Expert-Authored and Machine-Generated Short-Answer Questions for Assessing Students' Learning Performance

Owen H. T. Lu[1], Anna Y. Q. Huang[2], Danny C. L. Tsai[2] and Stephen J. H. Yang[2*]
[1]College of Computer Science, National Pingtung University, Taiwan // [2]Department of Computer Science and Information Engineering, National Central University, Taiwan // owen.lu.academic@gmail.com // anna.yuqing@gmail.com // dan860202@gmail.com // stephen.yang.ac@gmail.com
[*]Corresponding author

ABSTRACT: Human-guided machine learning can improve computing intelligence, and it can accurately assist humans in various tasks. In education research, artificial intelligence (AI) is applicable in many situations, such as predicting students' learning paths and strategies. In this study, we explore the benefits of repetitive practice of short-answer questions could enhance students' long-term memory for subsequent improvements in learning performance. However, frequent authoring questions and grading requires teachers' professionalism, experience, and considerable efforts. Therefore, this study using modern AI technologies, specifically natural language processing, to provide Automatic question generation (AQG) solution, a combined semantics-based and syntax-based question generation system: Hybrid automatic question generation (Hybrid-AQG) was proposed in this study. We assessed its functionality and student learning performance by asking 91 students to complete short-answer questions and then applied a process similar to the Turing test to evaluate the question and grading quality. The results demonstrated that modern AI technologies can generate highly realistic short-answer questions because: (1) Compared with the control group, the experimental group exhibited significantly better learning performance, implying that students acquired long-term memory of course knowledge through repetitive practice with machine-generated questioning. (2) The experimental group could better distinguish machine-generated and expert-authored questions. Nevertheless, both groups in distinguishing questions presented like guessing. (3) Machine grading was deficient in some respects; but the way students answer questions can be adapted for machine understanding through repetitive practice.

Keywords: Automatic question generation, Learning performance, Artificial intelligence, Practice testing, Turing test

1. Introduction

1.1. Artificial Intelligence in Education (AIED)

Artificial intelligence (AI) refers to machines thinking and acting rationally like humans (Russell & Norvig, 2002). One of the AI implementation is the agent software, which require machine to take actions to support human for solving issues or dealing with tasks. The most simplify agent was implemented by the rules, however, it is very difficult to input every rule into machine due to the environment complexity, and physical capacity limitations. Algorithms are sometimes implemented to reduce complexity—for example, the least-cost-path algorithm (Collischonn & Pilar, 2000). Since 1970, the application of artificial intelligence in education (AIED) has been a very interesting research topic. Intelligent Tutoring System (ITS) is the most common implementation studied in AIED (du Boulay, 2016). The ITS was studied aiming at identifying at-risk students to monitor the learning behavior of students and generate personalized learning recommendations (Woolf, 2010). ITS has shown considerable improvement in students' performance and outcomes in learning (Ma, Adesope, Nesbit, & Liu, 2014; Schroeder, Adesope, & Gilbert, 2013). In recent years, due to the huge data availability and improved digital technologies, the AIED has been much easier to study and implement. Chen, Xie, and Hwang (2020) systematically aggregated the artificial intelligence-based research performances in the field of education, and the statistical data shows that 74% of the researches was conducted in the past seven years from 1999 to 2019, which indicates the importance of this research topic in recent years. Although the above research summary shows the importance of AIED, Yang (2021) explain that bringing AI into education is not to just apply digital technology into the classroom; educators also need to be human-centric (Yang, 2021). Human-centered artificial intelligence (HAI) is defined as AI under human control and AI on the human condition (Yang, Ogata, Matsui, & Chen, 2021). Especially, the educators also need to be sure that the learners can achieve higher learning performance when AI is reasonably reliable. In practice, Lu, Huang, and Yang (2021) raised a typical case of machines losing reasonable reliability in which machines can ignore some risk students because the teacher adopted a leniency grading policy.

The proposed study was conducted to implement AI-based applications in the education field and focused on practicing the short-answer questions for three reasons. First, Hwang, Xie, Wah, and Gašević (2020) collected the research issues in AIED and found that improving students' learning performance using AI-based solutions is an important research topic that can be second only to designing AI tools. The proposed study believed in short-answer questions, which is one of the best ways to implement AI-based solutions, and the detailed explanation on the implementation is described in the following section. Second, according to prior studies, most research on the ITS has been based on numerical data driven applications, for example, Jovanović, Gašević, Dawson, Pardo and Mirriahi (2017) used students' login times per week as the data to construct the self-regulated learning model. Natural language processing (Chowdhary, 2020), and speech recognition (Deng, Hinton, & Kingsbury, 2013) have been proved to be useful in research, but they have not been adopted in education practice. For our short-answer system, natural language processing is fundamental to its operation. In natural language processing, semantics-based or syntax-based question-generation algorithms can be applied (Greving & Richter, 2018), further discussion of related studies in the Literature Review section. Third, Luckin, Holmes, Griffiths and Forcier (2016) mentioned that most of the current ITS designs are student-oriented. However, the gaps in the current AIED research should also consider the teacher's retention rate on ITS. If teachers are encouraged to design most learning activities on ITS, students can obtain the expected benefits in ITS. Therefore, in the real scenario, if the machine can be used to replace the teacher to evaluate students' learning performance, it is expected to increase the teacher's willingness to use.

1.2. The benefits of short-answer questions

The question and answer (Q&A) process yields benefits in many fields; for instance, medical diagnosis and computer system security usefully apply Q&A (Kaur & Bathla, 2015). In the education field, benefits of the Q&A process include (1) allowing student to use question-based practice to construct knowledge, (2) identifying student misunderstandings through learner feedback, (3) guiding learners to pay more attention to key material, (4) repeating concepts to enhance memory, (5) motivating learners to engage in the course, and (6) enabling teachers to understand the learning performance of each learner (Kaur & Bathla, 2015; Kurdi, Leo, Parsia, Sattler, & Al-Emari, 2020). Comparing with other exam methodologies, studies have demonstrated the efficiency of short-answer questions is superior to other modes of examination. For instance, Smith and Karpicke (2014) conducted an experiment with 80 students to investigate the effects of short-answer and multiple-choice questions on retrieval ability, and the results indicated that short-answer questions produced better learning performance due to the higher memory retrieval capability. Rush, Rankin and White (2016) demonstrated that answering short-answer questions requires learners to have a higher level of cognition than answering multiple-choice questions, which means that the learner is required to focus more on the review process. Furthermore, Greving and Richter (2018) recommended short-answer questions because practice with such questions improved student ability to retrieve material from their memories.

Repetitive or frequent requests students to evaluate the knowledge taught in the classroom is an advanced application of short-answer in education, commonly known as "practice testing" or "repeated testing" (Adesope, Trevisan, & Sundararajan, 2017; Wiklund-Hörnqvist, Jonsson, & Nyberg, 2014). Many studies have reported that short-answer-based practice tests elicit strong student performance. For instance, McDermott, Agarwal, D'Antonio, Roediger III, and McDaniel (2014) conducted four experiments with 512 participants and assigned different test plans for the experimental groups; the results indicated that frequent quizzes can improve students' learning outcomes and retention rates. Moreover, Larsen, Butler and Roediger III (2009) investigated the effect of repeated study on final recall, and their experimental results indicated that adopting a repeated study strategy led to higher final scores. Despite its evident value, creating short-answer practice material is time consuming and thus burdensome for teachers. Employing automatic question generation (AQG) may be a solution to this problem; in this technique, question–answer pairs are generated through analysis of a given text (Rus, Cai, & Graesser, 2008), however, the concept that applying AQG techniques into classroom for the practice testing only implemented in the laboratory settings (Greving & Richter, 2018) only.

