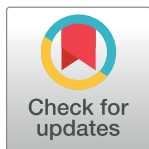# Digital case-based learning system in school

Peipei Gu[1], Jiayang Guo[2]*

1 Software Engineering College, Zhengzhou University of Light Industry, Zhengzhou, Henan, China,
2 School of Electrical and Computer Engineering, University of Cincinnati, Cincinnati, United States of America

* guojy@mail.uc.edu

Abstract

With the continuing growth of multi-media learning resources, it is important to offer methods helping learners to explore and acquire relevant learning information effectively. As services that organize multi-media learning materials together to support programming learning, the digital case-based learning system is needed. In order to create a case-oriented e-learning system, this paper concentrates on the digital case study of multi-media resources and learning processes with an integrated framework. An integration of multi-media resources, testing and learning strategies recommendation as the learning unit is proposed in the digital case-based learning framework. The learning mechanism of learning guidance, multi-media materials learning and testing feedback is supported in our project. An improved personalized genetic algorithm which incorporates preference information and usage degree into the crossover and mutation process is proposed to assemble the personalized test sheet for each learner. A learning strategies recommendation solution is proposed to recommend learning strategies for learners to help them to learn. The experiments are conducted to prove that the proposed approaches are capable of constructing personalized sheets and the effectiveness of the framework.

Introduction

With the rapid development of information technology, e-learning is becoming an inevitable trend of education reform throughout the world [1]. In our knowledge society, the requirements of continuing and lifelong education provide a vast platform for the fast development of e-learning. Because of the flexible learning time, various and abundant learning resources, and distance interaction, e-learning develop rapidly and change the development of education.

There are a variety of learning modes in e-learning environment: computer supported collaborative learning [2–4], personalized learning [5], adaptive learning [6], self-learning [7], distance learning [8], blended learning [9] and web-based learning [10], etc.

The quantity of online multimedia learning resources increases rapidly to fulfill the basic requirements of learning [11]. Along with the big data, machine learning technologies have been applied in learning system [12] [13]. And multimedia systems technologies applied in educational fields is of high interest and have been successfully and widely applied in teaching and learning for educational tools. A musical augmented reality system for children is

supported by Rodney et al. It is a useful educational tool, and especially in short and intense interactive learning conditions [14]. Pardo adopts "problem-project-based learning" to meet the dynamic setting in the field of engineering [15]. Hmelo-Silver et al. use video triggers and computer-based technology to facilitate cross-cultural groups in problem-based learning [16]. A system for 3D authoring and presentation in virtual reality environments is proposed by Osawa et al. to help users create 3D educational materials more easily [17]. An educational video compression technique that dynamically allocates the space on the grounds of the importance for each video segment in the educational videos by Mittal et al. [18]. Kulak et al. provides a representative review of case-based learning in science and describes the process of developing case-based learning modules adopted in biochemistry [19]. Krammer et al. conduct the intervention study of video analysis in teacher education to gain the video settings which impacts students' understanding [20]. Vilsmaier et al propose formats of case-based mutual learning sessions [21].

In our project, we aim at creating a simple environment for inexperienced programmers to build their programming patterns with a stimulating and specific training interface. It integrates multimedia learning resources and offers an easy assistant for learners to test themselves and acquire the guidance in the learning process.

The rest of this paper is organized as follows. Section 2 reviews related work of assembling algorithms for test sheets and approaches of recommending learning strategies in the computing and e-learning environment, respectively. Section 3 describes the framework integrated multi-media resources, learning guidance and testing to support digital case-based learning for variant learners. Section 4 presents our web-based testing algorithm and the solution recommending appropriate learning strategies for learners. Section 5 is the experiments and evaluation. Section 6 draws the conclusion of this paper and presents the future directions.

Related work

Algorithms for test sheet construction

The quality of the question bank which is as the candidate set for selecting questions to assemble the test sheet and the algorithm adopted in the construction procedure is the basis of the quality of a test sheet [22]. Many researchers have studied on the test sheet assemble algorithm. Some manually or randomly select test questions from question bank [23]. These approaches are easy, but low efficiency and could not meet the needs of multiple constraints generation situation. Then some researchers concentrate on studying intelligent test sheet generation problem to select an appropriate question set from the questions bank under the condition of multiple requirements [24]. Hwang adopts clustering techniques and dynamic programming to improve the procedure of test sheet with high quality according to specific requirements [24]. Hwang et al. present two improved genetic algorithms to construct test sheet to meet the needs of constraints of specified number of questions and specified range of questions [23]. Lee et al. present an Immune Algorithm to enhance the efficiency of near-optimal test sheet generation [25]. Yin et al. adopt particle swarm optimization (PSO) to improve the efficiency of generating near-optimal serial test sheets from large question bank for meeting multiple assessment requirements in test sheet generation [26].

