TEACHING EFFECTIVENESS OF INTELLIGENT TUTORING SYSTEMS

By

YANG HU

A dissertation submitted in partial fulfillment of
the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
School of Electrical Engineering and Computer Science

MAY 2019

© Copyright by YANG HU, 2019
All Rights Reserved
To the Faculty of Washington State University:

The members of the Committee appointed to examine the dissertation of YANG HU find it satisfactory and recommend that it be accepted.

Shira Lynn Broschat, Ph.D., Chair

Matthew E. Taylor, Ph.D., Co-Chair

Olusola O. Adesope, Ph.D.

Aaron S. Crandall, Ph.D.
I would like to thank Dominique Tilke, Taylor Adams, Zamzam Mumin, and my labmates for their contribution to this work.
In this dissertation, the author presents two projects regarding teaching strategies that apply to an intelligent tutoring system (ITS). The author applied multiple-solution teaching methods to the ITSs. The first project is an ITS that aims to help college students learn how to use Computer-Aided-Design (CAD) software, FreeCAD ITS. The second project is named SHiB ITS, and it seeks to help seniors learn the principles of finding the right spots to install a smart-home kit in their own house. In the two projects, the author experimented three teaching strategies, trial-and-error, practicing worked examples, and a combination of the two.

The results of the post-test survey of the FreeCAD ITS show that college student participants like to learn by practicing multiple-solutions. Also, students in the combination teaching strategy group performed better in post-tests than the ones in the other two groups. The results of the post-test survey of the SHiB ITS shows that both older and younger adults like to learn in having exercises of multiple solutions. The older adults benefit more from...
the trial-and-error teaching strategy, while the younger adults benefit more from the combination teaching strategy. The experimental results agree to each other.

The author then explored machine learning algorithms that uses an AI agent to teach another agent to perform a complicated task. The author proposed two algorithms that require a state importance input, a reinforcement learning (RL) algorithm that trains an RL teacher to teach, and an artificial neural network (ANN) that simulates behaviors of a teacher. Then the author applied the explored RL algorithm to train an RL teacher that knows how to perform tasks of SHiB ITS. The author added the trained teacher to the SHiB ITS and experimented the system with recruited participants. The results suggest that participants taught by the RL teacher performed better in the task that requires alternative solutions. In the post-survey, they significantly indicate that it is quick for them to learn to use the SHiB ITS with the RL teacher.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>ACKNOWLEDGMENT</strong></td>
<td>iii</td>
</tr>
<tr>
<td></td>
<td><strong>ABSTRACT</strong></td>
<td>iv</td>
</tr>
<tr>
<td></td>
<td><strong>LIST OF FIGURES</strong></td>
<td>x</td>
</tr>
<tr>
<td></td>
<td><strong>LIST OF TABLES</strong></td>
<td>xiii</td>
</tr>
<tr>
<td></td>
<td><strong>CHAPTER</strong></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Intelligent Tutoring System</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Teaching strategies for multiple solutions</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Teaching strategies regarding ages</td>
<td>4</td>
</tr>
<tr>
<td>1.4</td>
<td>Problems to address</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Background and Related Work</td>
<td>7</td>
</tr>
<tr>
<td>2.1</td>
<td>Teaching methods of CAD</td>
<td>7</td>
</tr>
<tr>
<td>2.2</td>
<td>ITSs for seniors</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>FreeCAD ITS and teaching effectiveness</td>
<td>14</td>
</tr>
<tr>
<td>3.1</td>
<td>FreeCAD Intelligent Tutoring System</td>
<td>14</td>
</tr>
<tr>
<td>3.2</td>
<td>Teaching effectiveness experiment</td>
<td>23</td>
</tr>
<tr>
<td>3.3</td>
<td>Results</td>
<td>24</td>
</tr>
<tr>
<td>3.4</td>
<td>Conclusion</td>
<td>29</td>
</tr>
</tbody>
</table>
4 SHiB ITS and teaching effectiveness ........................................ 31
   4.1 Purpose of the study .................................................. 32
   4.2 SHiB Intelligent Tutoring System .................................. 33
   4.3 Teaching effectiveness experiment ................................. 34
       4.3.1 Participants .................................................... 34
       4.3.2 Measurement method ........................................... 37
       4.3.3 Procedure ...................................................... 38
   4.4 Results ................................................................. 39
       4.4.1 Teaching strategy effectiveness analysis .................. 39
       4.4.2 Participants’ step length analysis .......................... 43
       4.4.3 Effects of training validation analysis of Experiment II .. 45
       4.4.4 Results of post-survey ................................. 46
   4.5 Discussion and Conclusion ........................................... 48