In summary, implementing AI can benefit students and teachers in the field of education. Due to advancements in its technology, natural language processing may be able to help teachers easily generate short-answer practice tests for classroom use. Therefore, we adopted machine learning to create short-answer questions and investigated whether the generated questions had acceptable quality and whether students benefited from studying with such questions. Our research questions were as follows:

RQ1: Can students improve their learning performance with repeated short-answer question practice?
RQ2: In evaluating students' programming skill, do machine-generated questions exhibit similar quality to expert-authored questions?

RQ3: In evaluating students' programming skill, does machine-grading exhibit similar quality to expert-grading?

2. Literature review

AQG can be used to generate various types of questions, such as cloze questions and multiple-choice questions (Ch & Saha, 2018). We used AQG to create short-answer questions because such questions benefit students' long-term memory (Greving & Richter, 2018). The concept of AQG was defined by Le, Kojiri, & Pinkwart (2014) as: "*generating questions from various inputs such as raw text, database or sematic representation*" (p. 352), and it has been a popular research topic in recent years due to the emergence of natural language processing by neural networks, which is designed to mimic how human beings use language and serve as a tool for manipulating human language to meet specific requirements (Chowdhary, 2020). At least three related systematic reviews have been retrieved from the library system in past three years (Ch & Saha, 2018; Kurdi et al., 2020; Papasalouros & Chatzigiannakou, 2018), from which we have gained two valuable information: (1) approaches to implement AQG system, and (2) quality evaluation methods.

According to the systematic review of Kurdi et al. (2020), an AQG system can be implemented through four approaches, but only two of these methods account for more than 70% of instances of implementation. The first one is syntax-based approach, which extracted features such as: part-of-speech, and then select distractors based on a classification algorithm for constructing question sentences (Das & Majumder, 2017). The second approach is based on semantics and depends upon a comprehensive understanding of the context and additional information or knowledge to select meaningful sentences for constructing question sentences (Chan & Fan, 2019; Yao, Bouma, & Zhang, 2012). The other methods are limited by sentence patterns, and therefore, we only considered syntax-based and semantic-based approaches in this study and propose an ensemble method that combines semantic and syntactical approaches to automatically generate questions. For semantics, our system uses BERT (Bidirectional Encoder Representations from Transformers) (Devlin, Chang, Lee, & Toutanova, 2018), the syntax part uses Stanford CoreNLP (Manning, Surdeanu, Bauer, Finkel, Bethard, & McClosky, 2014), and the question construction part uses GTP2 (Generative Pre-Training)(Radford, Narasimhan, Salimans, & Sutskever, 2018). The main reason is that the above methods took transfer learning (Pan & Yang, 2009) approach, which allows follow-up developers to produce a domain specific model without collecting a large dataset, and the methods employed perform well in machine reading comprehension. More details about the collaboration between BERT, Stanford CoreNLP and GPT2 will be introduced in section 0

Another major concern is how to evaluate the quality of AQG methods. Most studies in this field have adopted a standard dataset for evaluating performance, with one of the most popular datasets being the SQuAD (Sandford Question Answering Dataset) (Ch & Saha, 2018), which consists of 100,000 question-answer pairs collected from Wikipedia articles. The SQuAD has been used in BERT, which we used in this study, and several pretraining models such as UNILM (Dong et al., 2019) or Glomo (Yang, Zhao, Dhingra, He, Cohen, William, Salakhutdinov & LeCun , 2018). On the other hand, how did priori studies quantify the performance evaluation results? Practically, a comparison will be performed between the questions generated through the proposed method and some other methods from related works, and the questions in SQuAD dataset will be served as the ground truth during the comparison process. Various metrics will be used for the quantify the comparison results—for example, BLEU (Bilingual Evaluation Understudy) (Papineni, Roukos, Ward, & Zhu, 2002) or ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004).

Through the above studies, we can observe that BERT, CoreNLP and GPT2 exhibit outstanding performance in semantic-based and syntax-based question generation (Klein & Nabi, 2019). The results guide us to consider those approaches as the implementation of the AQG methods. However, because the input to those pre-training models was the teacher's teaching material, no standard answers or ground truth were available for BLEU or ROUGE to use to evaluate the quality of interrogative sentences. Therefore, we reviewed the most typical evaluation approach: the Turing test (Turing, 2009), which was proposed by Alan Turing in 1950. In the test process, Turing suggest to assign an evaluator to judge the messages that delivered by human or machine through a dialog, and expect the evaluator cannot judge the difference between human and machine due to the machine present similar response as human. Several studies adopted Turing test in order to prove the performance; for example, (Hingston, 2009) designed a game bot, and after five rounds of games where machines imitated humans, they analyzed game behavior. The result of the analysis declared: "*Computers cannot play like humans—yet*." In another instance, Alarifi, Alsaleh, & Al-Salman (2016) proposed a classifier by using the graph theory to detect fake identities on social network. To demonstrate the performance under the situation that lack of the ground truth datasets, they used the Turing test as well. Finally, in the summary report compiled by Kurdi et al. (2020), they also recommended using the expert review process for assessing machine-generated

questions. Thus, to evaluate the quality of our AQG method, we adopted the Turing test approach, which is detailed in the next section.

3. Method and experiments

3.1. Participants

This study conducted an empirical experiment to assess the impact of practice testing on learning performance. The experiment was executed in a freshman university programming course during three weeks in October 2020, and the primary course content was the basic Python programming language. A total of 91 students from two classes participated in the experiment. All the participants were students in the computer science department. We divided the students into two groups: the first one was the control group with 50 students, and they learned through conventional learning activities; the second one was the experimental group with 41 students, and they learned through practice testing.

3.2. Experimental design and learning activities

The design of the experiment adopted in this study is shown in Figure 1. The learning activity was divided into three phases: initial phase, course instruction, and performance evaluation. Before the course began, the teacher uploaded the learning materials to **BookRoll** (Flanagan & Ogata, 2017). In the **initial phase**, the teacher assigned a pre-test to evaluate the students' programming skills at the beginning of the course; in the next step, instruction in the if–else programming syntax was given and practice was assigned to the experimental group. The practice in this step was delivered by the short-answer system we proposed in this study (explained further in the next section). The second phase was the **course instruction phase**, during which only general classroom activities were conducted and the experimental group completed the practice test generated by the machine, as in the initial phase. The third phase was the **performance evaluation phase**. In addition to regular instruction activities and practice tests for the experimental group, both groups took a post-test to evaluate their programming skill; the short-answer questions used in the post-test are listed in the Appendix I. The grading results of the pre-test and post-test were compared. Furthermore, the post-test was graded by both experts and machine, respectively, and the grading results by expert will be the ground truth to (1) compare with the pre-test to investigate students' learning performance improvements and (2) compare with machine-grading to evaluate the quality of it.