Learning strategies recommendation

Appropriate adoption of learning strategies can contribute to the efficiency of learners learning procedure. Some researchers concentrate on recommending learning strategies adaptively in specific areas such as language and mathematics learning areas. Ghinea et al. concentrate on recommending learning strategies according to personal tutoring requirements. Some
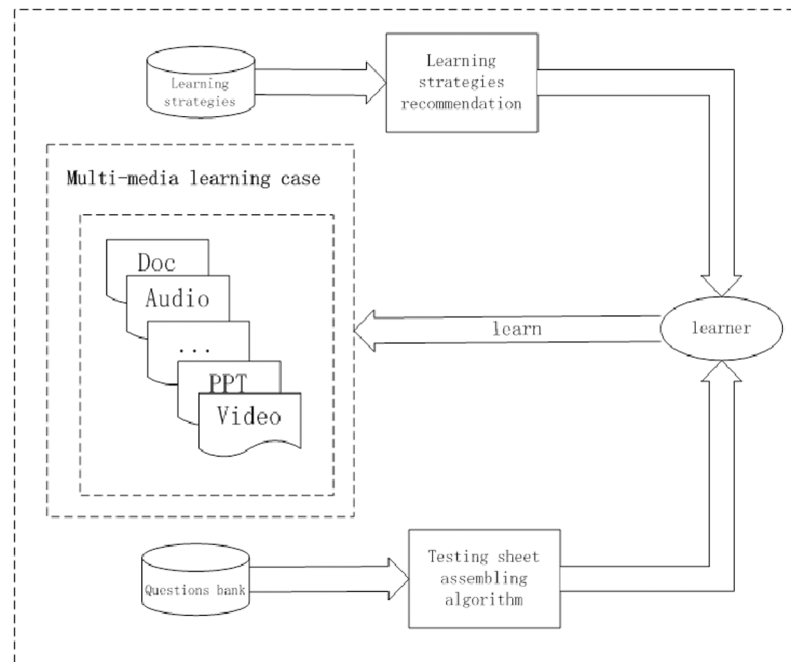
Fig 1. The framework of the digital case-based learning system.

researchers study on the experimental system in programming learning [27] [28]. Chang et al. develop a programming learning system for beginners with the completion strategy [29]. Paula et al. propose a recommendation system to help students in programming contests [30]. Motivation would aid learners to achieve efficient learning in higher education [31].

Digital case-based learning framework

In the case-based learning system, it is a problem how to integrate resources as the learning case.

One learning material could be audio, ppt, doc, or video. So it is an important question to help learners learning effectively with multi-media resources. In this section, every learning unit includes three parts: cases, learning strategies and testings. The cases are for learning, learning strategies are for offering an assistant to learners to learn better, and testings can provide learner getting his/her knowledge hierarchy. Learning strategies and testings can make the learning process easier.

A framework of digital case-based learning is proposed which covers the whole learning process. The proposed framework which is as Fig 1 shown consists of two components: web-based testing and learning strategies recommendation.

In web-based testing, a test sheet is generated according to multiple constraints to satisfy different learner's evaluation requirements. An improved personalized genetic algorithm to assemble personalized test sheet with more non-mastered questions and more questions not adopted frequently.

In learning strategies recommendation, a solution is proposed to solve the general learning strategies recommendation problem in the digital case-based learning system. It adopts decision tree to adjust the learning strategies recommended set.

Main technologies

The proposed framework covers learner's whole learning process. In this section, we emphasize on introducing web-based testing and learning strategies recommendation solution.

Web-based testing

Description of problem. The personalized test sheet generation problem is a multi-objective problem with multi-assessment constraints for student j. It concentrates on generating a test sheet which satisfies multiple constraints and levels of mastered knowledge points for student j by selecting a certain number of questions from a candidate question set $Qt_1, Qt_2, \ldots, Qt_n$.

There are four relative attributes of each question Qt_i: difficulty degree $diff_{i, 1 \le i \le n}$, discrimination degree $dis_{i, 1 \le i \le n}$, usage degree $u_{i, 1 \le i \le n}$, preference information $pref_{ji, 1 \le i \le n}$. The multiple test sheet constraints of the personalized test sheet generation problem include question quantity q, expected difficulty degree $diff$ and expected discrimination degree dis.

We assume that there are n questions $Qt_1, Qt_2, \ldots, Qt_n$ in the question bank. m knowledge points $Kp_1, Kp_2, \ldots, Kp_m$ are involved in the test. In our test subject, we assume that one knowledge point corresponds to many questions and one question belongs to one knowledge point.

The variables used in describing problem are defined as follows:

1) $x_{i, 1 \le i \le n}$: $x_i = 1$ represents that question Qt_i is included in the test, and $x_i = 0$ otherwise.

2) q: question quantity in the final test sheet.

3) $dis_{i, 1 \le i \le n}$: degree of the discrimination of Qt_i.

4) $diff_{i, 1 \le i \le n}$: degree of the difficulty of Qt_i.

5) $u_{i, 1 \le i \le n}$: degree of the usage of Qt_i.

6) dis: the expected discrimination degree of the whole test.

7) $diff$: the expected difficulty degree of the whole test sheet.

8) $pref_{ji, 1 \le i \le n}$: the preference information in Qt_i of student j, $pref_{ji} = 0$ represents that student j has mastered question Qt_i, and $pref_{ji} = 1$ otherwise (detailed calculation equation is shown as (4)).

9) $af\%$: expected percentage of level of non-mastered concepts of a subject in the test.

10) $Z(pref_{j1}, pref_{j2}, \ldots, pref_{ji}, \ldots, pref_{jn})$: preference information of student j.