5 SHiB ITS in actual smart-home kit installation ...................... 52
   5.1 Methods ................................................................. 53
       5.1.1 Participants .................................................... 54
       5.1.2 Procedure ...................................................... 54
       5.1.3 Training Contents .............................................. 56
       5.1.4 Evaluation Metrics ........................................... 60
   5.2 Results and Discussion ................................................ 62
       5.2.1 Performance of Installation .................................. 62
       5.2.2 Time Spent ..................................................... 63
       5.2.3 Post-survey Results .......................................... 64
   5.3 Conclusion ............................................................ 66
   5.4 Additional experiment material .................................... 68
6  Additional Analysis of user data collected from the SHiB ITS exper-
   iment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 69
   6.1  Purpose of the study . . . . . . . . . . . . . . . . . . . . . . . . . . . 69
   6.2  Data type . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 70
   6.3  Related works . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 72
   6.4  An overview of the pattern of data - PCA . . . . . . . . . . . . . . . 74
   6.5  Apply mixed HMM to the data . . . . . . . . . . . . . . . . . . . . . . 79
   6.6  Conclusion and limitation . . . . . . . . . . . . . . . . . . . . . . . . . 82

7  Learn to Teach an agent to be a teacher . . . . . . . . . . . . . . . . . . 83
   7.1  Purpose of the study . . . . . . . . . . . . . . . . . . . . . . . . . . . . 83
   7.2  Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 85
      7.2.1  Parameter Effects . . . . . . . . . . . . . . . . . . . . . . . . . . . 85
      7.2.2  Approximate SARSA . . . . . . . . . . . . . . . . . . . . . . . . . 89
      7.2.3  RL teacher behavior modeling by ANN . . . . . . . . . . . . . . . 96
   7.3  Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100
      7.3.1  Results of Parameter Effects . . . . . . . . . . . . . . . . . . . . . 100
      7.3.2  Results of Approximate function . . . . . . . . . . . . . . . . . . . 103
      7.3.3  Results of ANN . . . . . . . . . . . . . . . . . . . . . . . . . . . . 106
   7.4  Conclusion and limits . . . . . . . . . . . . . . . . . . . . . . . . . . . . 108

8  Reinforcement Learning Teacher for SHiB ITS . . . . . . . . . . . . . . . 113
   8.1  Purpose of study . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 114
   8.2  Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 115
      8.2.1  Feature construction, Q-value re-evaluation, and reward shaping 115
      8.2.2  Recruitment and teaching outcome evaluation metrics . . . . . . 116
   8.3  Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 118
      8.3.1  Performance of RL student . . . . . . . . . . . . . . . . . . . . . . 118
8.3.2 Performance of participants ........................................ 124
8.4 Conclusion and limitations ........................................... 128

9 Conclusion and Future work ............................................. 131
  9.1 Summary of presented work ........................................ 131
  9.2 Future work ....................................................... 135
LIST OF FIGURES

3.1 Flow Chart of FreeCAD Intelligent Tutorial System .......................... 16
3.2 An overall view of user interface ................................................. 16
3.3 Details on the top of the user interface ....................................... 17
3.4 (a) Details of the right-hand side of the user interface.  (b) Goal model for
        pre-test, training section and post-test 1 and 2 .......................... 17
3.5 Three methods that employed in the textbook ............................. 19
3.6 Example of a method to construct solid model using FreeCAD tutorial . 20
3.7 User interface of training section in group 1 and group 3 ............... 21
3.8 An example of guidance to construct the goal model ..................... 22
3.9 An example of tutorial suggests ways to alert the user after an incorrect step 22
3.10 Goal model for (a) pre-test 3 and (b) post-test 4 ......................... 26
3.11 Percentage of participants test satisfaction ................................. 26
3.12 Average time spends at each section ....................................... 27

