Education and Information Technologies
https://doi.org/10.1007/s10639-021-10461-y

Question paper generation through progressive model and difficulty calculation on the Promexa Mobile Application

Rishabh Singh, et al. ***[full author details at the end of the article]***

Received: 6 November 2020 / Accepted: 4 February 2021/Published online: 24 February 2021

Abstract

Mobile learning has been increased in past years and has attracted the interests of academicians and educators in the past many years especially in higher education. The mobile-based online test is the buzzing in the current pandemic time. Institutions need to use online learning as a powerful tool for conducting exams and assess the students effectively. Integrating technology in education can be advantageous for universities and help engage better results for students. Therefore, it is important to understand each student their capacities and create a different test based on the required difficulty. Students should be graded based on their capabilities. The purpose of the research study is to develop the progressive model with the calibration of difficulty level according to the student capacity. To achieve the goal, a test of 20 python questions was conducted on 120 students with each question having difficulty given by 8 field experts. To verify the model, 5 categories were formed with different difficulty levels which in turn gave satisfactory results. To find a relation between the initial difficulty and the calculative difficulty based on the student response, a correlation test was conducted. After careful analysis of the question difficulty and student responses, it was observed that both are highly dependent on each other wherein the difficulty level of any question can be calculated using incorrect answers. The correlation coefficient obtained between them was 0.9833. Upon collecting the difficulty of the questions and student responses, respective grading could be done using the stated formula. Later on, the progressive model was simulated with five different cases (Best case, above-average case, below average case, the average case, worst case). The model outperformed in all the cases with appropriate difficulty levels. Online Tests have ushered a revolution in the assessment of students but yet they tend to be unpopular in India as the evaluation based on pen-paper approach is preferred. The main reasons for this are difficult to grade everyone at the same level, susceptible to cheating, and transition to open books. Using our study, universities can identify obstacles, and prepare an appropriate result-driven plan of action for implementing the mobile-based online test and make easy migration from paper-based test to online test.

Keywords Progressive model · Difficulty · Question · Question paper · Online test · Mobile based test

1 Introduction

The use of mobile has been drastically increased in past few years. Most of the mobile users are from the age of 18 to 29 years old (Crompton and Burke 2018). Mobile devices have spread at an unprecedented rate in the past decade and 95% of the global population live in an area covered by a mobile-cellular network. With the advent of mobile technology, mobile applications have become smarter and ubiquitous in the area of healthcare, education, e-commerce, transportation, Tourism, home, and industrial automation, etc. (Banerjee and Gupta 2015)(Mosa et al. 2012) (Ally and Prieto-Blázquez 2014) (Tan et al. 2017). The development of mobile applications is of two types. The first type is device-neutral mobile application i.e. in this type of application, the purpose is served through a mobile browser. Another type of mobile application is a native application that is coded on an android or iOS platform (Bowen and Pistilli 2012). The study says that – most people rely on smartphones rather than websites for better accessibility of information (Wong 2012).

Mobile learning has been increased in past years and has attracted the interests of academicians and educators in the past many years especially in higher education (Pimmer et al. 2016). Mobile devices are considered cultural tools that are transforming socio-cultural practices and structures in all spheres of life (Pachler et al. 2010). Many higher education institutes are promoting mobile-based education based on its flexibility. It is expected that the next generation of mobile learning will be ubiquitous and learners themselves will be more mobile and able to learn using multiple devices (Krull and Duart 2017). Mobile learning makes learners learn concepts in a very friendly and interactive manner as learners are not rendered immobile by desktop computers and traditional classrooms (Kaliisa and Picard 2017). Another viewpoint to look at mobile learning is "Anytime" and "Anywhere". Mobile learning is becoming a new research field including mobile-based assessment, mobile-based tests, and mobile-based classroom learning. Mobile learning is one of the successful examples of Bring your device (BYOD) concept in higher education.

There have been numerous researches happened in listing the advantage and disadvantages of mobile-learning in the past few years (Fojtik 2015) (Alhassan 2016) (Uther 2019). It is to be noted that mobile learning is not a kind of special learning rather it is part of regular learning which enhances the learning experience better and students need not be dependent on classroom stuff like paper-based notes, writing pads, presentations, and project materials. Everything is available on mobile devices through mobile applications. Some of the advantages of mobile learning include the following:

Mobility and Ubiquity (Sarrab et al. 2016)**:** Students need not use traditional classrooms for attending lectures and accessing resources. The entire contents are available on mobile anytime and anywhere.

Interactivity and accessibility (Torres Diaz et al. 2015) The educational applications are more interactive, the online courses like edx, Coursera are purely assignment and case study based which can be attempted anywhere.

Multi-platform and universality (Sharples et al. 2002) Most of the mobile learning applications are supportive of multiple platforms and universal. Through this, there is always new learning which helps to facilitate the exchange of information and supports self-regulated learning.

There have been some challenges and barriers to mobile-based learning. They are categorized into technological challenges, Personal challenges, or cultural challenges (Filho and Barbosa 2013) (Şad and Göktaş 2014) (Krotov 2015). The details about challenges are drawn in Fig. 1.

Assessment is equally essential along with learning. An assessment helps students to learn. When students can see how they are doing in a class, they can determine whether or not they understand the course material. Assessment sometimes motivates the students if they are properly employed with teaching. In recent years, Mobile Based Assessment (MBA) is playing a role in higher education across the world. In the COVID-19 pandemic, when all the higher education institutions are not operative and the learning model has shifted to online, an attitude towards assessment has changed. MBA is relatively a new model of assessment that is delivered through wireless or mobile devices. Mobile learning and assessment spans from the curriculum- led classroom instruction to informal highly mobile learning on the move (Sharples 2013). Although there is very little literature on MBA few research implementations like (Romero et al. 2009), (Sung et al. 2016) (Lai and Chen 2013), (Liaw and Huang 2015) and (Hwang and Chang 2011) were showed the fact that the perception of students in mobile-based assessment is positive. Although some issues exist with the use of mobile devices in the assessment. The issues such as usability issues compared to computer-based assessment (Huff 2015), Security related issues (Thamadharan and Maarop 2015), Personality and psychological issues (Y.-S. Wang et al. 2009). Such issues may be the refraining the use of mobile devices especially when assessment comes into the picture.

The contribution of our paper is listed below:

- We proposed a unique and novel mobile-based progressive model that automatically calculates the difficulty level of the multiple-choice questions (The initial level of difficulty is given).
- We also aimed that - the proposed algorithms for the progressive model are having adequate computational complexity as they are being used in the mobile-first platform (Oyelere et al. 2018).

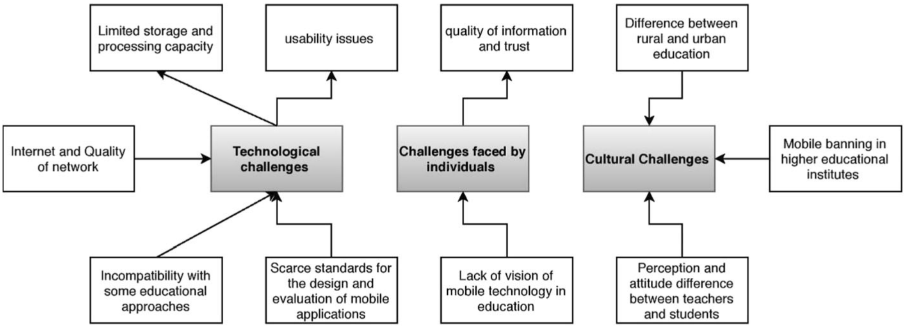

Limited storage and processing capacity
usability issues
quality of information and trust
Difference between rural and urban education
Internet and Quality of network
Technological challenges
Challenges faced by individuals
Cultural Challenges
Mobile banning in higher educational institutes
Incompatibility with some educational approaches
Scarce standards for the design and evaluation of mobile applications
Lack of vision of mobile technology in education
Perception and attitude difference between teachers and students

Fig. 1 Challenges of Mobile-based learning

- We adopted a flexible approach of selecting the difficulty level as per the type of exam conducted for the students.
- Our model applies to all types of adaptive and competitive exams like GRE (Davey and Lee 2011), CAT, and other logical reasoning-based tests.
- Our model is also based on the theoretical model proposed by (Nikou and Economides 2017).