The objective function can be defined as follows:

$$\text{Min } f = \frac{\left(\sum_{i=1}^{n} \left| diff_i - diff \right| x_i + \sum_{i=1}^{n} \left| dis_i - dis \right| x_i \right)}{\left(2 * \sum_{i=1}^{n} x_i \right)} - Z(pref_{j1}, \ldots pref_{ji}, \ldots pref_{jn}) \qquad (1)$$

where

$$Z(pref_{j1}, pref_{j2} \ldots pref_{ji}, \ldots pref_{jn}) = \begin{cases} 0.5 & , \quad \sum_{i=1}^{n} pref_{ji} x_i >= q * af\% \\ -0.5 & , \quad \sum_{i=1}^{n} pref_{ji} x_i < q * af\% \end{cases} \qquad (2)$$

In (1), variable x_i represents question Qt_i is chosen or not in the final test sheet. In (2), $pref_{ji}$ represents student j has mastered question Qt_i or not, $\sum_{i=1}^{n} pref_{ji} x_i$ represents the quantity of questions on the test sheet that are mastered by student j, and $q^* af\%$ indicates the expected quantity of non-mastered questions that should be selected in the final test sheet. The preference information $Z(pref_{j1}, \ldots, pref_{ji}, \ldots, pref_{jn})$ makes for narrowing the gap between final difficulty and discrimination degree and expected difficulty and discrimination degree.

Improved personalized genetic algorithm. In the previous work, to solve the personalized test sheet generating problem, we propose a personalized genetic algorithm named PGA which is an improved GA that incorporates students' preference information into crossover operator [32]. And based on this, incorporated with usage degree, we improve the former algorithm in crossover and mutation process to solve this problem effectively.

Preference information of student can help us better acquiring the learning level of the students. And the frequency of usage for questions can help constructing more effective test sheet. Improved personalized genetic algorithm (IPGA) can assemble test sheet satisfied students personal requirements.

In this paper, $pref_{jkps, \ 1 \leq s \leq m}$, the level of mastered knowledge point kp_s for student j and usage degree u_i are incorporated into the assembling process for constructing a personalized test sheet for student j. The $pref_{jkps, \ 1 \leq s \leq m}$ is defined as follows:

$$pref_{jkp_s, 1 \leq s \leq m} = \begin{cases} 0 & , YNum >= aNum * pf\% \\ 1 & , YNum < aNum * pf\% \end{cases}. \tag{3}$$

In (3), variable $YNum$ represents the right quantity of answering the questions which correspond to knowledge point kp_s for student j. Variable $aNum$ represents the quantity of answering the questions which correspond to knowledge point kp_s for student j. Variable $pf\%$ is the description of mastered level of knowledge point kp_s in answering right for student j. Different levels can be assigned based on the actual needs of the situations. For example, when more than $pf\%$ of questions are answered right, the mastered level of knowledge point kp_s is 0 (0 indicates student j has mastered the knowledge point kp_s, 1 is not).

The $pref_{ji, 1 \leq i \leq n}$ for question Qt_i which corresponds to knowledge point kp_s is as follows:

$$pref_{ji, \ 1 \leq i \leq n} = \begin{cases} 0 , & if \ student \ j \ answers \ Qt_i \ is \ right \\ 1 , & if \ student \ j \ answers \ Qt_i \ is \ wrong \\ pref_{jkp_s} , & if \ student \ j \ have \ not \ answewred \ Qt_i \end{cases}. \tag{4}$$

The usage degree of question Qt_i indicates the selected frequency in the former constructed test sheets. When students take an examination, some questions may be selected by teachers many times, so usage degree of question is incorporated into the construction procedure would be useful for the examination.

The definition of u_i is defined as follows:

$$u_i = 0.8^g \tag{5}$$

The initial value of u_i is 1. g is the frequency of occurrence of question Qt_i. Together with question Qt_i is selected in the final test sheet, the value of u_i is cut down. The lower u_i is, the more frequent question Qt_i has been selected in the final test sheet.
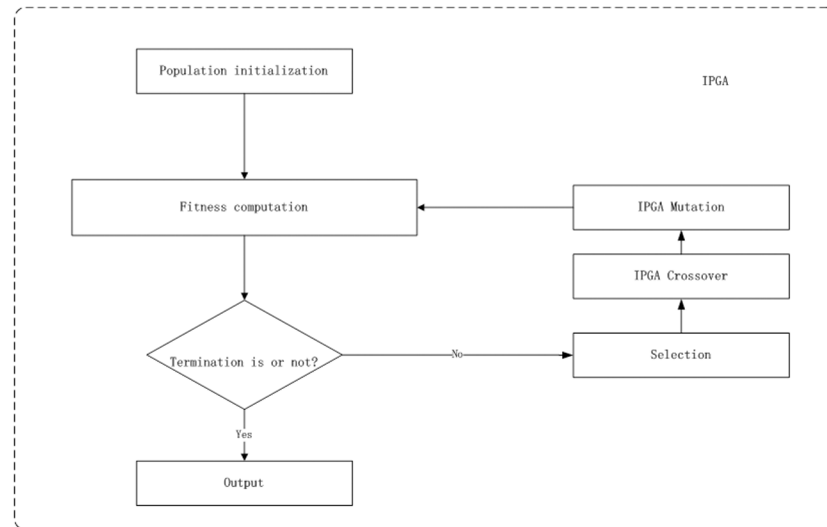
Fig 2. IPGA procedure.

IPGA process is based on the traditional GA. The procedure of IPGA is as Fig 2 shows. The description of the IPGA procedure is as follows:

1) Initialize the population.

2) Compute fitness with (1).

3) Go to step *8)* if the termination criterion is satisfied.

4) Select mother individual ind_{mum} and father individual ind_{dad} according to roulette algorithm.

5) Crossover ind_{mum} and ind_{dad}. (IPGA crossover procedure).

6) Mutate (IPGA mutation procedure).

7) Go to *2)*.

8) The best generation is the final test sheet.

IPGA incorporates preference information and usage degree into the crossover process as Fig 3 shows.
The description of the IPGA crossover procedure is as follows:

1) Decode mother individual ind_{mum} and father individual ind_{dad}.