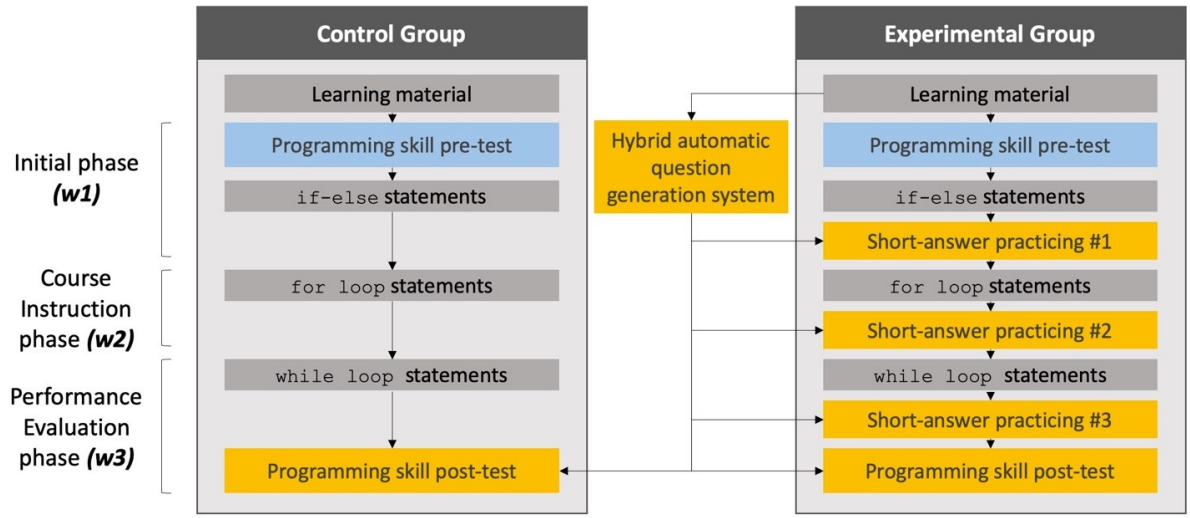


Figure 1. Experimental design and learning activities

3.3. Hybrid automatic question generation (Hybrid-AQG) system

The AQG system proposed by this study was combined sematic analysis and syntax analysis, it provides two functions: machine-questioning and machine-grading. Figure 2 shows the user interface designed in this study, which allows the instructor to review and modify questions generated by the machine and students to respond to the questions. The design principle of machine questioning is to let the machine understand the learning

material's content and generates question sentences. It consists of semantic analysis and syntactic analysis; therefore, the system here we named as Hybrid-AQG, and we listed questions that generated by the system in the Appendix II.

As shown in Figure 2, the primary goal of semantic analysis is to make the machine read the learning materials uploaded by the instructor and extract the keywords. This study used BERT (Devlin et al., 2018), it is a pre-trained model by using a large number of labeled data and allow developers to fine-tuning the model parameters (Pan & Yang, 2009). On the other hand, to make the general purpose pre-trained model understand more about Python language, here we adopted a Kaggle dataset which contains 40.1M question-answer pairs and tags as the training dataset for fine-tuning (Python Questions from Stack Overflow: https://www.kaggle.com/stackoverflow/pythonquestions). The extracted keywords then served as the answers to the questions; later, in the machine-grading part of the study, these keywords were used to evaluate students' responses. Syntactical analysis was also implemented, the goal of which is to extract sentences from a paragraph. For this, we used Stanford CoreNLP (Manning et al., 2014) to compose a tree structure to find complete sentences containing a subject, verb, and target.

Because of the syntactical analysis output a declarative sentence with keywords only, but we only need a sentence without keywords to generate the interrogative sentences. Therefore, we transformed the declarative sentences into interrogative sentences. Practically, we first removed the keyword specified by sematic analysis, and then we adopted the GPT2 (Radford et al., 2018) which is a machine learning technology that uses unsupervised learning to generate reasonable words according to the meaning of the context. As shown in Figure 2, we fed a sentence into the GPT2 model, and the model predicted the following word, "what"; thus, an interrogative sentence was achieved.

Next, questions and answers that had already been generated by the machine were provided to students to practice in class. Then, BERT was used in the machine-grading method to calculate the distance between sentences through sematic understanding. Accordingly, to know whether the keywords identified in the previous step were consistent with the answers of the students, we fed two sentences into BERT and received a quantified result, and this is the implementation of the machine-grading.

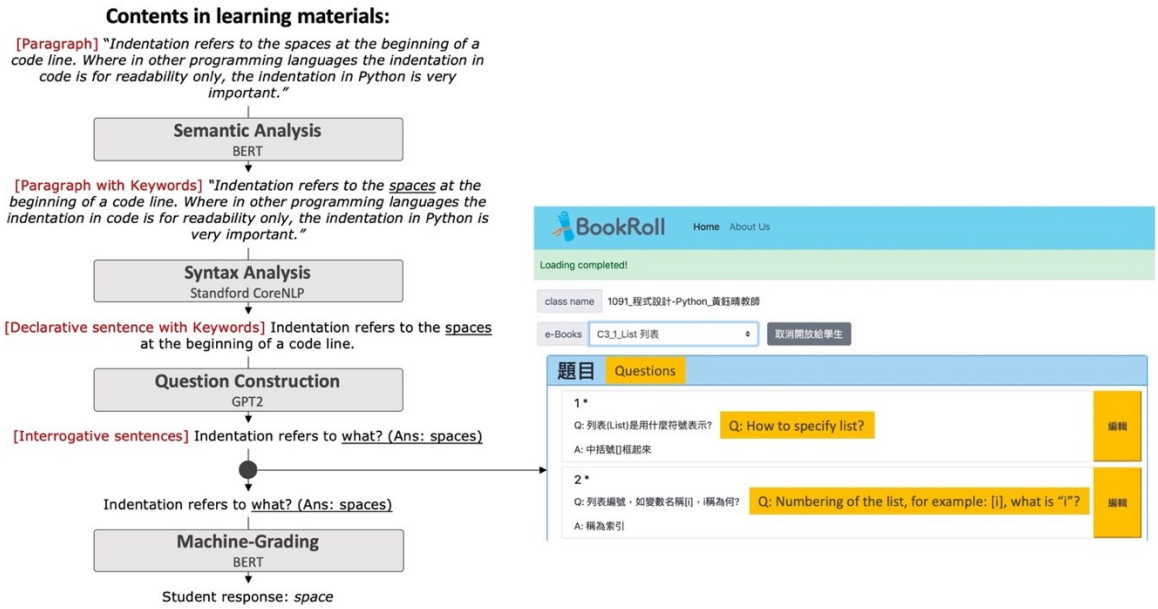


Figure 2. Data flow and screenshot of the Hybrid-AQG system

3.4. Evaluation the quality of machine-question and machine-grading

In the post-test stage, we conducted multiple evaluations by using a question jointly produced by the expert and the machine. We (1) compared results of the post-test with those of the pre-test to evaluate whether the students' programming ability had improved and to discern the difference between the experimental group and the control group, (2) evaluated the quality of machine-generated questions, and (3) evaluated the quality of machine grading. The first two items needed to be compared before and after to reduce the experiment's deviation, and

thus, the same expert scored both the pre-test and post-test. The following two subsections and Figure 3 explain the details of the assessment of items two and three.