Cerebranium is envisioned as an element with the ability to combine the functions of the cerebrum. We at Cerebranium delicately blend logic, reason, emotion, and sensory delight using cutting edge technology and user empathy to build holistic solutions for a digital future. The model proposed is for student assessment on the mobile learning platform. Using this model, the professors would be able to assess students using questions that have already been used for testing in the past. For establishing the correlation between the difficulty of a question and the students' accuracy a test was conducted for 120 students of computer engineering background and the data obtained was used for the application of suitable correlation calculations. The logic behind the implementation of this model is that a student who answers a difficult question correctly in a multiple-choice questions test should be graded higher for that question than for a question that is of a lower difficulty level. Hence, a student who answers all the questions correctly would be getting maximum possible marks and a student answering all questions wrongly would be getting a zero, but if several students have answered one question wrong then there is a possibility that their marks will differ depending upon the difficulty of the question answered wrongly by them. After the difficulties of all questions have been computed, the paper is generated using an algorithm that will segregate the paper into sets and increase or decrease the difficulty of the forthcoming sets depending upon the answers given in the previous set. Furthermore, the marks of every student are assessed based upon the total difficulty of questions answered correctly by the students using the maximum attainable difficulty and the maximum possible marks for that paper, hence providing maximum accuracy in students' assessment.

The outline of the paper is as follows. Section 2 is about a literature survey. To the best of our knowledge, we haven't found any related research which calculates the difficulty level through our set of steps, we, therefore, tried to cover the literature based upon fully-automated computer-based adaptive tests. Section 3 defines the methodology of the research work. Section 4 defines the parameter selection for the proposed model through correlation analysis. Section 5 is the proposed progressive model with algorithm and computational complexity analysis, Section 6 is the simulated results of the progressive model. Section 7 discusses the results, Section 8 gives a glimpse of the future of our model implemented in the Promexa application (developed by Cerebranium Inc.) finally Section 9, concludes the research work giving future work of the research.

2 Literature survey

The Literature work is not directly related to the model which we have built to estimate the difficulty of the questions.

In (Mishra and Jain 2016), classification is done on the QAs and it is based on specifically defined criteria like what are the different types of application domains (Restricted or Open-domain QAs), variety of questions that are asked (Hypothetical, Factoid or casual questions), type of analysis done on questions and source documents (Morphological, semantic and the like), the data consulted in a data source (structured or unstructured), different characteristics of data (language, size), matching function to generate student answer (probability model, set theory model and so on) and the various type of techniques that are used to retrieve the answers (Data mining, Natural Language processing are a few examples). QAS enables the users to ask natural language questions and get back the appropriate correct answers.

(Quinc and Nac 2011) examines the effect of time on the predicted difficulty of reference questions. The method chosen by them to investigate the research question is content analysis in a modified form. Characteristics of the question have been counted to show the distribution and relationships between the questions over time. The concept of difficulty cannot be measured directly hence it is examined through a rating of the overall questions - a modification of content analysis. The outcome was ambiguous as after analysis, it was not clear whether the difficulty has increased overtime or not. (Perez et al. 2012) investigates the ability of teachers to correctly identify the level of difficulty of a question compared to students' perception. To support them, they have built an automatic classification expert system that examines the difficulty level of questions posed through a competitive e-learning tool called QUESTOURnament. The expert system has been built by using a variety of genetic algorithms and fuzzy logic. This system examines the difficulty from a students' perception.

(Singhal et al. 2017) describes a framework that is used to generate questions that are based on user-defined difficulty levels. The users order the factors which are predefined to measure the difficulty levels. The framework decides which factors will be added or removed to modify the difficulty of the question depending on the order given by the user. This framework generates many questions and tests the concepts in various domains based on the user-defined difficulty level. (Q. Wang et al. 2014) takes a novel question difficulty estimation approach called Regularised Competition Model (RCM) that combines question user comparisons and questions' textual descriptions into the unified framework. They deal with data sparseness problems and cold-start problems by incorporating textual information. Further, they apply an unsupervised Machine Learning algorithm called the K-Nearest neighbor (KNN) approach to predict the difficulty levels of the new questions based on textual similarities.

(Intisar and Watanobe 2018) proposes an intelligent system that is based on fuzzy rules. The fuzzy rules have been derived by cluster analysis. The problems by this expert system are not directly labeled in terms of difficulty. Programming problems vary greatly from one another in terms of categories and skills that are required to solve the problem. After clustering, analysis is done to choose the best clustering algorithm which has the highest score to generate fuzzy rules which then partition the questions into easy, medium, and hard.

(Hensler and Beck 2006) proposes a model that shows that the difficulty and the response time is taken by the students are the most important predictors used when measuring student performance. The model works for multiple-choice cloze (MCC) questions and it extracts information from the question to assess the level of difficulty of the question. This information of difficulty of the question is used to interpret the performance of scores. They trained their model using multinomial logistic regression and found that factors like difficulty, response time by the students, length of the question, syntactic guess rate statistically impact the students' MCC question performance. In (Sullivan 2001), Quantitative classification of questions was done into different difficulty levels. Each measure is encoded into a spatial representation of inter-question similarity. Discriminant analysis is done based on the map to predict question difficulty and an accuracy of 80% was achieved across multiple performance measures.

In (Metzler and Croft 2005), The goal of question classification is to accurately assign labels to questions based on expected answer type. Statistical methods are efficient in handling different features. They analyzed that semantic features are more powerful than syntactic features. They use the SVM algorithm to classify the type of questions. They also conclude that combining both syntactic and semantic features allow more flexibility and generally gives better performance. (Perikos et al. 2016) estimates the difficulty level of different exercises in natural language (NL) to the first order of logic (FOL). The difficulty is based on different parameters which is the FOL formula and it is also based on the semantics of exercise which is the Natural Language sentence. The system will take input parameters like the number, type, and order of quantifiers and the different number of implications. The final difficulty level is based on both the semantic aspects of the NL sentence and the FOL formula structure. (Pado 2018) verifies that Bloom's Taxonomy which is a tool for estimating the difficulty of questions, accurately predicts the question difficulty of short - answer questions. Taxonomy considers the teaching materials to determine the difficulty. They also compute the variation in the answers based on the answer strings.

(Zhang et al. 2017) proposes a model that estimates the difficulty of the question from the question bank. It does so by taking in 4 attributes and 26 features based on Principal Component Analysis. Taking these features into account, machine learning algorithms - tree-based algorithms are used for the classification.

Overall, the related works have been done to predict the difficulty of questions based on Natural Language processing and other Machine Learning classification algorithms.

3 Methodology

3.1 Research design and participants

The dataset needed to be collected for establishing the correlation between the wrong answers and the difficulty. For this purpose, we conducted a quiz that consisted of twenty questions (refer to Table 13 in the appendix section) based on the Python Programming Language on a total of 120 students of Computer Engineering Course of the same institution. The initial difficulty for each question was taken as input from a total of 8 python experts on a scale of 0–5 which was then averaged and converted to the 0–1 difficultly scale. The data that would be used for obtaining correlation would consist of the total number of responses, the number of correct/wrong answers for every question, and the initial difficulty. Table 1 gives the responses of python experts and initial difficulty calculation.

Table 1 Inputs and Calculation of Initial Difficulty (Responses from python experts)

Q no.	Python Expert 1	Python Expert 2	Python Expert 3	Python Expert 4	Python Expert 5	Python Expert 6	Python Expert 7	Python Expert 8	Avg (0–5)	Initial Difficulty (0–1)
1	3	2	2	2	1	2	2	1	1.875	0.375
2	1	2	1	2	1	1	1	1	1.25	0.25
3	2	2	1	1	1	2	1	1	1.375	0.275
4	2	3	4	2	3	4	3	1	2.75	0.55
5	3	2	2	3	3	3	3	3	2.625	0.525
6	5	4	3	4	5	4	3	3	3.875	0.775
7	1	1	2	1	1	1	1	1	1.125	0.225
8	1	1	2	2	2	1	1	2	1.5	0.3
9	3	4	2	2	2	2	2	2	2.375	0.475
10	4	5	3	4	4	4	4	2	3.75	0.75
11	1	1	3	2	3	3	3	2	2.25	0.45
12	2	2	1	1	1	2	1	1	1.375	0.275
13	5	4	3	3	4	5	4	2	3.75	0.75
14	1	1	1	1	1	1	1	1	1	0.2
15	2	3	1	2	3	3	3	1	2.25	0.45
16	2	4	3	4	3	3	3	2	3	0.6
17	3	2	3	3	3	3	3	3	3.25	0.575
18	4	4	3	3	3	3	3	3	3.25	0.65
19	2	3	2	3	4	3	3	3	2.875	0.55
20	2	3	2	3	3	3	2	2	2.5	0.5

3.2 Data collection

On completion of the questionnaire and computing of initial difficulty, the questionnaire was distributed as a quiz on Google forms. The google form link was forwarded through social media platforms such as WhatsApp, Instagram stories, and Institute Portal. During the responses from the students, the non-disclosure of the personal information was assured. The students were also assured that – their responses will not be shared with the institute or any teacher. They have been also assured that; their responses do not have any effect on their academic assessment done by the institute. It is also made sure that each participant can answer the quiz only once so that honest responses would be recorded. This was then shared with the students of the computer department via the institution and they were given a time of two weeks for taking the quiz. The form had 23 questions, out of which the first three were to collect the demographics of the students. Table 2 gives the demographic details about the response collection.