2) Set the crossover location *s* is 0.

3) Get Qt_i and Qt_f in the corresponding location *s*.

4) Get pre_{ji} and u_i of question Qt_i, and get $pref_{jf}$ and u_f of question Qt_f.

5) Compare pre_{ji} and pre_{jf}, and u_i and u_f, if $pre_{ji} = 0$, $pre_{jf} = 1$ and $u_i < = u_f$ then exchange question Qt_i and Qt_f.

6) *s* plus one. Move crossover location *s* forward one position.

7) If *s* is the end of ind_{mum}, encode ind_{mum} and ind_{dad}, and then go to IPGA mutation procedure.

8) Go to 3).
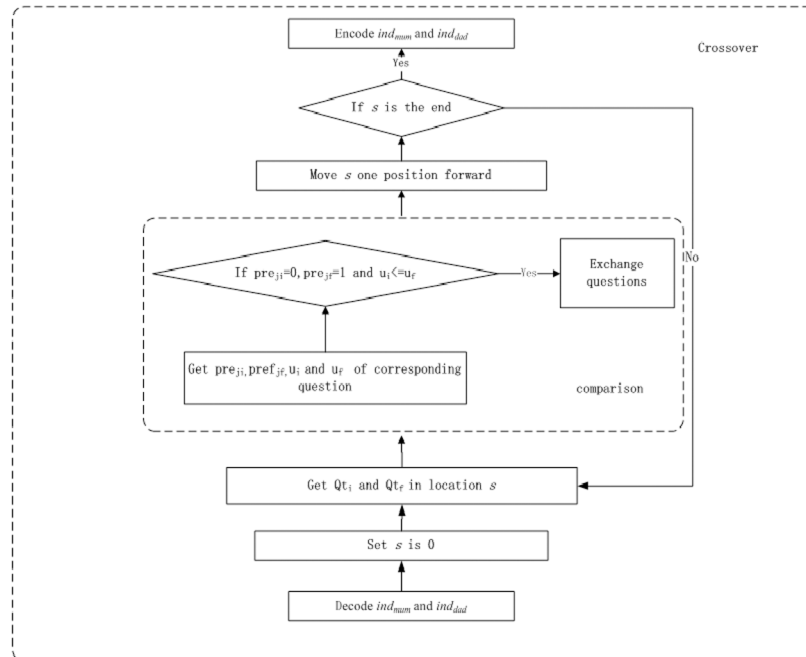
Fig 3. IPGA crossover procedure.

https://doi.org/10.1371/journal.pone.0187641.g003

Fig 4. IPGA mutation procedure.

https://doi.org/10.1371/journal.pone.0187641.g004
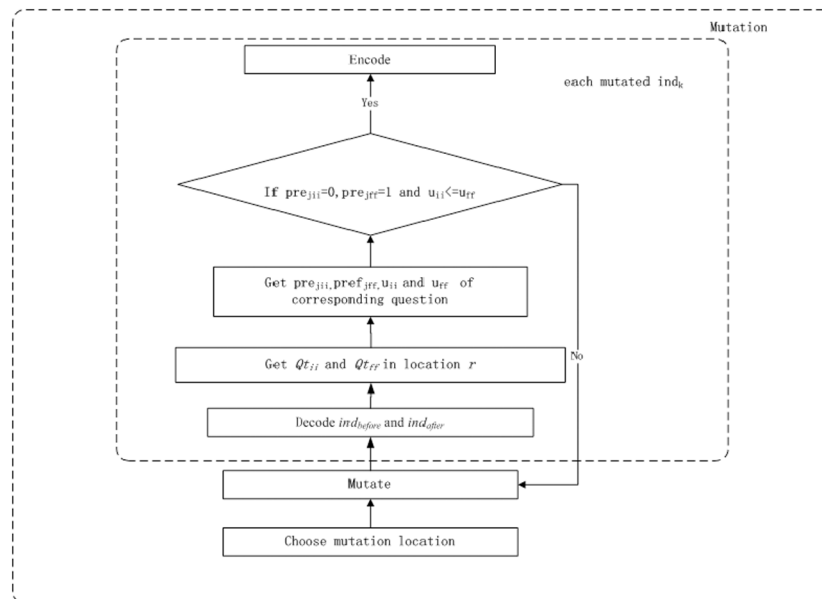
IPGA incorporates preference information and usage degree into the mutation process as Fig 4 shows.

The IPGA mutation procedure can be described as follows:

1) Choose mutation position randomly.

2) Mutation.

3) For each mutated ind_k.

3.1) Decode old individual ind_{before} and mutated individual ind_{after}.

3.2) Get Qt_{ii} and Qt_{ff} in the corresponding location r.

3.3) Get pre_{jii} and u_{ii} of question Qt_{ii}, and get $pref_{jff}$ and u_{ff} of question Qt_{ff}.

3.4) Compare pre_{jii} and $pref_{jff}$, and u_{ii} and u_{ff}, if the terminal condition ($pre_{jii} = 0$, $pref_{jff} = 1$ and $u_{ii} < = u_{ff}$) is not satisfied then go to 2).

3.5) Encode.

Learning strategies recommendation

In our previous work, we propose a learning strategies recommendation approach for the e-learning system which integrated with multiple learning sources, such as video, documents or some other teaching materials [33]. Based on the former general learning strategy recommendation algorithm, we propose a learning strategy recommendation solution for our digital case-based learning system.