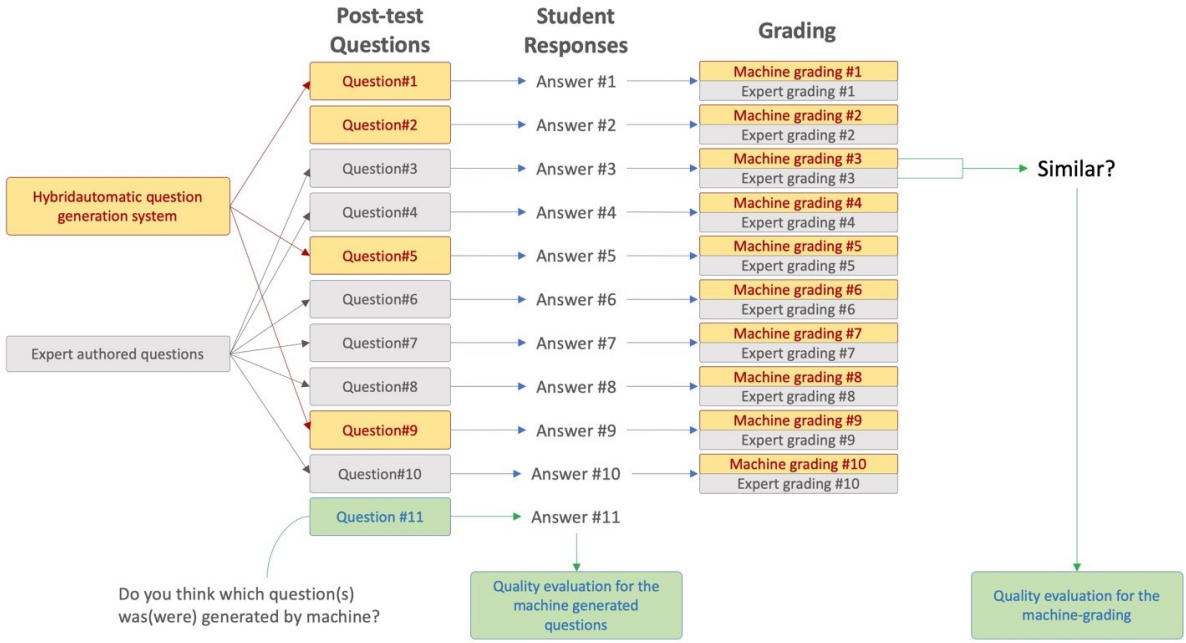


Figure 3. Flow in post-test for evaluating the quality of the machine-generated questions and machine-grading

Inspired by the Turing test, to evaluate the quality of machine-generated questions, we designed a test that featured both machine-generated questions and expert-authored questions and then evaluated whether students could distinguish them. To evaluate students' programming skills in the post-test, a total of 11 questions were presented; except for Questions 1, 2, 5, and 9, the rest were expert-authored questions. The main reason for only using machine question generation for 4 of 10 questions was because the Turing test requires machines to be involved in just 1/3 to 1/4 of an entire test. Further, in Question 11, we asked students to identify which question(s) was(were) generated by a machine to determine whether they could distinguish authorship. By contrast, if the machine-generated questions are qualified, we expected that the students could not answer the correct answer. Finally, in the post-test stage, to correctly quantify the students' programming skills and compare their results with their pre-test scores, the results of Question 11 were not considered.

Inspired by the Turing test, to evaluate the quality of machine-generated questions, we designed a test that featured both machine-generated questions and expert-authored questions and then evaluated whether students could distinguish them. To evaluate students' programming skills in the post-test, a total of 11 questions were presented; except for Questions 1, 2, 5, and 9, the rest were expert-authored questions. The main reason for only using machine question generation for 4 of 10 questions was because the Turing test requires machines to be involved in just 1/3 to 1/4 of an entire test. Further, in Question 11, we asked students to identify which question(s) was(were) generated by a machine to determine whether they could distinguish authorship. By contrast, if the machine-generated questions are qualified, we expected that the students could not answer the correct answer. Finally, in the post-test stage, to correctly quantify the students' programming skills and compare their results with their pre-test scores, the results of Question 11 were not considered.

Table 1. Confusion matrix to evaluate the quality of machine-questioning and machine-grading

	Evaluating machine-questioning quality		Evaluating machine-grading quality	
	Student distinguish	Actual	Machine classified	Expert confirmed
True-Positive (TP)	Machine-generated	Machine-generated	Correct	Correct
False-Positive (FP)	Machine-generated	Expert-authored	Correct	Incorrect
False-Negative (FN)	Expert-authored	Machine-generated	Incorrect	Correct
True-Negative (TN)	Expert-authored	Expert-authored	Incorrect	Incorrect

To quantify machine-generated question quality, we treated the answers to Question 11 as a binary classification problem and applied a confusion matrix for comparison. Four combinations are listed in Table 1, and we calculated accuracy, precision, and recall in light of responses designated as true positive (TP), false positive

(FP), true negative (TN), or false negative (FN) by using the following equations. This test used accuracy, recall and precision to evaluate the quality of machine-generated questions.

$$\text{Accuracy}=\frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

$$\text{Recall}=\frac{TP}{TP+FN} \quad (2)$$

$$\text{Precision}=\frac{TP}{TP+FP} \quad (3)$$

In evaluating machine-generated question quality by using the confusion matrix, accuracy indicated the ratio of the 10 questions correctly identified. If the accuracy was close to 1, the students could distinguish the questions generated by the machine, which would mean machine performance was not acceptable. However, if accuracy was close to 0.5, this would mean the quality of machine-generated questions approached that of expert-authored questions. Recall indicated the proportion of correctly identified machine-generated questions. A higher recall value indicated a higher rate of correctly identified questions. Precision referred to the proportion of number of questions that students think is machine-generated in actual number of machine-generated questions. The higher the precision value was, the higher was the ratio of correctly identified machine-generated questions.

To evaluate whether machine and expert grading was of similar quality, we adopted a confusion matrix, too. For each answer, whether an expert or a machine checked it, a binary result was given: "correct" or "incorrect." Then, we took the confirmed results from the expert as the ground truth; the four combinations are listed in Table 1. Accuracy, recall, and precision were again applied for assessment. Accuracy close to 1 suggested close similarly between expert and machine grading, but accuracy close to 0.5 suggested inconsistency. Recall indicated the ratio of answers correctly graded by experts to those correctly graded by machine learning. Precision indicated the ratio of correct answers verified by expert-grading.

4. Results

4.1. Reply RQ1 (Can students improve their learning performance with repeated short-answer question practicing?)

To measure students' learning performance, the teacher conducted a pre-test and post-test to evaluate programming skills in the first week and the third week. We used the independent samples *t*-test to assess the difference in learning performance between the control and experimental groups in the pre-test. Table 2 lists the results of the descriptive statistics and independent samples *t*-test of the pre-tests of the control and experimental groups. The scores for the pre-test of programming skills in the control and experimental groups were 77.0 and 73.38, respectively. The results listed in Table 2 indicate that the pre-test scores for the experimental group and control group did not differ significantly ($t = -1.117$, $p > .05$). This means that the students' programming skills were equal in the control and experimental groups.