Table 2 Demographic Information

Demographic details	Scale	Number
Name	–	–
Age	18–22	120
Type of Degree/Program	Undergraduate	89
	Post-Graduate	30
	Doctorate	01
Gender	Male	92
	Female	24
	Prefer not to say	04

3.3 Data analysis

Correlation is used in statistics as a technique to determine the degree to which two variables are related. The correlation coefficient is used to determine the relationship between the two variables. It is popularly known as Karl Pearson's Correlation coefficient. It measures the nature and strength of correlation between the two quantitative variables. The sign of the coefficient denotes the nature whereas the value of the coefficient denotes the strength. If the coefficient has a positive sign, it is called the direct relationship. It means that if the value of one variable increases, the other variable's value will increase too and vice versa. If the coefficient has a negative sign, then it is called an indirect or inverse relationship. This means that if the value of one variable increases, the value of another variable will decrease and vice versa. The value of the correlation ranges between −1 to 1 and this denotes the strength of the correlation. If the value is 0 then there is no relation between the variables. If the value is between 0 and 0.25 (both negative and positive side), then there is a weak correlation. If the value is between 0.25–0.5 ((both negative and positive side), then there is an intermediate correlation. If the value is between 0.75–1 (both negative and positive sides), there is a strong correlation between the variables. In this paper, we are checking the correlation between the initial difficulty and the total number of wrong answers, using the following formula.

$$r=\frac{\sum xy-\frac{\sum x\sum y}{n}}{\sqrt{\left(\sum x^2-\frac{\left(\sum x\right)^2}{n}\right)\cdot\left(\sum y^2-\frac{\left(\sum y\right)^2}{n}\right)}}$$

Where,

r is the correlation coefficient.
x is the total no. of wrong answers.
y is the initial difficulty.
n is the sample population.

4 Parameter selection for difficulty calculation

As mentioned above, a test of 20 questions was conducted on 120 students. The initial difficulty for each question was taken as input from a total of 8 python experts on a scale of 0–5 which was then averaged and converted to the 0–1 difficulty scale (as shown in Table 1). All the wrong answers for each question were summed up after the survey and a dataset was established (shown in Table 1) to calculate the correlation between the wrong answers (x) and the initial difficulty (y) using the above-mentioned formula and the total responses (n). The correlation coefficient (r) between them came out to be **0.9833**. This proves that both are highly positively correlated. If the total number of wrong answers for each question is high, it means that the difficulty of that particular question was also high and similarly, if the total number of wrong answers for a question is less, then the difficulty of that question was also less. It concludes that both are highly dependent on each other and the difficulty of a question can be calculated using the number of wrong answers as a factor (see Table 3).

5 Proposed progressive model

The proposed model for the student assessment system comprises three main steps, namely, computing the difficulty, generating the question paper, and evaluating the student (as shown in Fig. 2). Based on the above-established correlation between the difficulty of a question and the wrong responses the difficulty is computed first and the questions are categorized into different difficulty levels. The questions are then used to generate the question based on the algorithm as shown in Section 5.2 and finally the responses are then used to grade the student.

5.1 Computing the difficulty

Our objective is to compute the difficulty of a multiple-choice question that has been attempted by n number of students. In an ideal scenario, with no time, age, gender and other constraints in question, the difficulty (d) of any question can be stated as the ratio of the number of wrong answers (w) and the total number of responses (n).

$$(\text{i.e})\mathit{difficulty}\ (d) = \frac{\text{number of wrong responses } (w)}{\text{total number of responses } (n)}$$

The algorithm has been applied to the collected sample data and the results with the calculated difficulty are shown below in Table 4.

5.2 Generating the question paper

The generation of question papers is the key step in this progressive model. The principle followed here is that a student who has answered more questions and questions of higher difficulty in a set will be getting more difficult questions in the upcoming set and vice versa. We take the total number of questions (Q) and total marks

Table 3 Dataset and wrongly answered participants

Q No.	Questions	Wrong Responses (x)	Total Responses (n)	Initial Difficulty (y)
1	Which of these is NOT a reserved word in Python?	44	120	0.375
2	Which of these is the correct syntax for getting "Hello World" as output?	19	120	0.25
3	Which is the standard GUI library in Python?	27	120	0.275
4	Which of the following command causes the loop to skip the remainder of its body and immediately retest its condition before reiterating?	58	120	0.55
5	Which of the following is the correct syntax for a dictionary	54	120	0.525
6	Which of these is NOT a python Exception?	78	120	0.775
7	How to insert comments in Python?	19	120	0.225
8	Which of these is NOT a legal Variable name?	31	120	0.3
9	What is the output of the following code?	45	120	0.475
10	What is the command used to obtain the current date and time in the variable 'x' using the DateTime library?	76	120	0.75
11	Which of these values of 'x' will be maximum?	48	120	0.45
12	What will be the output of the following code?	13	120	0.275
13	Which of these will be used to randomize the order of the list 'x'?	76	120	0.75
14	What is the correct extension for python files?	8	120	0.2
15	What is the correct syntax to output the type of a variable or object in Python?	44	120	0.45
16	Which method can be used to return a string in upper case letters?	63	120	0.6
17	Choose The correct function to get the ASCII code of a character	57	120	0.575
18	What is the output of print ('%x, %X' % (15, 15))	61	120	0.65
19	Select the correct option to join two lists in Python	56	120	0.55
20	What is the output of the following list assignment?	47	120	0.5
		$\sum x=924$	$n=120$	$\sum y=9.45$

I like input. We keep the total number of questions per set constant at 8 and using the input Q we calculate the total number of sets (N). In a situation where the number of questions is not divisible by 8, the number of sets is taken as the next multiple of 8 with the last set having fewer questions than the others.

$$\text{(i.e)Number of sets } (N) = \frac{Total\ number\ of\ Questions\ (Q)}{8}$$

if $\frac{Q}{8}! = 0, Q =$ next multiple of 8,

The difficulty (range 0–1) will be categorized into 8 levels as follows (Table 5):

The progressive model will monitor the questions answered by the candidate in every set and decide the forthcoming sets accordingly i.e. for correct answers the

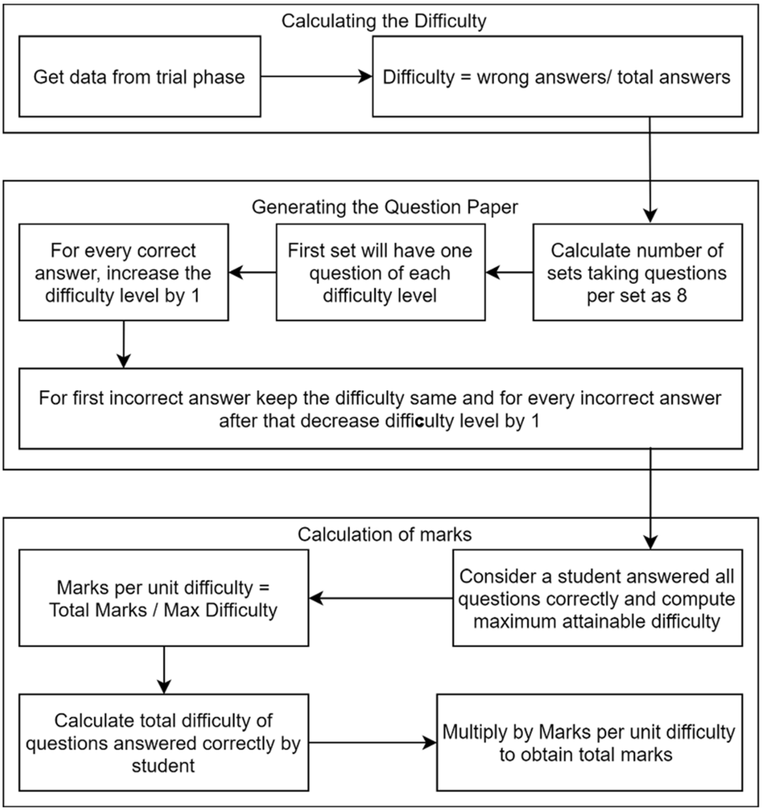

Calculating the Difficulty
Get data from trial phase
Difficulty = wrong answers/ total answers
Generating the Question Paper
For every correct answer, increase the difficulty level by 1
First set will have one question of each difficulty level
Calculate number of sets taking questions per set as 8
For first incorrect answer keep the difficulty same and for every incorrect answer after that decrease difficulty level by 1
Calculation of marks
Marks per unit difficulty = Total Marks / Max Difficulty
Consider a student answered all questions correctly and compute maximum attainable difficulty
Calculate total difficulty of questions answered correctly by student
Multiply by Marks per unit difficulty to obtain total marks

Fig. 2 Proposed Progressive Model- Block Diagram

difficulty of the next set increases and for wrong answers, it decreases. The process is as follows (Figs. 3 and 4),

- If answered incorrectly 1 time in the set, Difficulty remains the same
- If answered incorrectly 2nd time in the same set of questions, Difficulty—.

