able to provide a wide range of functionalities. Since GIFT possesses multiple tools, therefore, different users can benefit from it. Those users included researchers, students, instructors, and designers. The GIFT is time- and cost-efficient as well. [27].

Anne Lippert et al. explored multiple scaffolding techniques in AutoTutor, as an adoptive tutor to scaffold multiple levels of discourse processing to teach comprehension to the adults. This study scaffolds different six major learning affordances that help and teach adults comprehension strategies [28]. A significant result was shown by Haiying et al. by providing real-time scaffolding to the students in intelligent assessment systems. The authors significantly worked on real-time feedback but failed to accommodate the most applicable and advanced tutoring strategies of ITSs in their assessment system [29].

Another milestone was achieved by ElectronixTutor that helps learners to learn STEM topics with multiple learning resources for the electronic domain. Instead of scaffolding technique, the ElectronixTutor successfully combines multiple learning resources into a single intelligent tutoring system. A prototype of ElectronixTutor is implemented, tested, and showed higher learning gains in the students [30].

Another study investigates the intelligent tutoring system on a large scale for automated personalized feedback to improve student learning. The study uses a machine learning approach, which takes individual students' performance and generates personalized feedback. This provides multiple feedback techniques with personalized hints, Wikipedia-based explanations, and mathematical hints [31].

Helping students in problem-based learning, the research explores students' difficulties in problem-based systems and scaffolding designing of learner-centered scaffolding systems for addressing their challenges. This study helped students to enhance students' experience of autonomy and competence [32]. Huang et al. introduced a multimethod approach to improve adaptivity in intelligent tutoring systems. This study scaffolds different data mining and instructional strategies in the knowledge component and redesigned the Algebraic Math Tutor. The results were encouraging and show that students showed higher learning gains on more difficult tasks [33].

In 2014, Ma et al. [34] analyzed in their studies the effects of ITS on-learning outcomes. They examined how much different ITS systems, based on the type of comparison, type of knowledge, learning outcome etc., effect the learning outcomes. They analyze that the use of ITS is greatly associated with the other forms of learning methodologies.

In 2015, Dr. Vyshnavi et al. [35] implemented an ITS on top of learning management systems. In their studies, the ITS was divided into two frameworks. The ITS framework elements consist of competencies, instructional, and assessments. A taxonomy table tool was used to align these components.

In 2017, Zrigui and Abdelaziz [36] focused on the main module of ITS, domain module. They developed an ITS architecture providing an interface to the author with the functionality to insert didactical and pedagogical knowledge. This system was aimed to identify the knowledge-based contents and their inter-relationship to minimize the hurdles while building domain knowledge.

In 2020, El Ouaazizi and Akharraz [37] in their math education ITS system (ITSME) include different ontologies to provide a dynamic and effective e-learning process for the linear algebra course. They use two ontologies, a domain ontology of the linear algebra course (OLAC) and ontology of e-learning field (OeLF). The first one designs the linear algebra contents (knowledge), and the second one models the additional features of the learning process. They present in their study the importance of using ontologies in e-learning ITS and the prominence of links between the basic knowledge-based elements.

Another study carried out in 2020 by Huang et al. designs an ITS, which deployed a multimethod approach to design loop adaptivity. They focused on identifying knowledge components that are demonstrably difficult for students to learn and optimize the effective and efficient practice of them. They also investigate the learning differences between treatment condition, through data-driven redesigned tutor, control condition, and original tutors. The empirical results showed that students spent much less time on the tutor than planned [33].

A significant investigation explores the origin of scaffolding in theoretical-historical connection. Scaffolding is explored with Vygotsky's zone of proximal development (ZPD), explicitly the evolution of the notion of scaffolding. This study describes the introduced scaffolding supported by various tools and social scaffolds in the classroom environment. This study discusses the notion of distributed scaffolding of fading and transfer of responsibility features [38].

Sanjay Singh and Vikram Singh [39] proposed a novel framework that considers questions quality and errors as key factors in the designing of practicing items for student profile in intelligent tutoring systems. In their study, they establish that the factors like question quality and errors in the formulation of practice items are not considered in any of the models considered so far. To estimate the knowledge levels of the students in various topics from the answer responses given by the students, it has been proposed to augment the traditional models and devise a new framework that surpasses the existing models in terms of knowledge prediction accuracy and better student profiling. [39].