Table 2. Statistics results and independent sample *t*-test of pre-test for the control group and the experimental group

Group	*N*	Mean	*S.D.*	*t*	*p*
Experimental group	41	73.38	16.76	-1.117	.267
Control group	50	77.0	14.18		

Note. $^{*}p < .05$; $^{**}p < .01$; $^{***}p < .001$.

This study investigated the impact of repetitive short-answer practice on students' learning performance by using analysis of covariance (ANCOVA) to exclude the difference in programming skills of the control and experimental groups. The pre-test and post-test scores for programming skills were used as the covariate and dependent variables in ANCOVA, respectively. The result of Levene's test did not violate the homogeneity of variance ($F = 601$, $p = .440$), meaning that ANCOVA was applicable.

Table 3. Statistics results and ANCOVA of post-test for the control group and the experimental group

Group	*N*	Mean	*S.D.*	Adjusted Mean	*S.D.* Error	*F*	*p*
Experimental group	41	88.78	10.97	89.12	2.21	12.73	.000***
Control group	50	78.75	16.42	78.74	2.00		

Note. $^{*}p$ <.05; $^{**}p$ < .01; $^{***}p$ < .001.

Table 3 presents the descriptive statistics results and ANCOVA of the post-test for the control group and the experimental group. The adjusted means of the post-test scores in programming skills for the control and experimental groups were 78.74 and 89.12, respectively. According to the ANCOVA result, the experimental group had significantly higher post-test scores than did the control group (F = 12.73, p = .00). The results demonstrated that students can effectively improve their learning performance in programming skills through use of the Hybrid-AQG, or more specifically, the repetitive short-answer practice by machine-generated questions. Our results were consistent with those of prior studies that found that repetitive practice can enhance students' long-term memory to drive subsequent improvements in learning performance (Karpicke, 2017; Roediger III & Karpicke, 2006; Rowland, 2014), especially when short-answer practice is applied in the higher education context (Greving & Richter, 2018). Moreover, it is inevitable for students to be familiar with the topic for their performance, but it does not mean that the content of the questions is qualified. Therefore, we will continue to discuss the quality of machine-questioning and machine-grading in the following sessions.

4.2. Reply RQ2 (To evaluating students' programming skill, does machine-generated questions have similar quality with the expert-authored questions?)

This study adopted an evaluation process based on the Turing test to investigate the ability of students to identify machine-generated questions. The teacher designed Question 11 in the post-test, which asked students to identify which questions were generated by a machine. Figure 4 presents the results for the ability of students to distinguish machine- from expert-authored questions. Four questions, namely 1, 2, 5, and 9, generated by machine were correctly distinguished by 13 (32%), 22 (54%), 12 (29%), and 12 (29%) students, respectively, in the experimental group, and 9 (18%), 12 (24%), 9 (18%), and 16 (32%), respectively, in the control group. These results indicate that in experimental group, a higher proportion of students could correctly distinguish between the machine- and expert-authored questions, which we attribute to the students in the experimental group having already seen the patterns of machine-generated questions when using the short-answer practice system. This provides evidence that the experimental group students created long-term memories during repetitive practice.

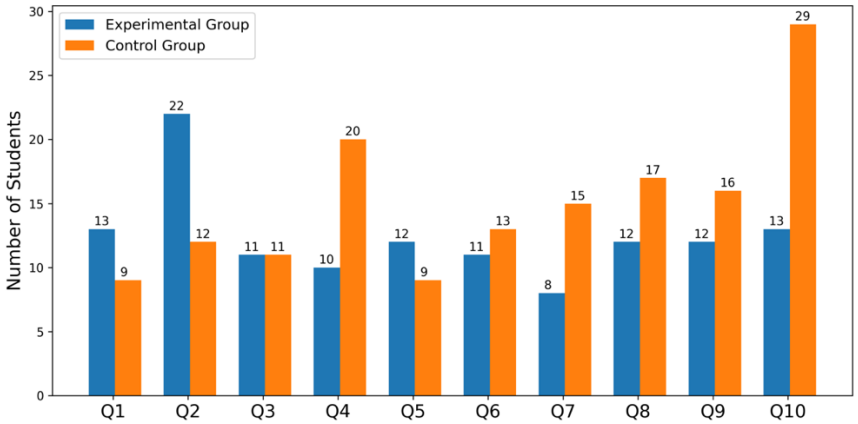
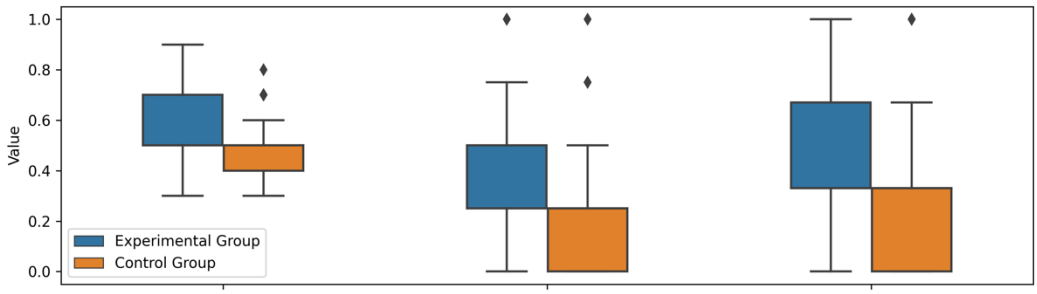


Figure 4. Distinguishing student results for machine- and expert-authored questions

We applied the confusion matrix to quantify the performance of students in distinguishing between machine-generated and expert-authored questions in the post-test to evaluate the quality of the machine-generated questions. Figure 5 presents the accuracy, precision, and recall of guessing results for the experimental and control groups in the post-test. The accuracy (t = −3.7, p < .001), precision (t = −2.48, p < .05), and recall (t = −2.53, p < .05) values of the experimental group were significantly higher than those of the control group.

The average accuracy of the control group is .48, which means that the control group answered questions 11 almost answering by guessing. Thus, the students in the control group could not distinguish which questions were generated by a machine. Compared with the control group, the experimental group exhibited a higher average accuracy: .585. Studies have indicated that practice can enable students to construct knowledge and that repeat practice using short-answer questions can enhance students' retrieval of information from memory (Kaur

& Bathla, 2015; Kurdi et al., 2020). We attribute the higher values of accuracy, recall, and precision in the experimental group to the students in the experimental group having had practice with similar machine-generated questions in the Hybrid-AQG system; such practice had a positive effect on their quality of review and deepened their long-term memory of the machine-generated questions. This is consistent with the observation in Greving and Richter (2018) study that short-answer practice can help students retrieve material from memory. However, even though the students in the experimental group had seen the machine-generated questions, the accuracy, precision, and recall still only reached .585, .436, and .36, respectively, indicating that the machine-generated questions were similar to expert-authored questions. Thus, we conclude that students perceive machine-generated questions and expert-authored questions similarly, indicating that the machine-generated questions are suitable for practice testing.