4.1 SHiB ITS user interface ....................................................... 35
4.2 Settings for training-exercise 1 (a) and training-exercises 2 and 3 (b) . 35
4.3 Settings for (a) post-test 1, (b) post-tests 2 and 3, and (c) post-test 4 .. 36
4.4 Exploration strategy in training-exercise 2 ............................... 36
4.5 Textbook strategy in training-exercise 1 ................................. 37
4.6 An illustration of the procedure ............................................ 39

5.1 Example of training contents for experimental group and control group .. 55
8.7 Comparision of number of steps participants spent in each training exercises 126


## LIST OF TABLES

3.1 Three methods instructed in textbook ................................. 19
3.2 Post-survey on FreeCAD tutorial experience .......................... 28
4.1 Participant performance in pre-test and post-tests .................... 41
4.2 Results of ANOVA test and Effect Size for participant performances in the four-posts .................................................. 43
4.3 Results of participants’ performance in the four post-tests, including the number (N), the mean step length, and the standard deviation of the step length ............................................. 44
4.4 Performance of participants of five groups in the post-tests .......... 46
4.5 Results of post-survey ...................................................... 47
5.1 Results of real installation .................................................. 63
5.2 Comparison of time spent in installation .................................. 64
5.3 Responses to Q13 of the experimental group and the control group .......................................................... 66
5.4 Questionnaire of post-survey ............................................... 68
6.1 Statistical results of PC1s and PC2s over 98 participants of 65 devices ........ 76
6.2 Mean accuracy of predictions of Normal-HMMs .......................... 81
8.1 Performance comparison of young adult participants across six groups .......................... 124
8.2 Step length comparison (a) training exercises, (b) post-tests ............ 125
8.3 Results of post-survey of six groups (a) results of aspect 1 through 5, (b) results of aspect 6 and 7 .................................................. 128
Chapter 1

Introduction

Education plays a vital role in human civilization. As Nelson Mandela mentioned that “Education is the most powerful weapon which you can use to change the world.” As technologies develop, the number of ways of study has increased gigantically. Intelligent tutoring systems (ITSs) (Anderson et al., 1985) belong to a category of educational technologies. Unlike a human instructor that can play many roles in education, ITSs are designed to represent different instructional roles, such as experts, tutors, mentors, and learning companions (Graesser et al., 2018). An ITS simulates a human tutor that practices with the student, cooperatively assisting the user in solving problems and learning skills or concepts. Most available ITSs concentrate on teaching children in topics including mathematics, engineering, chemistry, reading and writing, and so on. The U.S. education system begins to widely use ITSs with Cognitive Tutors, which are grounded in cognitive theory and cognitive modeling. Roughly over 560,000 students and 2,700 schools in all 50 United States employ ITSs in mathematics instruction (Eagle and Barnes, 2010). Evaluation studies also show that ITSs enhance learning compared to more typical forms of instruction (Waalkens et al., 2013).

The argument that encouraging learners to apply multiple solutions to a given problem improves learning efficiency has raised in education since 1989, with empirical evidence
coming from mathematics education. In-class studies indicate that having students compare multiple solutions for mathematics questions helps them make more significant gains in procedural knowledge and flexibility (Rittle-Johnson and Star, 2009; Rittle-Johnson et al., 2009; Waalkens et al., 2013). However, most ITSs require only a single solution in teaching. Few studies had investigated the effectiveness of ITSs apply the teaching strategies that could employ multiple-solutions.

In the remainder of this chapter, the author will introduce concepts of ITSs, teaching strategies for multiple solution teaching methods, the effectiveness of teaching strategies regarding user ages, and problems that will be addressed.

1.1 Intelligent Tutoring System

In a traditional classroom learning model, students are organized in classes that meet regularly at the same place at certain times of the day for a given length of time over a semester (Bates, 2015). On the other hand, an Intelligent Tutoring System (ITS) (Corbett et al., 1997) is a composite of a user interface, a tutoring model, a cognitive model, and a student model. A student interacts with the ITS via the user interface, such as submitting his or her answer or receiving feedback from the ITS. The tutoring model receives the student’s submission from the user interaction, transfers that to the cognitive model when it gets the feedback, and then it will respond to the user interface. The response depends on the ITS mechanism if the answer from a student is wrong, some ITSs will show hints to guide the student to the correct solution, and some ITSs will directly show the correct answer to the student. The tutoring model also talks about a student model for the sake of student’s performance recording, and some ITS tutoring model will retrieve student’s past learning outcomes to help the system making decisions of pedagogical plans. The cognitive model contains the knowledge of the domain, such as concepts, solutions to exercises, and so on. A student model includes the records of each student. Usually, it gives the learning
curve of the student, as well as the interaction log between the student and the ITS. The author presents two ITSs in this dissertation both record interaction records between users and the systems.