3. NDLAuthor

The NDLAuthor is an authoring tool for ITS. The NDLAuthor provides support to novice domain experts and teachers to develop a domain-specific tutoring system using any programming knowledge. The NDLAuthor maximizes the use of authoring tools for developing a tutoring system by providing a simple and component-based mechanism. Due to a lack of programming complexity, NDLAuthor enhances the productivity of domain-specific tutoring system development for novice domain experts and teachers. The NDLAuthor components are as follows.

4. Scaffolding Techniques

Scaffolding enables the instructor to construct a link between the learners' present knowledge and the content being taught. Scaffolding is effectively performed by a teacher by modeling a specific activity and gradually imparting information to the student so that he has a strong grip on the subject matter. Scaffolding may be disadvantageous to instructors since it requires them to transfer control to allow students to study at their own speed. It is also time-consuming; you may not have enough time to finish the full scaffolding during lessons. Scaffolding techniques are selected based on their successful use in different scenarios in various ITS. These scaffolding techniques are used by many domains' experts and teachers for tutoring purposes. The scaffolding techniques we used in our system are as follows:

(i) Example-based scaffolding
(ii) Using visual aids
(iii) Activating prior knowledge
(iv) Preteach vocabulary
(v) Breaking complex into small
(vi) Offering hints and partial solutions

5. Mapping Scaffolding with Different Scenarios

In human tutoring, scaffolding techniques are used in different scenarios. Human tutoring is the most effective tutoring used for hundreds of years. In human tutoring, the teacher or tutor scaffolds students' knowledge using scaffolding techniques according to a suitable scenario. Scaffolding becomes effective only if it is used in an appropriate scenario. To keep because of scenarios' importance, we mapped different scaffolding techniques with their relevant scenario to make scaffolding more effective. Mapping is based on the use of the scaffolding technique in a domain-specific scenario. Table 1 shows different scaffolding techniques mapping with scenarios.

6. System Design of the Tutor Module

Our tutoring module will use component-based authoring (ComBAT) technology. The ComBAT is a technology based on the development of individual components that can work as standalone and integrate with other components to build a complete system. In the present work, we are designing an authoring tool for tutoring modules that will support multiple scaffolding techniques in ITS. These scaffolding techniques will facilitate students and will enhance their learning abilities during tutoring.

The tutoring module will perform several functionalities to implement scaffolding techniques during tutoring.

6.1. Selecting Domain. The first step is to select the domain from the given list of domains. As we only worked on the mathematics domain, so only the mathematics domain is available.

6.2. Selecting Subject. After selecting a domain, the author will select a subject of that specific domain already selected in the previous step. For example, for mathematics, domain subjects can be basic mathematics, algebra, etc.

6.3. Selecting Topic. The third step is to select the topic of that specific subject. There can be many topics in a subject. So, the author will select the topic of that subject. For example, for basic mathematics, a subject topic can be mathematical operations.

6.4. Selecting Question. After topic selection, a list of questions will be available along with expert, intermediate, novice, and bad answers to those questions. The author will select a question to apply scaffolding for that question.

Table 1: Scaffolding mapping with different scenarios.

Scaffolding technique	Used in scenario	Subject/domain
Example-based	1. When students are new to the course and do not know much 2. Providing help to students stuck at a stage of course 3. Learning argumentation skills 4. Learning to construct concept maps	Mathematics, physics, statistics, computer science, English, and social science
Using visual aids	1. To encourage the student with attention 2. For enhancing the cognitive skills of the student 3. Demonstrating the topic using animations and visual aids	Basic mathematics, language's learning, and medical science
Activating prior knowledge	1. To recall knowledge about a certain topic 2. Preexisting knowledge acts as a foundation to learn new knowledge 3. Connecting new knowledge with basic knowledge	Mathematics, engineering science, medical science, and social science
Preteach vocabulary	1. To introduce new instruction/format 2. To learn/use the new language	Languages and computer programming languages
Breaking complex into small	1. To solve a complex problem 2. To motivate the learner to solve and learn from small chunks of problems	Problem-solving, mathematics, and analytical
Offering hints/partial solutions	1. Motivate learner to proceed further rather than stuck in a problem for hours 2. Helps to solve a problem	Problem-solving subjects, computer-aided programs, and software solutions

6.5. Providing Acceptance Ranges. The authors will provide acceptance ranges for each type of answer that according to the author what the acceptance range will be for expert, intermediate, and novice answers.