Following is the Algorithm of the entire Question paper generation process:

5.3 Evaluating the student

In a progressive model, for accurate evaluation the marks will be allotted based on the difficulty of the questions attempted correctly by the student. As students will be awarded marks based on the total difficulty answered correctly and not the total number of questions attempted correctly, answering the same number of questions might yield different marks. First, we calculate the maximum attainable difficulty by assuming that a student has attempted all questions correctly and should obtain maximum marks. This is done by adding the difficulty of all the questions attempted and getting the sum that will be the maximum attainable difficulty (D_{max}). This value would be used for the

Table 4 Sample Data with Calculated Difficulty

Sr no.	Questions	Wrong Responses (w)	Total Responses (n)	Difficulty Calculated (d)
1	Which of these is NOT a reserved word in Python?	44	120	0.3666666667
2	Which of these is the correct syntax for getting "Hello World" as output?	19	120	0.1583333333
3	Which is the standard GUI library in Python?	27	120	0.225
4	Which of the following command causes the loop to skip the remainder of its body and immediately retest its condition before reiterating?	58	120	0.4833333333
5	Which of the following is the correct syntax for a dictionary	54	120	0.45
6	Which of these is NOT a python Exception?	78	120	0.65
7	How to insert comments in Python?	19	120	0.1583333333
8	Which of these is NOT a legal Variable name?	31	120	0.2583333333
9	What is the output of the following code?	45	120	0.375
10	What is the command used to obtain the current date and time in the variable 'x' using the DateTime library?	76	120	0.6333333333
11	Which of these values of 'x' will be maximum?	48	120	0.4
12	What will be the output of the following code?	13	120	0.1083333333
13	Which of these will be used to randomize the order of the list 'x'?	76	120	0.6333333333
14	What is the correct extension for python files?	8	120	0.06666666667
15	What is the correct syntax to output the type of a variable or object in Python?	44	120	0.3666666667
16	Which method can be used to return a string in upper case letters?	63	120	0.525
17	Choose The correct function to get the ASCII code of a character	57	120	0.475
18	What is the output of print('%x, %X' % (15, 15))	61	120	0.5083333333
19	Select the correct option to join two lists in Python	56	120	0.4666666667
20	What is the output of the following list assignment?	47	120	0.3916666667

Table 5 Difficulty range category of the proposed model

Difficulty Levels	Difficulty range
Level 1	0–0.125
Level 2	0.126–0.25
Level 3	0.256–0.35
Level 4	0.356–0.5
Level 5	0.51–0.65
Level 6	0.656–0.75
Level 7	0.756–0.85
Level 8	0.856–1

```
Algorithm 1: Function Check-for-Set
  Input: X
  Data: Global W, Global LQ
  // W = List of Wrongs remaining
  // LQ = Level Presently Question on
  for $i \leftarrow 0$ to $7$ do
      res ← response of X[i];
      // res either has response has Right or Wrong
      if $res ==$ "Right" then
          LQ[i] ← LQ[i] + 1;
          W[i] ← 2;
      else if $res ==$ "Wrong" then
          if $W[i] == 1$ then
              LQ[i] ← LQ[i] - 1;
              W[i] ← 2;
          else
              W[i] ← W[i] - 1;
          end
      end
  end
```

Fig. 3 Algorithm for Check for set in the progressive model

evaluation of every student who enrolled in the examination. Now, we calculate the total difficulty of the questions attempted by the student correctly and label it as *D*. For obtaining the evaluating quotient (*q*) of the student we get the ratio of the above values,

$$(\text{i.e})q = \frac{D}{D_{max}}$$

To convert the above ratio into a value corresponding to the marking scheme we calculate the multiplying factor (*x*) using the maximum marks (M_{ax}) and maximum attainable difficulty (D_{max}).

$$\text{the formula}, x = \frac{M_{max}}{D_{max.}} \qquad \text{(UsingUsing)}$$

```
Algorithm 2: Progressive Difficulty Examinations
  Input: Q
  Data: Global W, Global LQ
  // Q = Number of Questions
  // W = List of Wrongs remaining
  // LQ = Level Presently Question on
  if $Q\%8 \neq 0$ then
      Q ← Next multiple of 8;
  end
  N ← Q / 8;
  // N = Total Number of sets
  W ← [2, 2, 2, 2, 2, 2, 2, 2];
  LQ ← [1, 2, 3, 4, 5, 6, 7, 8];
  for $i \leftarrow 1$ to $N$ do
      Construct Question set with Levels in LQ;
      responses ← Take responses;
      Call Check-for-Set(responses);
      End of set;
  end
```

Fig. 4 Algorithm for Question paper generation process

Now, to obtain the marks obtained by the student (M) concerning the maximum marks we multiply the evaluating quotient (q) by the multiplying factor (x).

$$(\text{i.e})M = q \times x$$

In this way, we can calculate the total marks obtained by any student using the difficulty of the questions answered by them correctly.

5.4 Computational complexity of progressive model

5.4.1 Time complexity

The most common method of calculating time complexity is by counting each of the elemental steps performed by any given algorithm, to complete its execution. For example, consider a single operation for the addition of two numbers, the time taken for its execution is assumed to a constant, C, which can be considered as taking 1 unit of time,

$$c = a + b$$

Taking another example of a *for loop* executing from 1 to n. The main loop will be executed $n+1$ times, and each of the statements inside the loop *"n"* times. The *for loop* can be depicted as,

```
for (i=0; i<N; i++)                                   --------- n+1

{

        Statement 1;                                  --------- 1*n

        Statement 2;                                  --------- 1*n

}
```

In the above *for loop* block, as the loop executes from 0 to n, both the statements inside the loop will execute exactly n times. Whereas the main block (*for* statement) itself executes $n + 1$ times as the condition is checked even when $\text{i} = n$.

For this For Loop, the time complexity comes out to be 3n + 1 which can be written as O(n).

5.4.2 Space complexity

Space complexity refers to the total memory space that a program or an algorithm occupies during its execution. It includes the memory space occupied by the input variables as well as any auxiliary or temporary space created at the time of execution.

Space complexity for any algorithm is calculated by checking the maximum amount of memory that would be occupied by all the variables used in the algorithm. For example, a single integer value would have space complexity as a constant C, which can be considered as 1, or an array of size *"n" will* require *n* blocks of memory and hence its complexity would be O(n).

The space complexities for the two algorithms mentions in Section 5.2. can be calculated as:

Algorithm 1: Function Check-For-Set

Considering the input as N questions in one set and calculating the space complexity line-by-line,

Line 1 – Requires n+1 time

Line 2 to 13 – Requires n time

So, the total time taken for this algorithm will be 2N+1, which can be considered as O(n) and the space complexity is O(n).

Algorithm 2:Progressive Difficulty Examination

Again, considering n questions in one set and calculating the space complexity line-by-line,

Lines 1 to 6 – Requires 1 time each,

Line 7 – Requires N+1 time

Lines 8 to 11 – Requires N time each

*Line 10 – Will require N*n time as the Check-For-Set function has a time complexity of O(n).*

*So, the total time taken for this algorithm is equal to O(N*n). and the space complexity is O(n).*

6 Results and simulation

This section gives the simulated results of the proposed progressive model. The example shown in this section consists of 40 different questions of initial difficult level which is spread across 5 different sets. The difficulty level of each question is calculated based on 5 different cases. The details of each case are given in Table 6.

As per the formula of generating question paper in Section 5.2., the total number of sets is 40/8 = 5.

Table 7 gives the different difficulty level and range which corresponds to each difficulty level.

The rationale behind choosing 8 levels of difficulty is restricted to the current simulation, in other words, there can be variations in choosing the difficulty levels as per the number of questions and students. In the current simulation, we took 40 questions with 5 sets. The lower and upper limit of the difficulty range is being kept from 0 to 1 which can also be changed. The selection of appropriate levels of difficulty and corresponding ranges will be done by the application developer who incorporated

Table 6 Different cases of result simulation

Case	Description	Results simulation table
Best Case	In this case, it is assumed that all the questions are correct so that the difficulty of the maximum questions is attained by the student.	Table 8
Below Average Case	In this case, it is assumed that most of the questions are wrong, and very few questions are marked correct by the student.	Table 9
Average Case	In this case, 50% of questions are wrong and 50% of questions are marked correctly.	Table 10
Above Average Case	In this case, it is assumed that most of the questions are correct and very few questions are marked incorrect by the student.	Table 11
Worse Case	In this case, it is assumed that all the questions are marked wrong.	Table 12

this model or instructor who wants to conduct the exam using this model or examination officials who wish to conduct examinations for university or academic institution using said, progressive model. To the best of our knowledge, it is desirable to choose difficulty levels with a maximum integer value. The more granularity level of the difficulty level of the model gives better calibration of a generation of questions. With 8 difficulty levels, the model is simulated with different cases in further sections.