When a student is learning with an ITS, the tutoring system helps the student to focus on one question at a time. The system can check each step made by the student and figure out what aspects of the knowledge needs more exercises or instructions, and then assign more exercises related to that specific part or illustrate the experience with more details.

1.2 Teaching strategies for multiple solutions

Prior studies (Große, 2014; Levav-Waynberg and Leikin, 2012; Rittle-Johnson and Star, 2007) reported class experiment results which show students in the multiple-solution group performed better on the post-test than those in the control group. Encouraging multiple solutions problem solving is also employed by ITSs. However, most of the systems focus on teaching students mathematics and only a few focuses on other subjects (Hu et al., 2016; Tenison and MacLellan, 2014; Waalkens et al., 2013; Willenham, 2009). To the author’s knowledge, no prior published study has reported the effects of encouraging seniors to solve problems in multiple ways.

Practicing worked examples is a traditional way of learning. Examples can help teachers foster adaptive transfer among learners (Atkinson et al., 2000). One previous study (Bokosmaty et al., 2015) found that practicing worked geometry problems helped novice math students in understanding mathematical rules and problem-solving. Another study (Paas and Van Gog, 2006) suggests that practicing essential worked examples will enable learners to have more cognitive resources in cognitive activities. The main advantage of practicing worked examples within an ITS is that they reduce learning time while fostering conceptual understanding (Alexander, 2014).

Trial-and-error is another way of learning that needs the learners to explore the solution
on their own. The learning method is characterized by continuing to try different attempts
to get success or until the learner gives up. The learning approach is successful in simple
problems and games. People often use this method to learn fields that they have little
experience in, such as tries out strategies, keeping the new strategy if and only if it leads to
a higher reward (Young, 2009).

1.3 Teaching strategies regarding ages

A survey (Leung et al., 2012) of interactive learning environments found that people of
different ages have different learning preferences regarding teaching strategies. Seniors may
prefer to learn through step-by-step instruction, while young ones may prefer to learn by
trial-and-error. The survey reports senior people to prefer to learn alone, which fits into the
learning environments provided by ITSs. However, few studies (Kulik and Fletcher, 2016;
Vogel et al., 2006) focus on the effectiveness of different teaching strategies that are employed
by ITSs. Previous studies (Ribeiro and de Barros, 2014; Struve and Wandke, 2009) show that
senior participants benefit more from observing or experiencing trial-and-error as opposed
to instructional training. However, there also are studies (Bruder et al., 2014; Hagiya et al.,
2015; Leung et al., 2010; Toyota et al., 2014) that indicate senior participants learn better
by following instructional materials step-by-step. Little research focuses on comparing the
effects of the teaching strategies on senior participants.

1.4 Problems to address

In this thesis, the author aims to address four problems regarding Intelligent Tutoring Sys-
tems.

1. Can the contents taught by Intelligent Tutoring Systems (ITSs) extend to other areas?
2. What will be the most appropriate ITS teaching strategies applying multiple-solution for both young and senior adults?

3. What is the teaching effectiveness of applying multiple solution teaching methods in ITSs?

4. What are the effects of having an Artificial Intelligent agent as the teacher in an ITS?

For the first problem, the author developed two Intelligent Tutoring Systems. One is designed aims to help college students learn basic operations of CAD software. The other seeks to help seniors to learn the principles of choosing positions to place smart-home sensors in houses. The two ITSs differ from the subjects that available ITSs focus. The feedback of the post-surveys of the two ITSs was collected from participants on purpose to evaluate the utility of the ITSs.