6.6. Providing Reinforcement Technique for an Expert Answer. If the student answer is of the expert level, so we cannot apply scaffolding to the expert answer as there is no need for scaffolding for an expert answer. Here, the author will provide a reinforcement technique to encourage the student. Reinforcement can be in the form of an appreciation message.

6.7. Providing Scaffolding Technique for Intermediate Answer. If the student answer is of intermediate level, i.e., according to the acceptance criteria of the intermediate answer already provided by the author, then the author will select a scaffolding technique to scaffold student knowledge and to help the student in providing an expert answer. Scaffolding will be provided in three iterations for the intermediate answer. If a student failed to correctly answer the question after the first scaffolding technique, then the second iteration of scaffolding will be applied.

6.8. Providing Scaffolding Technique for Novice Answer. Same as the intermediate scenario, if the student's answer is of novice level, then the author will provide scaffolding techniques to help the student in the learning process. Same as the intermediate scenario, scaffolding will be deployed in three iterations.

6.9. Providing Scaffolding Technique for Bad Answer. For a bad answer, the author will also provide scaffolding techniques in three iterations to help the student in learning so the student can provide a correct answer to a question.

7. Implementation of Proposed Tutor Model Authoring Component

The system design discussed in the previous chapter is implemented into a software component of the tutor module for NDLAuthor. This prototype contains all the basic functionalities of the tutor module along with the deployment of different scaffolding techniques. The tutor module is integrated with the domain module and all the domain knowledge, i.e., questions come from the domain module. Figure 2 illustrates the question section screen of ITS.

7.1. Descriptive Question Scaffolding. For descriptive question scaffolding, users define ranges for excellent, intermediate, novice, and bad answers Figure 3 shows the scaffolding technique selection for descriptive questions.

7.2. Calculation-Based Question Scaffolding. For calculation-based question scaffolding, excellent, intermediate, novice, and bad answers are defined. Figure 4 shows scaffolding techniques for calculation-based questions.

7.3. Tester Tutor (ITS). The test page is a tutoring system used to validate scaffolding techniques. On the test page, the student will answer questions and the tutor will check whether the answer is an expert, intermediate, novice, or bad answer. Figure 5 shows the real-time interaction of student with the ITS and the deployment of scaffolding techniques.

8. System Evaluation

Two evaluation techniques were applied. The first technique was to measure students' learning gain in a single scaffolding technique and multiple scaffolding techniques.

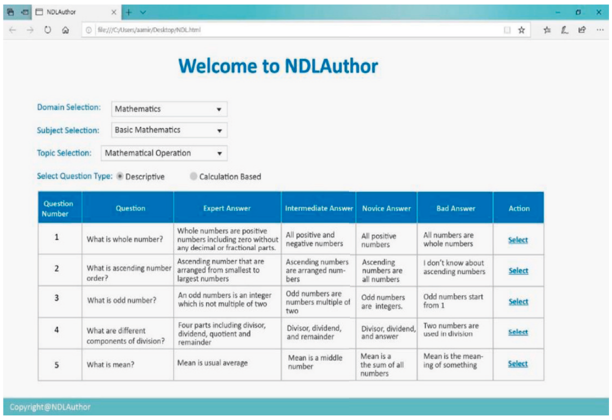

Figure 2: Question selection.