	Accuracy		Recall		Precision	
	Experimental	Control	Experimental	Control	Experimental	Control
Mean	0.585	0.482	0.36	0.23	0.436	0.285
Std.	0.139	0.127	0.268	0.231	0.299	0.275
t-Test	-3.70***		-2.48*		-2.53*	

Figure 5. Distinguishing experimental and control group results for machine- and expert-authored questions by using the metrics of accuracy, precision, and recall

Recall was evaluated as the ratio of guesses that correctly identified the four machine-generated questions. For students in the experimental group, the values of TP, FP, FN, and TN were 59, 65, 105, and 181, respectively. For students in the control group, the values of TP, FP, FN, and TN were 46, 105, 154, and 195, respectively. Figure 5 indicates that the precision values for the experimental group and the control group (experimental group: .436; control group: .285) were higher than the recall values (experimental group: .36; control group: 0.23), meaning that students in both the experimental group and the control group struggled to distinguish which questions were machine-generated. In both the experimental and control groups, the values for precision were higher than those for recall, as evidenced by fewer FPs than FNs. FN meant that a student distinguished that the question was generated by an expert, but it was actually a machine-generated question. Higher FN values indicated that students tended to treat machine-generated questions as expert-authored questions. This may have been because machine-generated questions were similar to expert-authored questions, and thus, students struggled to distinguish them—hence, the lower recall value. This outcome indicates that the text content used by the machine when generating the questions (using teaching materials and natural language processing) was quite close to the text content used by the expert when designing questions, meaning that the machine-generated questions in this study are suitable for practice testing due to students being unable to distinguish between machine-generated and expert-authored questions.

Table 4. Correlation analysis of the number of correct answers and the number of students who identified the question as machine generated

	Mean/Std. of students		Spearman correlation	
	Number of answer the question correctly	Number of students identified machine-generated question	Coefficient	p-value
Experimental	35.50/4.53	9.60/3.84	.09	.79
Control	30.80/12.88	9.90/7.81	.83	.003**

Note. $^{*}p$ <.05; $^{**}p$ < .01; $^{***}p$ < .001.

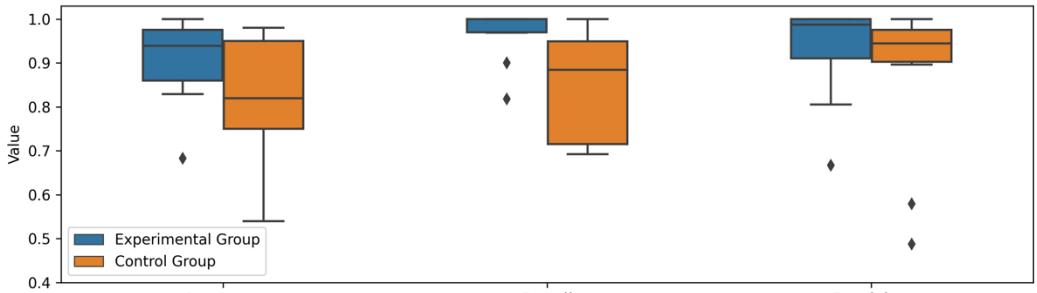
To explore why the students are hard to distinguish the questions are generated from machine or expert, the teacher interviewed the control group students how they setup the identification rules. Most students replied that computers are not as smart as humans, and therefore, they adopted an identification rule that sought the simplest questions in the list. Therefore, in order to continuous explore the quality of machine-questioning, we have to proof students in the control group adopted an identification rule like looking for the simplest question, we use Spearman correlation analysis to explore the relationship between the number of students answering correctly and the number of students who identified that the question is belongs to machine-generated. Table 4 lists the

descriptive statistics of Spearman correlation analysis results between answering correlation analysis between the rate of answer the question correctly and machine-expert identification rate.

As evident in Table 4, the number of students in the control group who answered a question correctly and the number of students who identified the question as being machine generated had a significant correlation ($r = .83$, $p > .01$). This result shows that for the students in the control group, more students answered simple questions correctly, which they identified as machine generated. This finding is consistent with what the students described as their identification rule. This suggests that without the benefit of the Hybrid-AQG system, the students defaulted to identifying machine-generated questions by their simplicity. By contrast, the experimental group exhibited no such correlation between correct answers and machine- generated question identification ($r = .09$, $p > .05$), which we attribute to students having practiced with the Hybrid-AQG system, enabling students to identify whether a question was machine-generated or expert-authored based on memory. This phenomenon evident in the experimental group is consistent with research results (Greving & Richter, 2018) indicating that the Hybrid-AQG system can enable students to retrieve more material from memory.

4.3. Reply RQ3 (To evaluating students' programming skill, does machine-grading have the similar quality with the expert-grading?)

To measure the quality of machine grading in the post-test, this study used a confusion matrix to evaluate the difference between machine grading and expert grading. The process of evaluating machine-generated and expert-authored questions was described in detail in Methods section. Figure 6 presents the accuracy, recall, and precision of machine grading quality; the accuracy ($t = 4.135$, $p < .001$), precision ($t = 2.084$, $p < .05$), and recall ($t = 4.689$, $p < .001$) values for the experimental group were significantly higher than those for the control group.



	Accuracy		Recall		Precision	
	Experimental	Control	Experimental	Control	Experimental	Control
Mean	0.907	0.822	0.96	0.842	0.913	0.856
Std.	0.096	0.1	0.079	0.144	0.125	0.132
t-Test	4.135***		4.689***		2.084*	

Figure 6. Similarity analysis between expert-grading and machine-grading, in metrics of accuracy, precision and recall in between experimental and control groups

The accuracy of the experimental group and the control group was .907 and .822, respectively. The results of the independent samples *t*-test of accuracy demonstrate that the experimental group was significantly more accurate than the control group ($t = 4.135$, $p < .001$). The main reason is that the content answered by the students in the experimental group makes the machine more interpretable. This may be the result of repeated practice by the students in the classroom. We infer that repeated practice using the Hybrid-AQG system can enhance the long-term memory of students, which is why the accuracy of machine grading and expert grading of the experimental group was higher than that for the students in the control group. This result combined with the results of the analysis of RQ1, which indicated that students in the experimental group had significantly higher learning performance in the post-test than students in control group did, lead us to conclude that practicing short-answer questions can enhance the long-term memory of students (as evidenced by the experimental group performance) and further improve their academic performance. These benefits of the Hybrid-AQG system are consistent with the results of (Greving & Richter, 2018), which suggested that repetitive practice can enhance student's ability to retrieve information from memory.

In this study, the recall values of the experimental group and the control group were.96 and .84, respectively, meaning that machine grading and expert grading were highly consistent for correct answers; thus, machine grading can replace expert grading to some extent. In this study, the precision values of the experimental group and the control group were 0.91 and .85, respectively, meaning that machine and expert grading are highly consistent for answers that a machine grades as correct.

For the experimental group, the values of TP, FP, FN, and TN were 318, 27, 11 and 54, respectively. The value of recall was higher than that of precision due to fewer FNs than FPs. FPs may have been because the machine identified the correct answer, but the answer contained only conceptual keywords rather than complete and clear content, causing experts to think the answer was wrong. To investigate the reasons for the FP-type answer, we examined students' FP-type answers and found that because the answer content contained keywords related to the concepts covered by the question, it deemed the correct answer in machine grading; however, because the answer content was not complete, experts graded it as a wrong answer. For example, one question asked, "*What is on the right side of the equal sign when assigning a value to a variable?*" In the FP answer, the content example was the actual content value. The machine-based grading was mainly based on whether the content value was mentioned in the answer, and thus, the machine-grading system rated this answer as correct, but the expert thought that the content value of the variable must be clearly stated, and thus, the answer was considered to be incomplete and rated as wrong. From this example, we suggest that the expert grading may be stricter than the machine-based grading. This may account for why the number of FPs was greater the number of FNs and may also be the reason that the value of recall was greater than precision in the experimental group.