Learning strategies. Scarcella & Oxford have researched the learning strategies and proposed the definition of learning strategies as "specific actions, behaviors, steps, or techniques—such as seeking out conventional partners or giving oneself encouragement to tackle a difficult language task—used by learners to enhance their own learning" [34].

In the digital case-based learning system, it is important to recommend learning strategies for learners for the learning strategies can help improve the ability effectively in using the multiple learning video resources.

In our previous work, we proposed a learner model to cover the various properties of learners for making the final learning strategies recommend list. We should collect five parts five parts of information: static properties, dynamic properties, affective information, history of learning strategy choosing, and test results of all tests [33] [35].

Five groups of learning strategies are adopted in our digital system:

1) Meta-cognitive strategies: It contains plan making and summarization of data. These learning strategies can support certain approaches to help learners studying prospective or summarize the previous knowledge.

2) Memory-related strategies: It contains learning and reviewing. These strategies can support a number of methods to improve the whole learning procedure.

3) Compensatory strategies: It contains some compensatory methods to avoid forgetting knowledge. In the normal learning process of acquiring knowledge, there must be some knowledge point we omitted or forgotten. These strategies can support some approaches to release early and iterate.

4) Affective strategies: It contains mental strategies, such as rewards or encouraging mechanism. These strategies can support certain approaches to re-build confidence and motivation of learners, attract learners' imagination.

5) Social strategies: It contains the ask and help from the community, friends, experts and other individual learners. It represents the openness of the digital system.
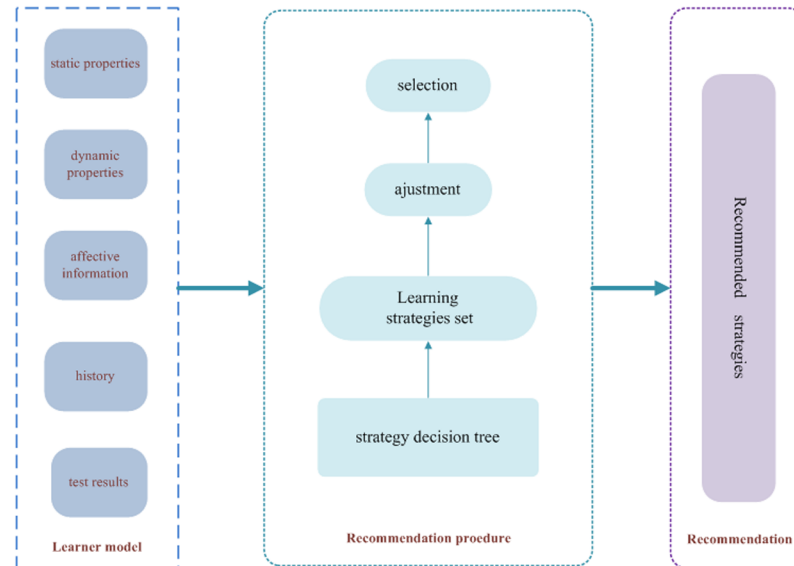
Fig 5. Learning strategy recommendation solution for the second and third type of learner.

A learning strategy recommendation solution. A learning strategy recommendation solution for recommending proper learning strategies is proposed which includes two aspects: modeling learners, recommendation procedure as Fig 5 shows.

There are three types of learners in the digital case-based learning system. 1) Unregistered learners or registered learners who are the newbies. 2) Registered learners who have no idea about the learning strategies recommendation solution or have no interest in the recommended learning strategies. 3) Registered learners who are familiar with the recommended solution and prefer adopting the learning strategies recommended.

For the first type of learner, we recommend top 2 from the learning strategies recommendation history of all learners ranking according to the calculation of (6).

Learning strategies recommendation process for the second and third type of learner can be described as follows:

1) Select learning strategy based on the learning strategies decision tree.

2) Add learning strategy as the candidate in learning strategies recommendation set.

3) Adjust the candidate learning strategies recommendation set.

4) Select top 2 from the candidate learning strategies recommendation set as the final recommendation set to offer.

A learning strategies decision tree is constructed based on C4.5 [36] (see S1 File).

In the solution, when learning strategies set is selected based on the learning strategies decision tree, we should execute adjustment on the set. The adjustment can be described as follows:

1) Select the favorite top 3 learning strategy.

2) Add them into the learning strategies candidate set.

3) Calculate the influence factor according to (6).

4) Rank learning strategies in the set based on the calculation of the influence factor *if*.

Table 1. Difficulty and discrimination degree classification.

Degree	Lowest	Lower	Normal	Higher	Highest
Value	1.0	2.0	3.0	4.0	5.0

We assume r is the frequency of occurrence of one learning strategy in the learning strategies adoption history. m is the size of learning strategies in adoption history. e is the recommended weight. If the learning strategy is recommended based on the learning strategies decision tree, e is 1, else e is 0.8 for the reason of balancing the learning strategy from the decision tree and the history.

The calculation rules of influence factor if is as follows:

$$if = e * \frac{r}{m} \qquad (6)$$

Experiments and evaluation

Web-based testing

To evaluate the performance of the proposed IPGA, a series of experiments are conducted by comparing it with Traditional GA in three aspects: execution time, usage degree, final result quality, and final result distribution between mastered and non-mastered questions.

The simulation experiments are conducted for constructing final test sheet which contains 20 questions of the best difficulty and discrimination degree with applying the algorithms 10 times. All algorithms used in the experiments are coded in Java Language conducted on a personal computer with Intel (R) Core (TM) 2 Duo CPU @ 2.53GHz and 1.93GB memory.