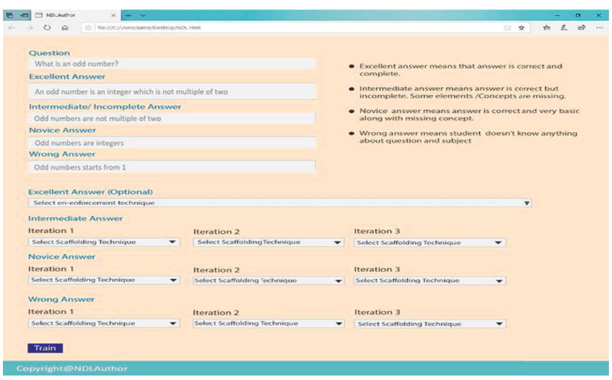

Figure 3: Scaffolding for descriptive questions.

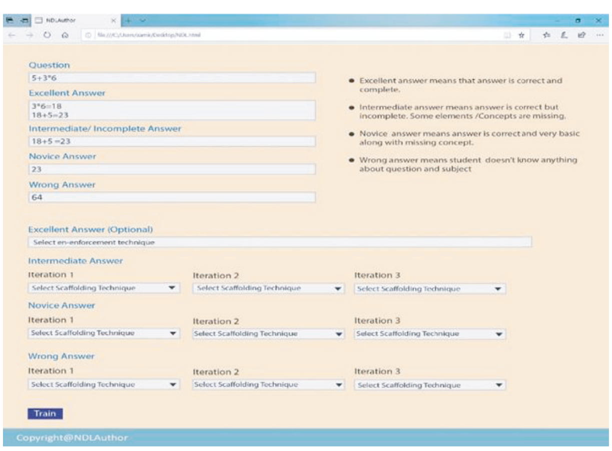

Figure 4: Scaffolding for calculation-based questions.

Second, a postexperiment survey was conducted to record user responses regarding the usability and application of the proposed tutor module authoring tool.

Pretests and posttests were taken in both conditions, i.e., using the single scaffolding technique and multiple scaffolding techniques. Paired t-test analysis was performed on

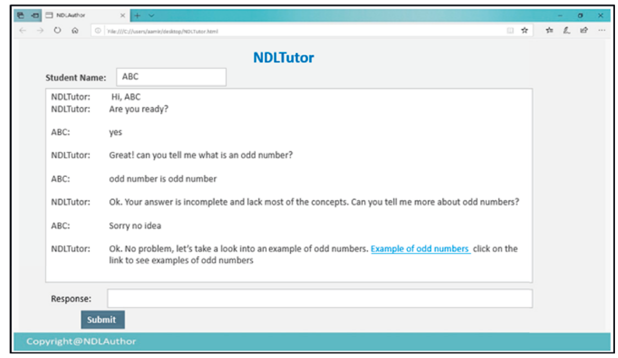

Figure 5: Tester tutor.

the measurements. Descriptive results of mean, standard deviation, and frequency analysis were measured. Learning gain was measured for these conditions using this following formula:

$$\text{learning gain} = \text{preexperiment test} - \text{postexperiment test}. \tag{1}$$

Analysis of learning gain in both conditions was evaluated and assessed for student performance measurement.

8.1. Postexperiment Survey. We experimented to test system performance and efficiency. The experiment consisted of the task of building an interaction session for ITS using multiple scaffolding techniques. A postexperiment survey was conducted to record user responses regarding the usability and application of the proposed tutor module authoring tool. For the experiment, we selected 20 novice users. These participants were students of the postgraduate program and were selected because they were also working as junior teaching faculty at various institutes. None of the participants had previous experience of authoring an ITS. The experiment consisted of a single session with the NDLAuthor tutor module authoring tool. Before the session, participants were oriented on how to use the tutor module of NDLAuthor and the different options available to them. Each participant was asked to create an interactive session using the tutor module of NDLAuthor.

After sessions with participants, a postexperiment survey was conducted to record participant's responses and experiences with the NDLAuthor tutor model. The postexperiment survey consisted of five questions regarding tutor model performance, usability, and application.

9. Results and Discussion

The result shows prominent improvement in students learning gain by using multiple scaffolding techniques.

9.1. Learning Gain. During experimentation, students firstly interacted with the ITS and the single scaffolding technique was applied to calculate students' learning gain and then in

Table 2: Learning gain of single scaffolding technique.