6.1 Best case simulation of progressive model

As all the questions are answered correctly in this case, this student (Table 8) can attain the maximum possible difficulty that any student can achieve, which turned out to be 27.728 for this set of simulations. Hence, the marks obtained by the student in the best-case simulation were 100% of the maximum marks. We were also able to observe that when a student answers a question correctly the corresponding question in the forthcoming set would be a comparatively difficult one but the student would yield more marks for the same.

Table 7 Different difficulty level and range

Difficulty Level	Difficulty Range	Questions
QD1	0.000–0.125	20
QD2	0.126–0.250	20
QD3	0.251–0.375	20
QD4	0.376–0.500	20
QD5	0.501–0.625	20
QD6	0.626–0.750	20
QD7	0.751–0.875	20
QD8	0.876–1.000	20

Table 8 Best Case simulation of Progressive Model

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
SET 1	QD1	QD2	QD3	QD4	QD5	QD6	QD7	QD8
T/F	1	1	1	1	1	1	1	1
Diff	0.12	0.191	0.276	0.456	0.61	0.655	0.799	0.989
SET 2	QD2	QD3	QD4	QD5	QD6	QD7	QD8	QD8
T/F	1	1	1	1	1	1	1	1
Diff	0.177	0.288	0.499	0.611	0.645	0.786	0.899	0.945
SET 3	QD3	QD4	QD5	QD6	QD7	QD8	QD8	QD8
T/F	1	1	1	1	1	1	1	1
Diff	0.299	0.514	0.61	0.698	0.856	0.945	0.895	0.978
SET 4	QD4	QD5	QD6	QD7	QD8	QD8	QD8	QD8
T/F	1	1	1	1	1	1	1	1
Diff	0.398	0.614	0.711	0.812	0.957	0.888	0.912	0.942
SET 5	QD5	QD6	QD7	QD8	QD8	QD8	QD8	QD8
T/F	1	1	1	1	1	1	1	1
Diff	0.598	0.687	0.785	0.912	0.935	0.945	0.894	0.997

6.2 Below average simulation of progressive model

Very few questions are answered correctly in this case, this student (Table 9) was able to attain a cumulative difficulty of 6.593. The marks obtained by the student in the below-average case simulation were 23.77% of maximum marks. Here, we were able to observe that when a student gets several wrong answers in a particular set, they encounter easier questions in the forthcoming sets increasing their chances of getting them right but yielding lesser marks.

6.3 Average case simulation progressive model

Several questions are answered correctly in this case, this student (Table 10) was able to attain a cumulative difficulty of 13.687. The marks obtained by the student in the average case simulation were 49.36% of maximum marks. As the rate of increase of difficulty is more than the rate of decrease of difficulty the student ended up with more difficult questions in the final set when compared to the other sets.

6.4 Above average simulation of progressive model

Most questions are answered correctly in this case, this student (Table 11) was able to attain a cumulative difficulty of 19.185. The marks obtained by the student in the above-average case simulation were 69.19% of maximum marks. When a student attempts a particular question wrongly in two corresponding sets, not only do they lose the marks allotted for those two questions but this also means that the next corresponding question will have a lesser weightage.

Table 9 Below average simulation of Progressive Model

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
SET 1	QD1	QD2	QD3	QD4	QD5	QD6	QD7	QD8
T/F	1	0	1	0	0	0	0	0
Diff	0.121	0.186	0.354	0.394	0.555	0.648	0.779	0.901
SET 2	QD2	QD2	QD4	QD4	QD5	QD6	QD7	QD8
T/F	0	1	0	1	1	0	0	0
Diff	0.215	0.236	0.446	0.479	0.623	0.696	0.845	0.915
SET 3	QD2	QD3	QD4	QD5	QD6	QD5	QD6	QD7
T/F	1	0	0	0	0	1	0	0
Diff	0.247	0.268	0.386	0.598	0.715	0.621	0.659	0.845
SET 4	QD3	QD3	QD3	QD5	QD6	QD6	QD5	QD6
T/F	1	0	1	0	1	0	0	1
Diff	0.367	0.267	0.37	0.567	0.739	0.687	0.566	0.724
SET 5	QD4	QD2	QD4	QD4	QD7	QD6	QD4	QD7
T/F	1	1	0	1	0	0	1	0
Diff	0.499	0.248	0.445	0.476	0.778	0.689	0.489	0.87

6.5 Worse case simulation of progressive model

None of the questions are answered correctly in this case, this student (Table 12) was able to attain a cumulative difficulty of 0. The marks obtained by the student in the worst-case simulation were 0% of the maximum marks. When a student attempts a

Table 10 Average case simulation of Progressive Model

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
SET 1	QD1	QD2	QD3	QD4	QD5	QD6	QD7	QD8
T/F	1	1	1	1	0	0	0	1
Diff	0.11	0.241	0.329	0.485	0.584	0.663	0.778	0.993
SET 2	QD2	QD3	QD4	QD5	QD5	QD6	QD6	QD8
T/F	1	1	1	0	0	0	1	0
Diff	0.244	0.367	0.479	0.598	0.584	0.667	0.869	0.924
SET 3	QD3	QD4	QD5	QD5	QD4	QD5	QD8	QD6
T/F	1	0	1	0	0	1	1	0
Diff	0.349	0.485	0.611	0.587	0.547	0.622	0.992	0.637
SET 4	QD4	QD4	QD6	QD4	QD3	QD6	QD8	QD6
T/F	1	0	0	0	1	0	1	1
Diff	0.493	0.4	0.641	0.448	0.371	0.645	0.948	0.739
SET 5	QD5	QD3	QD6	QD3	QD4	QD6	QD8	QD6
T/F	1	1	1	1	1	0	1	1
Diff	0.619	0.368	0.744	0.372	0.488	0.661	0.987	0.867

Table 11 Above average simulation of Progressive Model

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
SET 1	QD1	QD2	QD3	QD4	QD5	QD6	QD7	QD8
T/F	1	1	0	1	1	0	1	0
Diff	0.124	0.236	0.321	0.459	0.614	0.647	0.855	0.881
SET 2	QD2	QD3	QD3	QD5	QD6	QD6	QD8	QD8
T/F	1	1	0	0	1	1	0	1
Diff	0.211	0.366	0.314	0.596	0.698	0.742	0.893	0.991
SET 3	QD3	QD4	QD2	QD5	QD7	QD7	QD8	QD8
T/F	1	1	1	1	0	1	0	1
Diff	0.371	0.487	0.226	0.622	0.763	0.849	0.872	0.972
SET 4	QD4	QD5	QD3	QD6	QD7	QD8	QD7	QD8
T/F	1	1	1	1	0	1	1	1
Diff	0.482	0.624	0.367	0.697	0.766	0.999	0.866	0.971
SET 5	QD5	QD6	QD4	QD7	QD6	QD8	QD8	QD8
T/F	1	1	1	1	1	0	1	1
Diff	0.6	0.713	0.481	0.863	0.742	0.885	0.963	0.994

particular question wrongly in two corresponding sets if they answer the corresponding questions wrongly then the difficulty level of the questions is reduced after every wrong answer from that point onwards.

To summarise the simulated results and for better visualization, the bar graph of different cases are shown in Fig. 5.

7 Discussions

Mobile learning has grown extensively over the past decade. Moreover, due to the pandemic all the schools, colleges, institutions, corporates, all were shifted to an online medium to keep the workflow ongoing. Mobile assessments are the need of the hour. Whether it is the school/college examinations or competitive examinations or Placement tests, everyone needs an adaptive progressive model (Cella et al. 2007).

The accuracy of the data that was collected for the calculation of the difficulty of the questions increases with the increase in the size of the sample. This will however not be completely error-free as the responses received will always depend on the skillset and learning that the students have received. Hence, for validation of the data that we received, we used the help of inputs from experts in the topic which in this case was Python programming. The past literature includes the classification of the questions into different difficulty levels. It is about the different kinds of algorithms that are used to classify the questions as per their difficulty level. Past research has also mentioned the calculation of difficulty based on the number of right and wrong answers to a particular question along with the response time taken into consideration (Table 13).