For the second problem, the author applied three teaching strategies: explore, textbook, and combination. The textbook is the instructive teaching strategy, which is considered the control condition. Exploration is a trial-and-error teaching strategy; combination combines trial-and-error and instructive. Exploration and combination are considered the experimental conditions.

For the third problem, the author implemented an AI search algorithm and rules of finding the correct answers to let the tutorial systems contain multiple solutions for problems. In the FreeCAD tutorial system, the author implemented an AI search algorithm to compute all the possible action sequences to make each model. In the SHiB tutorial system, the author applied the smart-home device installation principles to the system. As principles are more flexible than fixed rules for situations. A valid solution works for a placement arrangement will not necessarily be a good one for another.

For the fourth problem, the author explored and developed three types of methods: intuitive method, reinforcement learning method, and artificial neural network method. With the consideration of generality, the author chose a Reinforcement Learning (RL) algorithm
to train an RL agent to learn teaching skills to teach users of SHiB ITS. The author trained the RL teacher by RL students. The RL teacher advises users before they placed the devices in a training exercise. Suggestions are not always given, and the RL teacher advises only when it thinks the user is in a situation that needs a piece of advice.
Chapter 2

Background and Related Work

In this chapter, the author will present background information and state-of-art of the teaching applications of Computer-Aided-Design (CAD) software, assistant applications for senior adults, and researchs regarding teaching methods.

2.1 Teaching methods of CAD

General Motors Research Laboratories started to develop Computer-Aided-Design (CAD) software in the early 1960s. Today, CAD software is an important industrial art extensively used in many applications, including automotive, shipbuilding, and aerospace industries, industrial and architectural design, prosthetics, and many more (Wikipedia contributors, 2018). Traditional engineering drafting approaches use pencil and ruler, and a design transform requires erasing and redrawing. CAD software translates the problem into changing model parameters so that design modifications become much quicker. Other advantages of CAD software are the zoom in and zoom out features, which simulate a camera lens, so that a designer can inspect details of elements of an assembly as well as evaluate the shape of an assembly as a whole. Three-dimensional (3D) solid models can rotate on any axis so that designers could have a full understanding of the object.
Mastery of CAD software has become necessary for mechanical engineering students and civil engineering students. In Washington State University, taking a CAD class is a prerequisite for Mechanical Engineering first-year students. However, a majority of universities often find it challenging to devote a significant amount of time to CAD instruction in the curriculum (Ault and Fraser, 2013) as there is a lack of computational tools help to train potential users to use CAD. Universities usually employ CAD software tutorials in their engineering graphics lab period to teach solid modeling skills and CAD software functions and features. There are numerous tutorials available for popular CAD software, and we summarize them into three types:

1. Tutorials published by the vendor

2. Tutorials published by professionals

3. Personal tutorials published on video websites

Most CAD software vendors have tutorials on their websites: Solidworks published static guidance of essential mechanical parts as well as videos on the website (Planchard, 2017) and the AutoCAD research team published embedded computational tutorials (Li et al., 2012) based on gamification theory to encourage learning. CAD software vendors’ tutorials, in both written material and interactive learning environment, are primarily skill-based and emphasize introducing software functions and features (Ault, 2011). The tutorial gives the learners a goal model, which will usually be a common mechanical part, such as screw nuts or shells, and then the tutorial will instruct the user to construct the given model step-by-step with CAD software tools. Tutorials published by professionals are usually more advanced and are in a textbook-like format (Toogood, 2009). Advanced tutorials focus on refining solid modeling skills and are often more complex than those of software vendors. They demonstrate excellent modeling skills by providing several similar exercises and helping the learner become familiar with these skills. The third type of tutorial is the pre-recorded
process of solid model construction, some of which also have oral instruction. Publishers of this type of tutorial include individuals, online education communities, and some are CAD vendors. One can say that the traditional way of CAD software learning is a one-way teaching style, in which students are required to follow every step of the tutorial materials and have to scrutinize mistakes by themselves as there is no interaction between students and the tutorial material.