No.	Single scaffolding tutoring				Learning gain	
	Pretest		Posttest			
	Mean	SD	Mean	SD	Mean	SD
48	3.48	2.052	4.75	1.657	1.7708	1.05668

Table 3: Single scaffolding technique frequency measurement.

Single scaffolding technique Frequency measurement	
0.00	5
1.00	15
2.00	17
3.00	8
4.00	3
Total	48

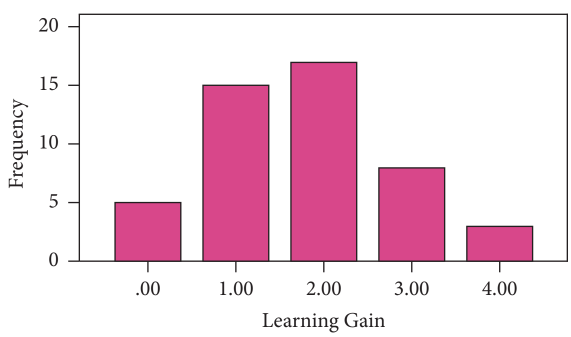

Figure 6: Single scaffolding technique learning gain result.

another session multiple scaffolding techniques were applied to again calculate students' learning gain.

9.1.1. Using Single Scaffolding Techniques. To measure students' learning gain, we performed statistical analysis by applying paired t-tests on pretest and posttest. Keeping 47 degrees of freedom, paired *t*-tests showed a significant difference ($p < 0.05$). This statistical result is presented in Table 2.

We performed a *t*-test on the learning gain of the single scaffolding technique to compare it with the learning gain of multiple scaffolding techniques. *T*-test of learning gain has a mean of 1.77, standard deviation of 1.056, and high significance ($p < 0.05$).

Analysis measurement of learning gain also showed significant results. Fifteen students improved by 1 point, 17 students improved by 2 points, and so on. Overall learning gain frequency analysis is shown in Table 3.

Also, for a more clear representation analysis of learning gain, results are shown through a bar graph in Figure 6.

9.1.2. Using Multiple Scaffolding Techniques. Using the second condition to measure students' learning gain, we performed statistical analysis by applying paired t-tests on pretest and posttest. Keeping 47 degrees of freedom, paired

Table 4: Learning gain of multiple scaffolding techniques.

No.	Multiple scaffolding tutoring				Learning gain	
	Pretest		Posttest			
	Mean	SD	Mean	SD	Mean	SD
48	3.79	1.700	6.083	1.568	2.583	1.441

Table 5: Multiple scaffolding techniques' frequency measurement.

Multiple scaffolding techniques Frequency measurement	
0.00	5
1.00	7
2.00	9
3.00	12
4.00	12
5.00	3
Total	48

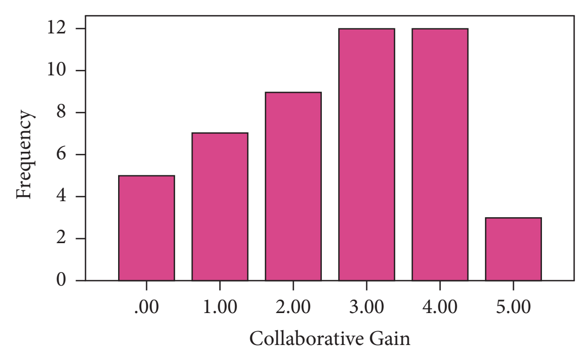

Figure 7: Multiple scaffolding techniques' learning gain results.

t-tests showed a significant difference ($p < 0.05$). Also, compared with the single scaffolding technique, multiple scaffolding techniques showed improvement in learning gain. These statistical results are presented in Table 4.

Learning gain of multiple scaffolding techniques has a high mean value of 2.583 that is greater than single scaffolding technique experience. Also, a less standard deviation of multiple scaffolding was recorded as compared to the single scaffolding technique. Frequency analyses of learning gain of multiple scaffolding techniques have also significantly higher performance than single scaffolding techniques. Frequency measurement is given in Table 5.

Also, for a more clear comparison of single and multiple scaffolding techniques, frequency analysis of learning gain of multiple scaffolding techniques is given in Figure 7.