The major problem that the researchers face while calculating difficulty is that it is subjective. Everyone has different opinions on the level of difficulty for a particular

Table 12 Worst case simulation of Progressive Model

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
SET 1	QD1	QD2	QD3	QD4	QD5	QD6	QD7	QD8
T/F	0	0	0	0	0	0	0	0
Diff	0.087	0.145	0.278	0.399	0.589	0.745	0.845	0.945
SET 2	QD1	QD2	QD3	QD4	QD5	QD6	QD7	QD8
T/F	0	0	0	0	0	0	0	0
Diff	0.099	0.223	0.345	0.412	0.612	0.712	0.812	0.912
SET 3	QD1	QD1	QD2	QD3	QD4	QD5	QD6	QD7
T/F	0	0	0	0	0	0	0	0
Diff	0.092	0.089	0.189	0.259	0.387	0.533	0.627	0.856
SET 4	QD1	QD1	QD1	QD2	QD3	QD4	QD5	QD6
T/F	0	0	0	0	0	0	0	0
Diff	0.11	0.109	0.032	0.191	0.262	0.39	0.51	0.635
SET 5	QD1	QD1	QD1	QD1	QD2	QD3	QD4	QD5
T/F	0	0	0	0	0	0	0	0
Diff	0.121	0.045	0.057	0.098	0.193	0.265	0.394	0.555

question (Pérez et al. 2012) (van de Watering and van der Rijt 2006). For example, the teacher might find a question comparatively easier, but it might not be the same for the students. Or a question of syntax code might seem easier to a person who knows coding than to a person who is unaware of it. Therefore, a progressive model was apt for this kind of situation. It takes the initial difficulty level of a particular question and depending upon how quickly and correctly the test taker answers it, it progressively

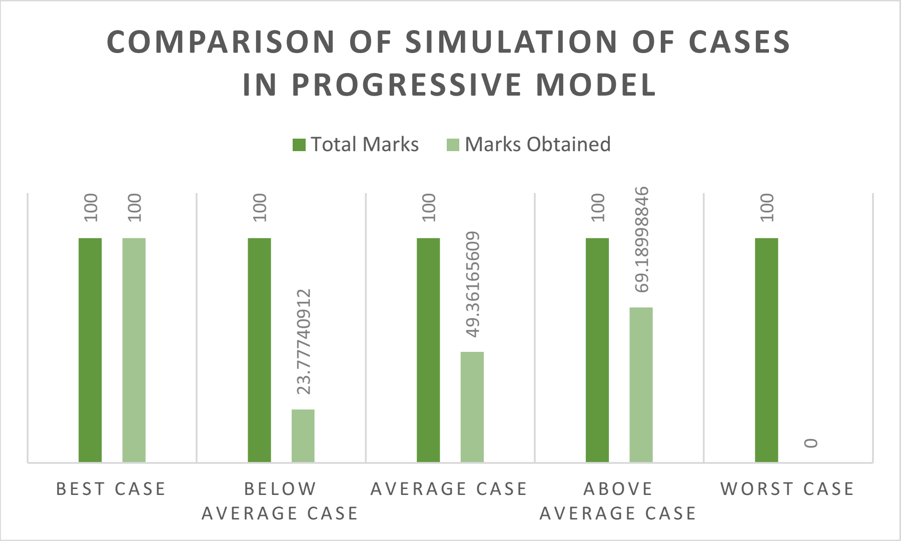


Fig. 5 Comparison of simulation of cases in the progressive model

increases or decreases the difficulty level and accordingly calculates the result. The time complexity of our model is O(n) hence it is linear.

Our progressive model and overall research work are closely related to the Theoretical model presented by (Nikou and Economides 2017). The proposed model, Mobile-Based Assessment Acceptance Model (MBAAM) is based on the Technology Acceptance Model (TAM). MBAAM examined the factors such as Personal Innovativeness, Mobile-Self-Efficacy, Perceived Trust, Content, Cognitive Feedback, User Interface, and Perceived Ubiquity Value has a great impact on the Behavioural Intention to Use Mobile bases assessment and learning. Due to the automatic calculation of difficulty and consistent feedback about students' responses to a particular question, This makes our model based on the framework of MBAAM.

While interpreting the results, we found out that, as expected the student who answered all questions correctly irrespective of the difficulty of the question scored 100% marks (best case scenario). Similarly, the student who answered all questions wrongly got 0 marks (worst case scenario). Whereas, the below-average case turned out to be around 24 marks which are extremely close to 25 marks, which is what we expected. The average case too ended up at 49 marks which are close to the expected 50% mark. The only case that was a little below the expected value of 75 marks was the above-average case which came to be 69 marks. 8 levels of difficulty have only been taken for the sole purpose of simulation. It is not fixed for the model. Any institution/workplace using the model can take its difficulty level and set its bars. The model has been made flexible to take the appropriate difficulty level as best suited by the institution.

This model can be best suited wherever adaptive tests are to be taken. It can be used in schools and colleges. It can be used in institutions or for placement aptitude tests. It can be used for competitive examinations like GRE, GMAT, and so on. Since it takes the initial level of difficulty into account and then gives questions based on the test taker response, this model is best suited for examinations where relative grading is required.

Because of the COVID-19 pandemic, mobile learning and assessments have taken a huge hit. Therefore, there comes a need to build an accurate progressive model for the test takers so that the level is not compromised (Tony et al. 2020). Hence our model fits best because it assesses the students based on their skill level and the questions, they come across adapts to the same. The students with lower skill levels or understanding of the subject encounter easier questions as they make progress in their test but receive fewer marks for them too, making it the ideal unbiased model for student assessment.

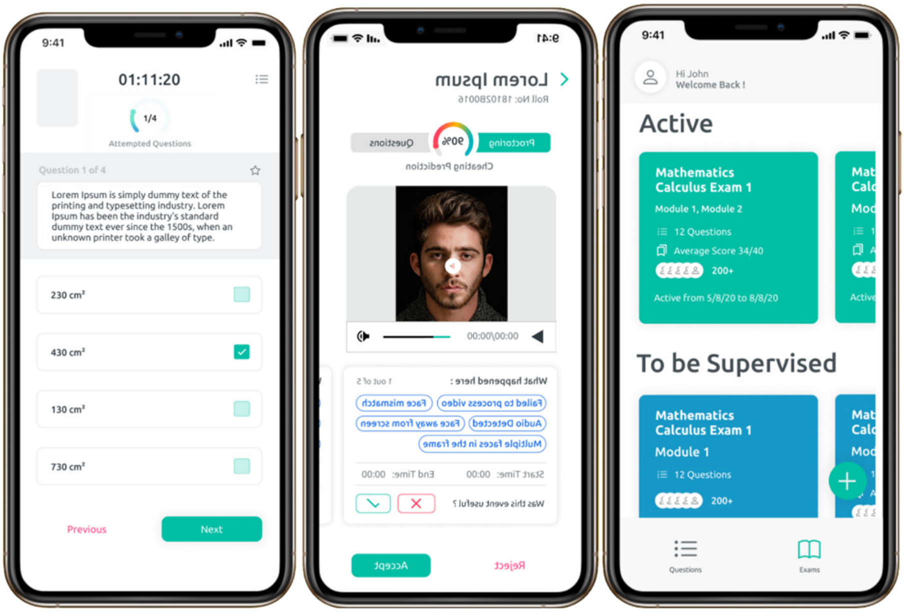
The model was designed in such a way that unlike traditional methods, the grades obtained by the students not only depend on the number of questions answered correctly but also the difficulty of those questions. Hence, the results obtained for the three average cases differ a lot in terms of the marks obtained despite the difference in the number of questions that have been answered correctly in the three cases being quite less.

8 Future of progressive model

This industry-led research work is an outcome of the project – *Promexa* which is built for exam integrity protection as a mobile-first, fully decentralized exam conduction platform which is end-to-end data-driven exam management, multi-factor, and AI-assisted cheating detection technology. Promexa is made by Cerebranium (see https://cerebranium.com/promexa/) and it is being proactively used in various universities as an AI-assisted proctoring system to remotely conduct exams. The mobile application uses this progressive model for generating and calculating the difficulty of the question paper. Promexa has various features apart from progressive model and automation of difficulty calculation of questions. A few of the features have been listed below:

- Integrity: The application has detailed log record and video feeds which helps universities to assure exam integrity.
- Offline exam conduction: zero network connectivity is requiring which can ensure seamless conduction of exams.
- Battery usage in mobile: The application requires a low battery for smooth conduction of exams than any other AI powered proctoring systems.
- Question bank management and dashboard assistance: seamless subject management and with modular question bank and simplified dashboard for accessing data for students, faculty, and admin of the application.

Figure 6 shows the prototype of the Promexa application with basic features. The beta version of the application is used by various educational institutions across the world to conduct online exams.



Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type.

Fig. 6 Prototype of Promexa application

9 Conclusion

With the increasing digitalization and the growth in the field of mobile learning, the proposed model enables the professors to assess student understanding and conduct tests for them on the mobile platform. A thorough literature survey on the existing models was conducted before conducting any test and developing the model. The existing problems in this field like the inaccuracy of grading and uneven judgment in randomization of questions have been resolved to a great extent using suitable algorithms. We conducted a training data test using 20 python questions and established the correlation between the difficulty of any question and the number of wrong answers for that particular question. Then the model that consists of three steps, calculating the difficulty, generating the question paper, and evaluating the student, was developed to resolve the issues that currently exist in the various mobile test applications. With the help of the given structure, students will be able to attempt questions that are suitable to their understanding of the subject and be graded accordingly.