There are all the parameters of 10000 simulation questions which refer to 121 knowledge points in our candidate testing bank (see S2 File). There are 5 degrees of the difficulty and discrimination in the test sheet assembling procedure as described in Table 1.

Value 1.0~5.0 are adopted to represent the lowest~the highest difficulty and discrimination degree respectively. And $af\%$ which is the expected percentage of non-mastered questions that should be selected in the final test sheet is set to 60%.

There are 3 learners in the digital case-based learning system involving in these experiments, and they are Zhao, Qian and Sun. The level of mastered knowledge point is as described in Table 2. The percentage value is high means the better the learner get the corresponding knowledge point. Otherwise, the low percentage value indicates low learning level referring to the knowledge point. For example, the mastery degree of Qian is higher than Zhao and This would impact on the test sheet assemble procedure.

The experiments results of the difference of final average difficulty with expected difficulty in applying with IPGA and Traditional GA for 3 learners in 10 times are shown in Table 3. The experiments results of the difference of final average discrimination with expected discrimination in applying with IPGA and Traditional GA for 3 learners in 10 times are shown in Table 4. The expected difficulty and discrimination degree of *(diff, dis)* are *(1.0, 1.0), (2.0, 2.0), (3.0, 3.0), (4.0, 4.0) and (5.0, 5.0)*. The difference of (Diff-dif) and (Dis-dis) between IPGA

Table 2. Percentage of mastered knowledge point referring to different learners.

Learners	Zhao	Qian	Sun
Percentage	36.4%	58.7%	67.8%

Table 3. The difference of final average difficulty with expected difficulty between IPGA and Traditional GA.

(diff, dis)	Difference of (Diff-diff) between IPGA and Traditional GA		
	Zhao	**Qian**	**Sun**
(1.0, 1.0)	-0.0015	0.019	0.0155
(2.0, 2.0)	0.0055	0.0035	0.032
(3.0, 3.0)	-0.003	0.0075	0.0055
(4.0, 4.0)	-0.0035	0.01	-0.0195
(5.0, 5.0)	-0.018	-0.009	-0.1036

Diff is the final average difficulty degree in assembling 10 times procedure. diff is the expected difficulty degree.

and Traditional GA are negative which indicates IPGA gets small gap between final average degrees and expected degrees than Traditional GA and it means we could achieve better final difficulty and discrimination degree in applying with IPGA. As Tables 3 and 4 show, there are 17 negative values which mean we can get better final difficulty and discrimination degree in applying with IPGA 17 times and Traditional GA 13 times. So we achieve better experiment results with IPGA than Traditional GA in difficulty and discrimination degree.

The total quantities of non-mastered questions by applying IPGA and Traditional GA for learner Zhao, Qian and Sun are shown in Table 5. The difference of total quantities of non-mastered questions between IPGA and Traditional GA for 10 times are positive which indicates the quantity of non-mastered questions in applying with IPGA are more than Traditional

Table 4. The difference of final average discrimination with expected discrimination between IPGA and Traditional GA.

(diff, dis)	Difference of (Dis-dif) between IPGA and Traditional GA		
	Zhao	**Qian**	**Sun**
(1.0, 1.0)	0.009	0.0165	0.0335
(2.0, 2.0)	-0.006	0.0235	0.0295
(3.0, 3.0)	-0.0335	-0.0205	-0.019
(4.0, 4.0)	-0.015	-0.0445	-0.0195
(5.0, 5.0)	-0.0085	-0.0405	-0.0185

Dis is the final average discrimination degree in assembling 10 times procedure. dis is the expected discrimination degree.

Table 5. Experiment results of the difference of total quantities of non-mastered questions between IPGA and Traditional GA in applying 10 times on the best test sheet construction for 3 learners.

(diff, dis)	Difference of the total quantities of non mastered questions between IPGA and Traditional GA in 10 times		
	Zhao	**Qian**	**Sun**
(1.0, 1.0)	6	41	43
(2.0, 2.0)	31	25	42
(3.0, 3.0)	23	39	30
(4.0, 4.0)	13	27	23
(5.0, 5.0)	10	29	41

Table 6. Experiment results of difference of average usage degree between IPGA and Traditional GA.

(diff, dis)	Difference of the average usage degree between IPGA and Traditional GA		
	Zhao	**Qian**	**Sun**
(1.0, 1.0)	0.00965	0.00565	0.01005
(2.0, 2.0)	-0.00075	-0.00025	0.00455
(3.0, 3.0)	-0.00665	0.00715	-0.0021
(4.0, 4.0)	-0.0141	0.00565	0.0035
(5.0, 5.0)	-0.00325	0.00105	0.01935

https://doi.org/10.1371/journal.pone.0187641.t006

GA and it means we could achieve better final non-mastered questions in applying with IPGA. When assembling the test sheets, IPGA can get more non-mastered questions in final test sheet. IPGA achieves good non-mastered questions distribution than traditional GA.

The average usage degree by applying IPGA and Traditional GA for learner Zhao, Qian and Sun are shown in Table 6. The difference of average usage degree between IPGA and Traditional GA are positive which indicates we can get high usage degree in applying with IPGA. As Table 6 shows, there are 9 positive values which mean we can get final test sheet of higher usage degree 9 times with IPGA and Traditional GA 6 times. So we achieve more questions of low frequency with IPGA than Traditional GA.