9.2. Postexperiment Survey. This evaluation was based on five basic questions to know participants' perceptions and experiences of using the NDLAuthor tutor model. These questions focused on the novice user experience of using the tutor model to create an ITS system. The participant's response in Table 6 and system evaluation indicated the following basic outcomes.

Table 6: Participants' perceptions and experience of using the NDLAuthor tutor model.

Survey questions	<strongly agree..strongly disagree> (5)	(4)	(3)	(2)	(1)	Mean
NDLAuthor tutor model interface is very user friendly	12	5	0	2	1	4.25
Tutor model provides help to facilitate user in deploying scaffolding techniques	11	7	0	2	0	4.35
Tutor model provides adequate control of selecting and deploying scaffolding techniques according to tutoring strategy	13	4	0	2	1	4.30
Tutor model makes the authoring process easy	15	3	0	2	0	4.55
The testing phase helps visualize how the tutoring model will work in a real-time system	11	3	5	1	0	4.20

10. Conclusions

Different scaffolding techniques were identified from different domains of studies. These scaffolding techniques are actively used in human tutoring to scaffold students' knowledge and provide help in learning. After the identification of scaffolding techniques, these techniques were mapped with scenarios that can occur during the learning process. Mapping scenarios helped us to identify the deployment of useful scaffolding techniques in a scenario. The scaffolding technique works efficiently if it is deployed in a relevant scenario. After the mapping of scaffolding techniques with different scenarios, we mapped different scaffolding techniques with mathematics domain characteristics. As we implemented our NDLAuthor tutor model only for the mathematics domain, we mapped different scaffolding techniques with mathematics characteristics. We divided mathematics questions into two categories, i.e., descriptive questions and calculation-based questions. Descriptive questions are simple narrative-type questions without any mathematical steps while calculation-based questions contain mathematical calculations. After scaffolding mapping, we deployed the NDLAuthor tutor model to validate our mapping of scaffolding. The evaluation of the study has a notable result that shows the effectiveness of the tutor model that ensures a very user-friendly interface, deploying scaffolding techniques, and adequate control of selecting and deploying scaffolding techniques, and making the authoring process easy.

Data Availability

The data used in this research can be obtained from the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors are grateful to the Taif University Researchers' Supporting Project (TURSP-2020/36), Taif University, Taif, Saudi Arabia.

References

[1] I. U. Haq, A. Anwar, I. U. Rehman et al., "Dynamic group formation with intelligent tutor collaborative learning: a novel approach for next generation collaboration," *IEEE Access*, vol. 9, pp. 143406–143422, 2021.

[2] A. C. Graesser, P. Chipman, B. Haynes, and A. Olney, "AutoTutor: an intelligent tutoring system with mixed-initiative dialogue," *IEEE Transactions on Education*, vol. 48, no. 4, pp. 612–618, 2005.

[3] I. U. Haq, A. Anwar, I. Basharat, and K. Sultan, "Intelligent tutoring supported collaborative learning (itscl): a hybrid framework," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 8, 2020.

[4] R. K. Kenneth, R. A. John, H. H. William, and A. M. Mary, "Intelligent tutoring goes to school in the big city," *International Journal of Artificial Intelligence in Education*, vol. 8, no. 1, pp. 30–43, 1997.

[5] S. B. Gilbert, S. B. Blessing, and E. Guo, "Authoring effective embedded tutors: an overview of the extensible problem specific tutor (xPST) system," *International Journal of Artificial Intelligence in Education*, vol. 25, no. 3, pp. 428–454, 2015.

[6] T. Murray, "An overview of intelligent tutoring system Authoring tools: updated analysis of the state of the art," in *The Authoring Tools for Advanced Technology Learning Environments*, pp. 491–544, Springer, Dordrecht, 2003.

[7] T. Murray, "Authoring intelligent tutoring systems: an analysis of the state of the art," *International Journal of Artificial Intelligence in Education*, vol. 10, pp. 98–129, 1999.