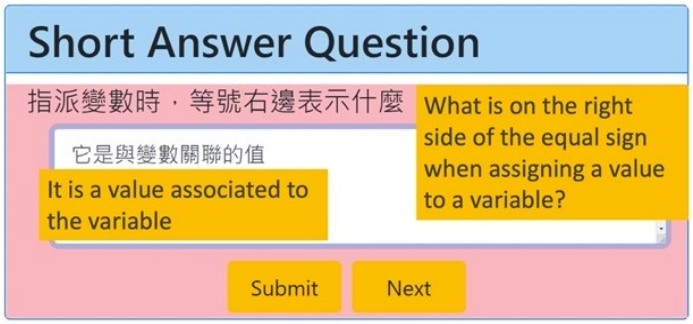


Figure 7. False-Negative machine grading example

For the students in the control group, the values of TP, FP, FN, and TN were 257, 38, 51, and 154, respectively. Because fewer FPs were recorded than FNs, the precision value was higher than the recall value. FN indicated the machine classified the answer as incorrect, but the expert confirmed the answer as correct. Here we provide an example as shown in Figure 7: "*What is on the right side of the equal sign when assigning a value to a variable?*" One FN answer from student is: "*It is a value associated to the variable,*" and the issue in this sentence is the pronoun "it". Because experts know that the pronoun "it" in the answer refers to a variable but the machine learning system has no context to conclude this, it fails to understand what the pronoun in the answer refers to. Therefore, the machine-based grading regards the answer as wrong. For the students in the control group, because they did not use the short-answer practice system and had never experienced machine grading, they did not know how to answer the answer with content that the machine could understand. Therefore, most answers were incorrectly scored in machine grading. This is why the precision value of the students in the control group was higher than their recall value.

5. Conclusions

The technical applications of modern AI are diverse, such as computer vision or speech recognition. This study focuses on natural language processing, aims to implement AI applications for the education purpose and look forward to the benefits of emerging AI technologies that can bring into education. To this end, this study proposed Hybrid-AQG based on the advanced transfer learning technologies BERT, GPT2, and CoreNLP. The system can perform semantic and syntactical analysis of a teacher's teaching materials, generate multiple question–answer pairs, and enables students to engage in repeat practice of questions after class. Through implementation of this system, a teacher's burden is reduced and students' long-term memory of course content can be enhanced.

After a 3-week experiment, we verified three hypotheses through data analysis. First, repetitive practice was proven to be beneficial to students' long-term memory for subsequent improvements in learning performance, even when using practice questions generated by a machine. Second, in our experiment, only some students could identify when a question was machine rather than expert generated because of long-term memory; however, most students could not distinguish them. This reflects the maturity and usefulness of combining semantic and syntax approaches for generating questions. In the control group, students simply defaulted to identifying simple questions a machine generated. Last, this study employed a semantic method to implement the machine-grading functionality. However, it turned out that grading short-answer questions still requires

technology that can understand the context and order of keywords, otherwise, only keywords check can be achieved in current study.

Finally, there are two limitations to this study. The first is that we didn't discuss the teaching materials provided by teachers. Still, the quality of the teaching materials and the format setting will affect the machine-generated questions' quality. For example, some teachers preferred to use pictures and even sample code in teaching materials, and both ways presentation will cause some garbled information in the output of machine-questioning, and it required the teacher to review and remove. The second limitation is the issue of the question-type. In this study, we only used short-answer questions; however, other popular types need to be verified the effectiveness and quality, such as the multiple-choice questions and cloze questions.

Acknowledgement

This work is supported by Ministry of Science and Technology, Taiwan under grants MOST-109-2511-H-008-007-MY3, MOST-108-2511-H-008-009-MY3, MOST-110-2511-H-153-001, and Ministry of Education, Taiwan.

References

Adesope, O. O., Trevisan, D. A., & Sundararajan, N. (2017). Rethinking the use of tests: A Meta-analysis of practice testing. *Review of Educational Research, 87*(3), 659–701.

Alarifi, A., Alsaleh, M., & Al-Salman, A. (2016). Twitter turing test: Identifying social machines. *Information Sciences, 372*, 332–346.

Ch, D. R., & Saha, S. K. (2018). Automatic multiple choice question generation from text: A Survey. *IEEE Transactions on Learning Technologies, 13(*1), 14–25.

Chan, Y.-H., & Fan, Y.-C. (2019). A Recurrent BERT-based model for question generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering* (pp. 154–162). doi:10.18653/v1/D19-5821

Chen, X., Xie, H., & Hwang, G.-J. (2020). A Multi-perspective study on artificial intelligence in education: Grants, conferences, journals, software tools, institutions, and researchers. *Computers and Education: Artificial Intelligence*, 100005. doi:10.1016/j.caeai.2020.100005Get

Chowdhary, K. R. (2020). Natural language processing. In *Fundamentals of artificial intelligence* (pp. 603–649). Springer. doi:10.1007/978-81-322-3972-7_19

Collischonn, W., & Pilar, J. V. (2000). A Direction dependent least-cost-path algorithm for roads and canals. *International Journal of Geographical Information Science, 14*(4), 397–406.

Das, B., & Majumder, M. (2017). Factual open cloze question generation for assessment of learner's knowledge. *International Journal of Educational Technology in Higher Education, 14*(1), 1–12.

Deng, L., Hinton, G., & Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: An Overview. In *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 8599–8603). doi:10.1109/ICASSP.2013.6639344

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding*. Retrieved from https://arxiv.org/abs/1810.04805

Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., & Hon, H.-W. (2019). Unified language model pre-training for natural language understanding and generation. Retrieved from https://arxiv.org/abs/1905.03197

du Boulay, B. (2016). Artificial intelligence as an effective classroom assistant. *IEEE Intelligent Systems, 31*(6), 76–81.

Flanagan, B., & Ogata, H. (2017). Integration of learning analytics research and production systems while protecting privacy. In *Proceedings of the 25th International Conference on Computers in Education* (pp. 333–338). Christchurch, New Zealand: Asia-Pacific Society for Computers in Education.

Greving, S., & Richter, T. (2018). Examining the testing effect in university teaching: Retrievability and question format matter. *Frontiers in Psychology, 9*, 2412. doi:10.3389/fpsyg.2018.02412

Hingston, P. (2009). A Turing test for computer game bots. *IEEE Transactions on Computational Intelligence and AI in Games, 1*(3), 169–186.

Hwang, G.-J., Xie, H., Wah, B. W., & Gašević, D. (2020). Vision, challenges, roles and research issues of artificial intelligence in education. *Computers and Education: Artificial Intelligence, 1*, 100001. doi:10.1016/j.caeai.2020.100001

Jovanović, J., Gašević, D., Dawson, S., Pardo, A., & Mirriahi, N. (2017). Learning analytics to unveil learning strategies in a flipped classroom. *The Internet and Higher Education, 33*(4), 74–85.