Fig 6 presents the execution time of IPGA and Traditional GA. There is unusual 12806.9 milliseconds when applying with IPGA for learner Zhao in (1.0, 1.0) because of its questions distribution, so we remove the value from the final figure. As shown, IPGA consumes shorter
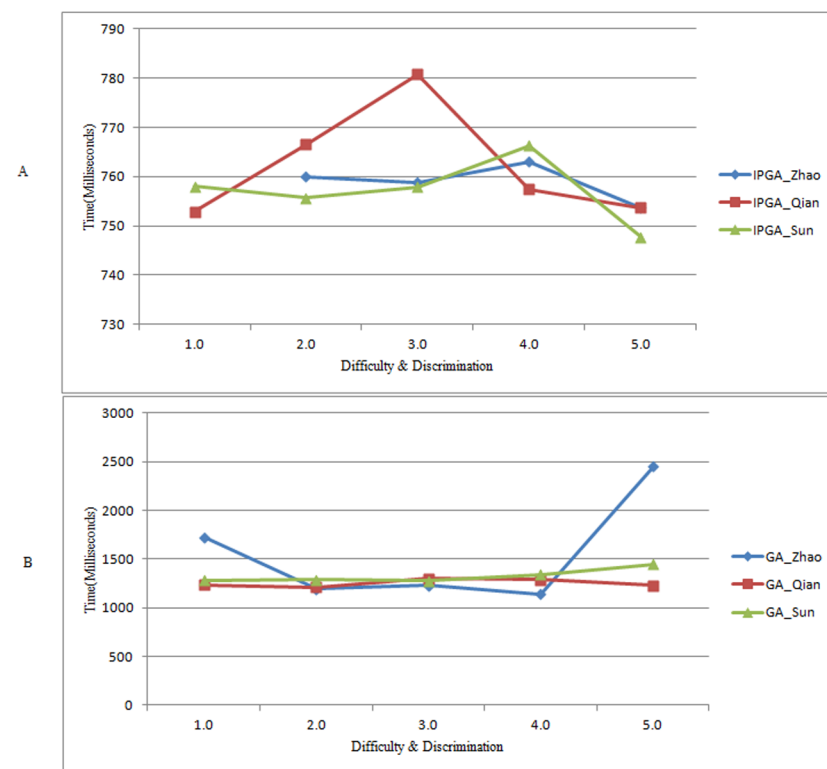

Fig 6. Average execution time in different difficulty and discrimination degrees for each learner.
Average execution time in applying with IPGA (A). Average execution time in applying with Traditional GA (B).

https://doi.org/10.1371/journal.pone.0187641.g006

time to construct test sheet satisfying multiple constraints for learners in most times. Improved crossover and mutation procedure are proved to be effective for optimization.

Learning strategies recommendation

In our project, final learning strategies list for learners are selected based on the learning strategies recommendation solution.

If a learner accesses system as a newbie, top 2 learning strategies are recommended according to the learners learning strategies adopting history. For the other registered learners, top 2 learning strategies are recommended based on learning strategies decision tree and adoption history. It can save time and energy for learners when only top 2 learning strategies recommended to them. For example, top 2 learning strategies of learner Zhao are video learning review and community help, so relevant video resources and specific information such as posts, experience, etc, are provided according to the learning strategies. The learner can access relevant learning materials on the home page.

When learners choose to learn video case, multiple assisted learning approaches are provided. Learners can comment, ask or take notes when learning the video case. And when learners finish the test, it marks automatically and rewards learners credits to attract and encourage learners.

Conclusion

This paper discussed and analyzed kinds of situation which constructing a case-based learning system of multimedia resources. Integrated with learning strategies recommendation and learning testing, the case-based learning system provides the personalized testing and appropriate programming learning strategies for learners.

In our experiments, IPGA algorithm could select availably personalized test sheet for the individual learner. Programming learning strategies recommended to learner achieved good evaluation from the learners. And the case-based learning system can effectively provide a whole learning procedure for the different learner.

In our future work, we will study on the integration and usage approaches of multimedia resources to exert the functionality of digital case-based learning system.

Supporting information

S1 File. LearningStrategiesDecisionTree.
(DOC)

S2 File. SimulatedDataSet.
(XLS)

S3 File. AssemblingResult.
(XLS)

Author Contributions

Investigation: Jiayang Guo.

Methodology: Peipei Gu.

Project administration: Peipei Gu.

Software: Jiayang Guo.

Writing – original draft: Peipei Gu.

Writing – review & editing: Jiayang Guo.

References

1. Kruse K. The Benefits and Drawbacks of e-Learning. It Educ. 2004;

2. Iinuma M, Matsuhashi T, Nakamura T, Chiyokura H. Student Awareness Change in Computer Supported Collaborative Learning (CSCL) Environment. 2016; 6: 448–452.

3. Ke F, Hsu YC. Mobile augmented-reality artifact creation as a component of mobile computer-supported collaborative learning. Internet High Educ. 2015; 26: 33–41.

4. Sadeghi H, Kardan AA. A novel justice-based linear model for optimal learner group formation in computer-supported collaborative learning environments. Comput Human Behav. 2015; 48: 436–447.

5. Ali SM, Ghani I, Latiff MSA. Interaction-based Collaborative Recommendation: A Personalized Learning Environment (PLE) Perspective. Ksii Trans Internet Inf Syst. 2015; 9: 446–465.

6. Paramythis A, Loidlreisinger S. Adaptive Learning Environments and e-Learning Standards. Electron J e-Learning. 2004; 2: 181–194.

7. Zhang R. Study on Improving English Reading Ability of the Electrical Automation Specialty Students in Multimedia-assisted Self-learning. Int J Emerg Technol Learn. 2016; 11: 23.