[8] Z. Yang, M. Zhang, Z. Zhang, H. Li, C. Liu, and S. Ali, "Lecture information service based on multiple features fusion," *International Journal of Software Engineering and Knowledge Engineering*, vol. 31, no. 04, pp. 545–562, 2021 Apr.

[9] S. Ali, S. Baseer, I. A. Abbasi, B. Alouffi, W. Alosaimi, and J. Huang, "Analyzing the interactions among factors affecting cloud adoption for software testing: a two-stage ISM-ANN approach," *Soft Computing*, vol. 25, pp. 1–29, 2022 Apr.

[10] F. Shah, A. Anwar, I. ul haq, S. Hussain, S. Al-Hadhrami, and S. Al-Hadhrami, "Artificial intelligence as a service for immoral content detection and eradication," *Scientific Programming*, vol. 2022, pp. 1–9, 2022 Jan.

[11] F. Shah, Y. Liu, A. Anwar et al., "Machine learning: the backbone of intelligent trade credit-based systems," *Security and Communication Networks*, vol. 2022, pp. 1–10, 2022 Mar.

[12] M. Z. Khan, R. Naseem, A. Anwar et al., "A novel approach to automate complex software modularization using a fact

extraction system," *Journal of Mathematics*, vol. 2022, pp. 1–19, Article ID 8640596, 2022.

[13] Š. G. Ines, G. Ani, Š. Lijilana, and J. R. Timothy, "Student clustering Based on learning behavior data in the intelligent tutoring system," *International Journal of Distance Education Technologies*, vol. 18, no. 2, pp. 73–89, 2020.

[14] K. Kento, F. Yuki, T. Takahito, H. Tomoya, and H. Tsukasa, "Learner model for adaptive scaffolding in intelligent tutoring systems for organizing programming knowledge," in *Proceedings of the International Conference on Human-Computer Interaction*, pp. 63–74, Springer, Cham, June 2021.

[15] P. Brusilovsky, E. W. Schwarz, and G. Weber, *A Tool for Developing Adaptive Electronic Textbooks on*, WWW, Webnet, 1996.

[16] A. Munro, C. J. Mark, A. P. Quentin, S. S. David, and M. T. Douglas, "Authoring simulation-centered tutors with RIDES," *International Journal of Artificial Intelligence in Education*, vol. 8, no. 3-4, pp. 284–316, 1997.

[17] M. Kiyama, I. Satomi, I. Kayo, T. Masahiko, and F. Yoshimi, "Authoring methods for the Web-based intelligent CAI system CALAT and its application to telecommunications service," in *Proceedings of the AAAI-97: Conference of the American Association for Artificial Intelligence*, Providence, RI, USA, July 1997.

[18] D. C. Merrill, B. J. Reiser, M. Ranney, and J. G. Trafton, "Effective tutoring techniques: a comparison of human tutors and intelligent tutoring systems," *The Journal of the Learning Sciences*, vol. 2, no. 3, pp. 277–305, 1992.

[19] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier, "Cognitive tutors: lessons learned," *The Journal of the Learning Sciences*, vol. 4, no. 2, pp. 167–207, 1995.

[20] S. D. Johnson, J. W. Flesher, J. J. Jehng, and A. Ferej, "Enhancing electrical troubleshooting skills in a computer-coached practice environment," *Interactive Learning Environments*, vol. 3, no. 3, pp. 199–214, 1993.

[21] K. Steuck and M. M. Todd, *Evaluation of and Authentic Learning Environment for Teaching Scientific Inquiry Skills*, The American Educational Research Association, San Diego, 1997.

[22] J. L. Wheeler and J. W. Regian, "The use of a cognitive tutoring system in the improvement of the abstract reasoning component of word problem solving," *Computers in Human Behavior*, vol. 15, no. 2, pp. 243–254, 1999.

[23] K. E. Chang, Y. W. Kang, Y. D. Chienand, and S. C. The, "Learning recursion through a collaborative Socratic dialectic process," *Journal of Computers in Mathematics and Science Teaching*, vol. 18, no. 3, pp. 303–315, 1999.