The model was designed in such a way that unlike traditional methods, the grades obtained by the students not only depend on the number of questions answered correctly but also the difficulty of those questions. Hence, the results obtained for the three average cases differ a lot in terms of the marks obtained despite the difference in the number of questions that have been answered correctly in the three cases being quite less

Even after this model has been implemented there would be scope for improvement in various parts before it can be considered as the ideal way to assess and replace the traditional methods. The future development ideas for the model include:

i. Time is taken for attempting a question should be taken as a factor that affects difficulty, especially when it comes to different types of questions apart from the currently used multiple-choice questions.

ii. Even though the test can be taken offline, students will need an internet connection initially to download the test in that case.

iii. Before any question can be used for actual assessment it needs to be tested for a significant number of students so that the difficulty of the question can be calibrated, hence, making it difficult for a newly developed question to be used for evaluation directly.

iv. The inclusion of machine intelligence in the calibration of questions in our progressive model will be our next move in research work.

Appendix

Table 13 Sample Questionnaire used for Correlation analysis

Sr no.	Question	Choices	Correct Choice

1	Which of these is NOT a reserve word in Python?	elif
		import
		lambda
			extract

2	Which of these is the correct syntax for getting "Hello World" as output?	printf("Hello World")
		print(Hello World);
		print("Hello World");
		print("Hello World")
			print("Hello World");

3	Which is the standard GUI Library in Python?	Tkinter ToolKit TensorFlow UserInterface	Tkinter
4	Which of the following command causes the loop to skip the remainder of its body and immediately retest its condition prior to reiterating?	break continue pass jump	continue

5	Which of the following is the correct syntax for a dictionary?	dict={"Name"="Ram", "Age"=7, "Class"="First"}
		dict=["Name": "Ram", "Age": 7, "Class": "First"]
		dict={Name: "Ram", Age: "7", Class: "First"}
		dict={"Name": "Ram", "Age": 7, "Class": "First"}
			dict={"Name": "Ram", "Age": 7, "Class": "First"}

6	Which of these is NOT a python Exception?	RuntimeError EnvironmentError ZeroDivisionError BoundLocalError	BoundLocalError
7	How to insert comments in Python?	//This is a comment #This is a comment /* This is a comment */ None of these	#This is a comment

Table 13 (continued)

Sr no.	Question	Choices	Correct Choice
8	Which of these is NOT a legal Variable name?	MyVar My_Var My-Var _My_Var	My-Var

9	What is the output of the following code?
	txt="Hello World"[::-1]
	print(txt)
		Hello World
		ello World
		Hello Worl
		dlroW olleH
			dlroW olleH

10	What is the command used to obtain the current date and time in the variable 'x' using datetime library?	x=datetime.now()
		x=datetime.datetime.now()
		x=datetime.fetch()
		None of these
			x=datetime.datetime.now()

11	Which of these values of 'x' will be maximum?	x=3.5
		x=int("3")
		x=int(3.9)
		All will be Equal
			x=3.5

12	What will be the output of the following code?
	price=49
	txt="The price is {:.2f} dollars"
	print(txt.format(price))
		The price is 49 dollars
		The price is 49.0 dollars
		The price is 49.00 dollars
		Invalid Syntax
			The price is 49.00 dollars

13	Which of these will be used to randomize the order of the list 'x'?	random(x)
		shuffle(x)
		randomize(x)
		choice(x)
			shuffle(x)

14	What is the correct extension for python files?	.py .pyth .pyt .pt	.py

Table 13 (continued)

Sr no.	Question	Choices	Correct Choice
15	What is the correct syntax to output the type of a variable or object in Python?	print(typeof(x))?print(type(x))?print(typeofx)?print(typeOf(x))	print(type(x))

16	Which method can be used to return a string in upper case letters	upper()
		upperCase()
		toUpperCase()
		toUpper()
			upper()

17	Choose The correct function to get the ASCII code of a character	char(‘char’)
		ord(‘char’)
		ascii(‘char’)
		toChar(‘char’)
			ord(‘char’)

18	What is the output of print(‘%x, %X’ % (15, 15))	15 15 F F f f f F	f F

19	Select the correct option to join two lists in Python
	listOne=[‘a’,‘b’, ‘c’,‘d’]
	listTwo=[‘e’,‘f’,‘g’]
		newList=extend(listOne, listTwo)
		newList=listOne.extend(listTwo)
		newList.extend(listOne, listTwo)
		All of the Above
			newList= listOne.extend(listTwo)

20	What is the output of the following list assignment?
	aList=[4,8,12,16]
	aList[1:4]=[20,24,28]
	print(aList)
		[4, 20, 24, 28, 8, 12, 16]
		[4, 20, 24, 28]
		[20, 24, 28]
		[4, 8, 12, 28]
			[4, 20, 24, 28]

Acknowledgements Author would like to say thank you for the anonymous reviewers and respected editors for taking valuable time to go through the manuscript.

Declaration

Conflict of interest The authors of this research study declare that there is NO conflict of interest.

References

Alhassan, R. (2016). Mobile learning as a method of ubiquitous Learning: Students' Attitudes, Readiness, and Possible Barriers to Implementation in Higher Education. *Journal of Education and Learning*. https://doi.org/10.5539/jel.v5n1p176.

Ally, M., & Prieto-Blázquez, J. (2014). What is the future of mobile learning in education? International Journal of Educational Technology in Higher Education. https://doi.org/10.7238/rusc.v11i1.2033, 11.

Banerjee, A., & Gupta, S. K. S. (2015). Analysis of smart Mobile applications for healthcare under dynamic Context changes. *IEEE Transactions on Mobile Computing., 14*, 904–919. https://doi.org/10.1109/TMC.2014.2334606.

Bowen, K., & Pistilli, M. D. (2012). Student preferences for mobile app usage. Research Bulletin (Louisville, CO: EDUCAUSE Center for Applied Research, forthcoming). Available from http://www.educause.edu/ecar. Accessed 3 Dec 2020.

Cella, D., Gershon, R., Lai, J.-S., & Choi, S. (2007). The future of outcomes measurement: Item banking, tailored short-forms, and computerized adaptive assessment. *Quality of Life Research, 16*(1), 133–141. https://doi.org/10.1007/s11136-007-9204-6.

Crompton, H., & Burke, D. (2018). The use of mobile learning in higher education: A systematic review. *Computers and Education., 123*, 53–64. https://doi.org/10.1016/j.compedu.2018.04.007.

Davey, T., & Lee, Y.-H. (2011). Potential Impact of Context Effects On The Scoring And Equating Of The Multistage Gre® Revised General Test. *ETS Research Report Series, 2011*(2), i–44. https://doi.org/10.1002/j.2333-8504.2011.tb02262.x.

Filho, N, F, D., & Barbosa, E, F. (2013). A contribution to the quality evaluation of mobile learning environments. Proceedings - Frontiers in Education Conference, FIE. https://doi.org/10.1109/FIE.2013.6684851.

Fojtik, R. (2015). Ebooks and Mobile devices in education. *Procedia - Social and Behavioral Sciences., 182*, 742–745. https://doi.org/10.1016/j.sbspro.2015.04.824.

Hensler, B, S., & Beck, J. (2006). Better student assessing by finding difficulty factors in a fully automated comprehension measure. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/11774303_3.

Huff, K. C. (2015). The comparison of mobile devices to computers for web-based assessments. *Computers in Human Behavior, 49*, 208–212. https://doi.org/10.1016/j.chb.2015.03.008.

Hwang, G.-J., & Chang, H.-F. (2011). A formative assessment-based mobile learning approach to improving the learning attitudes and achievements of students. *Computers & Education, 56*(4), 1023–1031. https://doi.org/10.1016/j.compedu.2010.12.002.

Intisar, C, M., & Watanobe, Y. (2018). Cluster analysis to estimate the difficulty of programming problems. ACM International Conference Proceeding Series. https://doi.org/10.1145/3274856.3274862.

Kaliisa, R., & Picard, M. (2017). A systematic review on mobile learning in higher education: The African perspective. *Turkish Online Journal of Educational Technology (TOJET), 16*(1), 1–18.

Krotov, V. (2015). Critical success factors in m -learning: A socio-technical perspective. *Communications of the Association for Information Systems, 36*, 105–126. https://doi.org/10.17705/1cais.03606.

Krull, G., & Duart, J, M. (2017). Research trends in mobile learning in higher education: A systematic review of articles (2011 - 2015). *International Review of Research in Open and Distance Learning*. https://doi.org/10.19173/irrodl.v18i7.2893.