8. Umryk MA. ORGANIZATION OF DISTANCE LEARNING FOR MEETING THE NEEDS OF MODERN STUDENTS. Phys Status Solidi. 2015; 18: 699–709.

9. Garrison DR, Kanuka H. Blended learning: Uncovering its transformative potential in higher education. Internet High Educ. 2004; 7: 95–105.

10. Daramola JO, Bamigbola O. An Integrated Framework for Web-based Learning. 2015;

11. Huang W, Eze E, Webster D. Towards integrating semantics of multi-media resources and processes in e-Learning. Multimed Syst. 2006; 11: 203–215. https://doi.org/10.1007/s00530-005-0009-6

12. Zou Q, Zeng J, Cao L, Ji R. A novel features ranking metric with application to scalable visual and bioinformatics data classification. Neurocomputing. 2016; 173: 346–354.

13. Niu K, Niu Z, Su Y, Wang C, Lu H, Guan J. A Coupled User Clustering Algorithm Based on Mixed Data for Web-Based Learning Systems. Math Probl Eng. 2015; 2015.

14. Berry R, Makino M, Hikawa N, Suzuki M, Inoue N. Tunes on the table. Multimed Syst. 2006; 11: 280–289. https://doi.org/10.1007/s00530-005-0008-7

15. Pardo A. Problem-based learning combined with project-based learning a pilot application in Digital Signal Processing. 2014;

16. Hmelo-Silver CE, Lajoie SP, Chan LK, Khurana C, Lu J, Cruz-Panesso I, et al. Using online digital tools and video to support international problem-based learning. Proc Annu Hawaii Int Conf Syst Sci. 2013; 68–76. https://doi.org/10.1109/HICSS.2013.591

17. Osawa N, Kim S. Poster3D: A system for three-dimensional authoring and presentation in immersive virtual environments. Multimed Syst. 2006; 11: 216–225. https://doi.org/10.1007/s00530-005-0002-0

18. Mittal A, Gupta S, Jain S, Jain A. Content-based adaptive compression of educational videos using phase correlation techniques. Multimed Syst. 2006; 11: 249–259. https://doi.org/10.1007/s00530-006-0022-4

19. Kulak V, Newton G. A guide to using case-based learning in biochemistry education. Biochem Mol Biol Educ. 2015; 42: 457–473.

20. Krammer K, Hugener I, Frommelt M, Der Maur GFA, Biaggi S. Case-Based Learning in Initial Teacher Education: Assessing the Benefits and Challenges of Working with Student Videos and Other Teachers' Videos. Orb Sch. 2015; 2015: 119–137.

21. Vilsmaier U, Engbers M, Luthardt P, Maas-Deipenbrock RM, Wunderlich S, Scholz RW. Case-based Mutual Learning Sessions: knowledge integration and transfer in transdisciplinary processes. Sustain Sci. 2015; 10: 563–580.

22. Hwang GJ, Lin BMT, Lin TL. An effective approach for test-sheet composition with large-scale item banks. Comput Educ. 2006; 46: 122–139.

23. Hwang GJ, Lin BMT, Tseng HH, Lin TL. On the development of a computer-assisted testing system with genetic test sheet-generating approach. IEEE Trans Syst Man Cybern Part C Appl Rev. 2005; 35: 590–594.

24. Hwang GJ. A test-sheet-generating algorithm for multiple assessment requirements. IEEE Trans Educ. 2003; 46: 329–337.

25. Lee CL, Huang CH, Lin CJ. Test-Sheet Composition Using Immune Algorithm for E-Learning Application. International Conference on Industrial, Engineering, and Other Applications of Applied Intelligent Systems. 2007. pp. 823–833.

26. Yin PY, Chang KC, Hwang GJ, Hwang GH, Chan Y. A Particle Swarm Optimization Approach to Composing Serial Test Sheets for Multiple Assessment Criteria. J Educ Technol Soc. 2006; 9: 3–15.

27. Ghinea G, Chen SY. Perceived quality of multimedia educational content: A cognitive style approach. Multimed Syst. 2006; 11: 271–279. https://doi.org/10.1007/s00530-005-0007-8

28. Colace F, Santo M De. Adaptive hypermedia system in education: A user model and tracking strategy proposal. 2007; T2D–18–T2D–23.

29. Chang KE, Chiao BC, Chen SW, Hsiao RS. A programming learning system for beginners-a completion strategy approach. IEEE Trans Educ. 2000; 43: 211–220.

30. Paula LC De, Barbosa EF. A recommendation system to support the students performance in programming contests. Frontiers in Education Conference. 2014. pp. 1–8.

31. Jenkins T. The motivation of students of programming. Conference on Innovation and Technology in Computer Science Education. 2001. pp. 53–56.

32. Gu P, Niu Z, Chen X, Chen W. A personalized genetic algorithm approach for test sheet assembling. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 2011; 7048 LNCS: 164–173. https://doi.org/10.1007/978-3-642-25813-8_18

33. Niu Z, Gu P, Zhang W, Chen W. Learning Strategy Recommendation Agent. Springer Berlin Heidelberg; 2012.

34. Scarcella RC, Oxford RL. The Tapestry of Language Learning: The Individual in the Communicative Classroom. Cl Act. 1992; 16: 228.

35. Zhang W. Research on Learning Strategy-Oriented Service model in E-Learning. Beijing Inst Technol. 2012;

36. Quinlan JR. C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc.; 1993.