[24] S. Ainsworth, W. Ben, and W. David, "Using the REDEEM ITS authoring environment in naval training," in *Proceedings of the IEEE International Conference on Advanced Learning Technologies*, pp. 189–192, Madison, WI, USA, 06-08 August 2001.

[25] A. C. Graesser, S. Lu, G. T. Jackson et al., "AutoTutor: a tutor with dialogue in natural language," *Behavior Research Methods, Instruments, & Computers*, vol. 36, no. 2, pp. 180–192, 2004.

[26] V. Aleven, S. Jonathan, M. M. Bruce, and R. K. Kenneth, "Rapid authoring of intelligent tutors for real-world and experimental use," in *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies (ICALT'06)*, pp. 847–851, Kerkrade, Netherlands, 05-07 July 2006.

[27] A. M. Sinatra, B. S. Goldberg, and R. A. Sottilare, "The Generalized Intelligent Framework for Tutoring (GIFT) as a tool for human factors professionals," *Proceedings of the Human Factors and Ergonomics Society - Annual Meeting*, vol. 58, no. 1, pp. 1024–1027, 2014.

[28] A. Lippert, G. Jessica, C. Zhiqiang, and C. G. Arthur, "Using an adaptive intelligent tutoring system to promote learning affordances for adults with low literacy skills," in *Proceedings of the International Conference on Human-Computer Interaction*, pp. 327–339, Orlando, FL, USA, July 2019.

[29] L. Haiying, G. Janice, D. Rachel, and M. Raha, "The impact of multiple real-time scaffolding experiences on science inquiry practices," in *Proceedings of the International Conference on Intelligent Tutoring Systems*, pp. 99–109, Bucharest, Romania, June 2022.

[30] A. C. Graesser, X. Hu, B. D. Nye et al., "ElectronixTutor: an intelligent tutoring system with multiple learning resources for electronics," *International journal of STEM education*, vol. 5, no. 1, p. 15, 2018.

[31] K. Ekaterina, D. V. Dung, B. Robert, G. Varun, V. S. Lulian, and P. Joelle, "Automated personalized feedback improves learning gains in an intelligent tutoring system," in *Proceedings of the International Conference on Artificial Intelligence in Education*, pp. 140–146, Springer, The Netherlands, June 2021.

[32] N. J. Kim, B. R. Belland, and D. Axelrod, "Scaffolding for optimal challenge in K–12 problem-based learning," *Interdisciplinary Journal of Problem-Based Learning*, vol. 13, no. 1, 2018.

[33] Y. Huang, V. Aleven, E. McLaughlin, and K. Koedinger, "A general multi-method approach to design-loop adaptivity in intelligent tutoring systems," in *Proceedings of the International Conference on Artificial Intelligence in Education*, pp. 124–129, Springer, The Netherlands, June 2021.

[34] W. Ma, O. O. Adesope, J. C. Nesbit, and Q. Liu, "Intelligent tutoring systems and learning outcomes: a meta-analysis," *Journal of Educational Psychology*, vol. 106, no. 4, pp. 901–918, 2014.

[35] M. R. Vyshnavi, N. J. Rao, and R. Chandrashekar, "Implementation of an intelligent tutoring system using Moodle," in *Proceedings of the IEEE Frontiers in Education Conference, (FIE)*, pp. 1–9, IEEE, El Paso, TX, USA, October 2015.

[36] S. Zrigui and A. M. Abdelaziz, "Automatic insertion of subject content in domain knowledge of an intelligent tutoring system," in *Proceedings of the Intelligent Systems and Computer Vision (ISCV),*, pp. 1–5, IEEE, Fez, Morocco, April 2017.

[37] M. El Ouaazizi and I. Akharraz, "Ontology-based model for a linear algebra intelligent tutoring system," in *Proceedings of the 2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, pp. 1–3, IEEE, Taza, Morocco, December 2019.

[38] S. Puntambekar, "Distributed scaffolding: scaffolding students in classroom environments," *Educational Psychology Review*, vol. 34, pp. 451–472, 2021.

[39] S. Singh and S. Vikram, "A novel approach to student profiling in intelligent tutoring systems," *Wesleyan Journal of Research*, vol. 13, no. 27, 2020.