Lai, A., & Chen, C. (2013). Mobile-based peer assessment APP and elementary students' perception: Project works of computer curriculum as an example. 2013 IEEE 13th international conference on advanced learning technologies, 489–490. https://doi.org/10.1109/ICALT.2013.156.

Liaw, S. S., & Huang, H. M. (2015). How factors of personal attitudes and learning environments affect gender difference toward mobile distance learning acceptance. *International Review of Research in Open and Distance Learning, 16*(4), 104–132. https://doi.org/10.19173/irrodl.v16i4.2355.

Metzler, D., & Croft, W. B. (2005). Analysis of statistical question classification for fact-based questions. *Information Retrieval., 8*, 481–504. https://doi.org/10.1007/s10791-005-6995-3.

Mishra, A., & Jain, S. K. (2016). A survey on question answering systems with classification. *Journal of King Saud University - Computer and Information Sciences, 28*, 345–361. https://doi.org/10.1016/j.jksuci.2014.10.007.

Mosa, A, S, M., Yoo, I., & Sheets, L. (2012). A systematic review of healthcare applications for smartphones. In *BMC Medical Informatics and Decision Making*. https://doi.org/10.1186/1472-6947-12-67, 12.

Nikou, S. A., & Economides, A. A. (2017). Mobile-based assessment: Investigating the factors that influence behavioral intention to use. *Computers & Education, 109*, 56–73. https://doi.org/10.1016/j.compedu.2017.02.005.

Oyelere, S. S., Suhonen, J., Wajiga, G. M., & Sutinen, E. (2018). Design, development, and evaluation of a mobile learning application for computing education. *Education and Information Technologies, 23*(1), 467–495. https://doi.org/10.1007/s10639-017-9613-2.

Pachler, N., Cook, J., Bachmair, B., Kress, G., Seipold, J., Adami, E., & Rummler, K. (2010). Mobile learning: Structures, agency, practices. In *Mobile Learning: Structures, Agency, Practices*. https://doi.org/10.1007/978-1-4419-0585-7.

Pado, U. (2018). *Question Difficulty – How to Estimate Without Norming, How to Use for Automated Grading*. https://doi.org/10.18653/v1/w17-5001.

Perez, E, V., Santos, L, M, R., Perez, M, J, V., De Castro Fernandez, J, P., & Martin, R, G. (2012). Automatic classification of question difficulty level: Teachers' estimation vs. students' perception. Proceedings - Frontiers in Education Conference*,* FIE. https://doi.org/10.1109/FIE.2012.6462398.

Pérez, E, V, Santos, L, M, R., Pérez, M, J, V, Fernández, J, P. de C., & Martín, R, G. (2012). Automatic classification of question difficulty level: Teachers' estimation vs. students' perception. *2012 Frontiers in Education Conference Proceedings*, 1–5. https://doi.org/10.1109/FIE.2012.6462398

Perikos, I., Grivokostopoulou, F., Kovas, K., & Hatzilygeroudis, I. (2016). Automatic estimation of exercises' difficulty levels in a tutoring system for teaching the conversion of natural language into first-order logic. *Expert Systems., 33*, 569–580. https://doi.org/10.1111/exsy.12182.

Pimmer, C., Mateescu, M., & Gröhbiel, U. (2016). Mobile and ubiquitous learning in higher education settings. *A systematic review of empirical studies. In Computers in Human Behavior., 63*, 490–501. https://doi.org/10.1016/j.chb.2016.05.057.

Quinc, I., & Nac, E. (2011). *An investigation of reference question difficulty over time*. https://doi.org/10.17615/j4pv-hw82.

Romero, C., Ventura, S., & De Bra, P. (2009). Using mobile and web-based computerized tests to evaluate university students. *Computer Applications in Engineering Education, 17*(4), 435–447. https://doi.org/10.1002/cae.20242.

Şad, S. N., & Göktaş, Ö. (2014). Preservice teachers' perceptions about using mobile phones and laptops in education as mobile learning tools. *British Journal of Educational Technology., 45*, 606–618. https://doi.org/10.1111/bjet.12064.

Sarrab, M., Al-Shihi, H., & Khan, A, I. (2016). An empirical analysis of Mobile learning (M-learning) awareness and acceptance in higher education. *2015 International Conference on Computing and Network Communications, CoCoNet 2015*. https://doi.org/10.1109/CoCoNet.2015.7411307.

Sharples, M. (2013). Mobile learning: Research, practice and challenges. *Distance Education in China 3*(5), 5–11.

Sharples, M., Corlett, D., & Westmancott, O. (2002). The design and implementation of a mobile learning resource. *Personal and Ubiquitous Computing., 6*, 220–234. https://doi.org/10.1007/s007790200021.

Singhal, R., Goyal, S., & Henz, M. (2017). User-defined difficulty levels for automated question generation. *Proceedings - 2016 IEEE 28th International Conference on Tools with Artificial Intelligence, ICTAI 2016*. https://doi.org/10.1109/ICTAI.2016.0129.

Sullivan, T. (2001). Locating question difficulty through explorations in question space. Proceedings of the ACM International Conference on Digital Libraries. https://doi.org/10.1145/379437.379669.

Sung, Y. T., Chang, K. E., & Liu, T. C. (2016). The effects of integrating mobile devices with teaching and learning on students' learning performance: A meta-analysis and research synthesis. *Computers and Education, 94*, 252–275. https://doi.org/10.1016/j.compedu.2015.11.008.

Tan, G. W. H., Lee, V. H., Lin, B., & Ooi, K. B. (2017). Mobile applications in tourism: The future of the tourism industry? *Industrial Management and Data Systems., 117*, 560–581. https://doi.org/10.1108/IMDS-12-2015-0490.

Thamadharan, K., & Maarop, N. (2015). The acceptance of E-assessment considering security perspective: Work in Progress. *International Journal of Social, Behavioral, Economics, Business and Industrial Engineering, 9*(3), 874–879 http://www.waset.org/publications/10000820.

Tony, H., Cornelia, C., Seán, Ó. G., Kevin, B., Matthew, K., Sandy, S., Jeroen, B., Gerton, C., Wouter, H., Marie, E., Ton, K., Eria, M., & Panagiotis, K. (2020). Education in precarious times: A comparative study across six countries to identify design priorities for mobile learning in a pandemic. *Information and Learning Sciences, 121*(5/6), 433–442. https://doi.org/10.1108/ILS-04-2020-0089.

Torres Diaz, J. C., Torres Carrión, P. V., & Infante Moro, A. (2015). Mobile learning: Perspectives. *RUSC. Universities and Knowledge Society Journal, 12*(1), 38. https://doi.org/10.7238/rusc.v12i1.1944.

Uther, M. (2019). Mobile learning—Trends and practices. Education Sciences. https://doi.org/10.3390/educsci9010033, 9.

van de Watering, G., & van der Rijt, J. (2006). Teachers' and students' perceptions of assessments: A review and a study into the ability and accuracy of estimating the difficulty levels of assessment items. *Educational Research Review, 1*(2), 133–147. https://doi.org/10.1016/j.edurev.2006.05.001.

Wang, Y.-S., Wu, M.-C., & Wang, H.-Y. (2009). Investigating the determinants and age and gender differences in the acceptance of mobile learning. *British Journal of Educational Technology, 40*(1), 92–118. https://doi.org/10.1111/j.1467-8535.2007.00809.x.

Wang, Q., Liu, J., Wang, B., & Guo, L. (2014). A regularized competition model for question difficulty estimation in community question answering services. *EMNLP 2014–2014* Conference on empirical methods in natural language processing*,* Proceedings of the Conference. https://doi.org/10.3115/v1/d14-1118.

Wong, S. H. R. (2012). Which platform do our users prefer: Website or mobile app? *Reference Services Review., 40*, 103–115. https://doi.org/10.1108/00907321211203667.

Zhang, J., Liu, C., Yang, H., Feng, F., & Gong, X. (2017). *Automatically difficulty grading method based on knowledge tree*. https://doi.org/10.1007/978-3-319-63558-3_38.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Rishabh Singh[1,2] **· Devansh Timbadia**[1,2] **· Vidhi Kapoor**[2,3] **· Rishabh Reddy**[1,2] **· Prathamesh Churi**[1,2] **· Omkar Pimple**

✉ Prathamesh Churi
Prathamesh.churi@ieee.org

Rishabh Singh
rishabhmsingh@gmail.com

Devansh Timbadia
dev.timbadia44@gmail.com

Vidhi Kapoor
vidhikapoor20@gmail.com

Rishabh Reddy
rishabh3599@gmail.com

Omkar Pimple
Omkar@cerebranium.com

[1] Computer Engineering Department, Mukesh Patel School of Technology Management & Engineering, NMIMS University, Mumbai, India

[2] Cerebranium, Sepapaja 6, 15551 Tallinn, Estonia

[3] Data Science Department, Mukesh Patel School of Technology Management & Engineering, NMIMS University, Mumbai, India