Karpicke, J. D. (2017). Retrieval-based learning: A Decade of progress. In *Learning and Memory: A Comprehensive Reference* (2nd ed., pp. 487-514), Cambridge, MA: Elsevier.

Kaur, J., & Bathla, A. K. (2015). A Review on automatic question generation system from a given Hindi text. *International Journal of Research in Computer Applications and Robotics (IJRCAR), 3*(6), 87–92.

Klein, T., & Nabi, M. (2019). Learning to answer by learning to ask: Getting the best of GPT-2 and Bert worlds. Retrieved from https://arxiv.org/abs/1911.02365

Kurdi, G., Leo, J., Parsia, B., Sattler, U., & Al-Emari, S. (2020). A Systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education, 30*(1), 121–204.

Larsen, D. P., Butler, A. C., & Roediger III, H. L. (2009). Repeated testing improves long-term retention relative to repeated study: A Randomised controlled trial. *Medical Education, 43*(12), 1174–1181.

Le, N. T., Kojiri, T., & Pinkwart, N. (2014). Automatic question generation for educational applications–the state of art. In *Advanced computational methods for knowledge engineering* (pp. 325-338). doi:10.1007/978-3-319-06569-4_24

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop* (pp. 74–81). Barcelona, Spain: Association for Computational Linguistics.

Lu, O. H., Huang, A. Y., & Yang, S. J. (2021). Impact of teachers' grading policy on the identification of at-risk students in learning analytics. *Computers & Education, 163*, 104109. doi:10.1016/j.compedu.2020.104109

Luckin, R., Holmes, W., Griffiths, M., & Forcier, L. B. (2016). *Intelligence unleashed: An Argument for AI in education.* London, UK: Pearson Education.

Ma, W., Adesope, O. O., Nesbit, J. C., & Liu, Q. (2014). Intelligent tutoring systems and learning outcomes: A Meta-analysis. *Journal of Educational Psychology, 106*(4), 901-918.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 55–60). Baltimore, MD: Association for Computational Linguistics.

McDermott, K. B., Agarwal, P. K., D'Antonio, L., Roediger III, H. L., & McDaniel, M. A. (2014). Both multiple-choice and short-answer quizzes enhance later exam performance in middle and high school classes. *Journal of Experimental Psychology: Applied, 20*(1), 3-21.

Pan, S. J., & Yang, Q. (2009). A Survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering, 22*(10), 1345–1359.

Papasalouros, A., & Chatzigiannakou, M. (2018, July). *Semantic web and question generation: An Overview of the state of the art.* Paper presented at the International Association for Development of the Information Society (IADIS) International Conference on e-Learning, Madrid, Spain.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). Bleu: A Method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 311–318). Retrieved from https://www.aclweb.org/anthology/P02-1040.pdf

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving language understanding by generative pre-training*. Retrieved from https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf

Roediger III, H. L., & Karpicke, J. D. (2006). Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological Science, 17*(3), 249–255.

Rowland, C. A. (2014). The Effect of testing versus restudy on retention: A Meta-analytic review of the testing effect. *Psychological Bulletin, 140*(6), 1432- 1463. doi:10.1037/a0037559

Rus, V., Cai, Z., & Graesser, A. (2008). Question generation: Example of a multi-year evaluation campaign. In *Proceedings of the WS on the QGSTEC*. Retrieved from https://www.cs.memphis.edu/~vrus/questiongeneration/5-RusEtAl-QG08.pdf

Rush, B. R., Rankin, D. C., & White, B. J. (2016). The Impact of item-writing flaws and item complexity on examination item difficulty and discrimination value. *BMC Medical Education, 16*(1), 1–10.

Russell, S., & Norvig, P. (2002). Artificial intelligence: A Modern approach [PowerPoint slides]. Retrieved from https://storage.googleapis.com/pub-tools-public-publication-data/pdf/27702.pdf

Schroeder, N. L., Adesope, O. O., & Gilbert, R. B. (2013). How effective are pedagogical agents for learning? A Meta-analytic review. *Journal of Educational Computing Research, 49*(1), 1–39.

Smith, M. A., & Karpicke, J. D. (2014). Retrieval practice with short-answer, multiple-choice, and hybrid tests. *Memory, 22*(7), 784–802.

Turing, A. M. (2009). Computing machinery and intelligence. In *Parsing the Turing test* (pp. 23–65). Springer. doi:10.1007/978-1-4020-6710-5_3

Wiklund-Hörnqvist, C., Jonsson, B., & Nyberg, L. (2014). Strengthening concept learning by repeated testing. *Scandinavian Journal of Psychology, 55*(1), 10–16.

Woolf, B. P. (2010). *Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning*. Burlington, MA: Morgan Kaufmann.

Yang, S. J. H. (2021). Guest Editorial: Precision education - A New challenge for AI in education. *Educational Technology & Society, 24*(1), 105-108.

Yang, S. J. H., Ogata, H., Matsui, T., & Chen, N.-S. (2021). Human-centered artificial intelligence in education: Seeing the invisible through the visible. *Computers and Education: Artificial Intelligence*, *2*, 100008. doi:10.1016/j.caeai.2021.100008

Yang, Z., Zhao, J., Dhingra, B., He, K., Cohen, W. W., Salakhutdinov, R., & LeCun, Y. (2018). Glomo: Unsupervisedly learned relational graphs as transferable representations. Retrieved from https://arxiv.org/abs/1806.05662

Yao, X., Bouma, G., & Zhang, Y. (2012). Semantics-based question generation and implementation. *Dialogue & Discourse, 3*(2), 11–42.

Appendix I: Post-test questions (where * means the question was generated by machine)

1. *What are the rules of naming variables in Python?
2. *What is the left side of equal symbol when assigning a variable?
3. What is "//" and "**" means?
4. What is the default value of sep parameter in print() function?
5. *What is the data type after a division operation?
6. How to present a Python list in symbol?
7. What is the len() function for a Python list?
8. The program would execute "if" or "else" if the condition is not Ture?
9. *What is the order of the and、or、not operator?
10. What is the for loop means?

Appendix II: Machine-generated questions (where * means the question also listed in the post-test)

1. What format will the program source code be saved in?
2. What does the computer hope to be able to do?
3. What is wrong with the program syntax?
4. What does the interpreter do?
5. What is wrong with the program logic?
6. What is the purpose of learning languages?
7. What are the grammatical errors?
8. *What is the left side of equal symbol when assigning a variable?
9. What is the right side of equal symbol when assigning a variable?
10. What is the variable to set the content value through?
11. *What are the rules of naming variables in Python?
12. What is the variable name?
13. What does String not support?
14. What is stored in memory?
15. What is the data enclosed in?
16. What does the program print the result through?
17. What are the two types of arithmetic operators?
18. What is the highest priority in arithmetic operators?
19. *What is the data type after a division operation?
20. What is the purpose of using single or double quotes?
21. What is the purpose of using continue in the loop?
22. What is the difference between for loop and while loop?
23. What are the definitions of regional variables and global variables?
24. What can happen to the content in the list?
25. What does List use to hold elements?
26. *What is the order of the and、or、not operator?
27. What is the relationship between the grid and the number in the list?
28. How to set if conditional?
29. What does the string use to specify specific characters?
30. What is the string made of?