Learning to use CAD software involves developing declarative and strategic knowledge such as selection of solid modeling alternatives and use of modeling constraints (Ault and Fraser, 2013). Strategic knowledge (Chester, 2007) is concerned with knowledge of the alternate methods by which a specific task may be achieved and the process by which a choice may be made. Teaching students using a single method to construct a solid model does not target learners “designing ingenuity” but is limited to the memorization of design procedures of ready-to-use objects, templates, and the related CAD software skills (Paliokas, 2009). Chester (2007) discussed the importance of strategic knowledge in teaching CAD software, the paper concludes that the employment of strategic knowledge early in the modeling process may prevent later model failure. Osakue (2015) published an instructional method on teaching solid modeling skills with AutoCAD. The method proposes that, at a planning stage, decomposing complex solid models into segments and sketching each segment isometrically helps students learn the extrusion operation concept of CAD software, as well as Boolean operations. The feedback from students shows that breaking down complex solids into segments helps students understand solid modeling principles and such principles will be helpful in the use of other CAD software. However, there are few ITSs that are developed to teach students solid modeling skills. There are also few empirical studies in the literature on ITSs that relate to the effectiveness of teaching multiple strategies (Waalkens et al., 2013).
2.2 ITSs for seniors

Relatively little ITS research has been conducted to help senior participants. One reason could be that designing an ITS for seniors could be more challenging than for young people. The challenges include (Wolfson and Kraiger, 2014): reduced cognitive speed, decreased working memory capacity, decreased the ability to focus, reduced ability to coordinate and integrate different sources of information, and decreased meta-cognition. Most of these challenges are related to age.

Struve and Wandke (2009) developed an interactive learning system, named ALISA, to teach senior participants to use the ticket vending machines for the Berlin Public Transport. The authors believed that allowing exploration and mistakes would not benefit seniors. The study compared two types of training: video training associated with guided errors and error-free training. In the guided error training, participants observed videos that had incorrect performance and the process to correct them. In the error-free training, participants observed the videos with correct performances. The study recruited both young and senior participants to both training groups. The post-test results show young participants are correct more often than senior participants. The time spent by young participants is also shorter than the time spent by senior participants, indicating that young participants solved tasks faster than senior participants. The analysis of additional steps taken to accomplish tasks shows that young participants took significantly fewer steps than senior participants. The performance of senior participants under guided error training was better than those under the error-free training. Hagiya et al. (2015) proposed a tutoring system to instruct senior participants how to input Japanese characters on smart devices. The article mentions that senior participants prefer an instructive teaching style over trial-and-error. Therefore, the typing ITS performs the role of a human tutor by identifying errors and guiding resolve mistakes. They compared the teaching effects of the tutoring system with human tutors, showing that the extent to which participants improved via the tutoring system was similar.
to those that learned via human tutors.

Ribeiro and de Barros (2014) experimented to evaluate and compare two instructional approaches: video and an interactive tutorial system, in teaching senior participants to perform a task on a smartphone. The results show participants in the tutorial group performed better than those in the video group in the task of “adding a new alarm,” but no statistically significant difference was observed. Toyota et al. (2014) proposed a self-learning smartphone app for novice seniors. The app teaches six smartphone operations (Tap, Resume, Unlock, Move, Zoom in/out, and Rotate). The results of the pilot and full study show senior participants prefer a pre-recorded video demonstration instead of an animation. Bruder et al. (2014) presented an adaptive training system to instruct novice seniors on a mobile phone task. The interface of the training system adjusts its complexity to the participants’ experience and their capabilities. They embed the training system into a touchscreen computer to simulate the interface of a mobile phone. The study investigated two ways of adaptation: one is an adaptive user interface that modifies complexity and the second is adaptive in respect to training advice. Results show that the adaptive user interface complexity was better suited for supporting senior participants with little exposure to a mobile phone. The study suggests training systems for senior participants should provide complete instructions for each task, as well as interactively teach the content step-by-step, and adapt the complexity to participants’ experience. Leung et al. (2010) proposed a multi-layered interface design approach that instructs senior participants to use an address book application on a mobile phone. They compared the proposed approach to a control approach whose interface is more complicated. They compared the learning outcomes between young participants (age 21-36) and senior participants (age 65-81). The results indicate that seniors benefit more from an adaptive complexity interface than younger people.

A traditional way of learning is practicing worked examples, which means the tutor gives learners a problem statement and a procedure for solving the problem. Then the learners follow the tutor’s demo to understand the process of solving the example problem and
then to learn to transfer the solution to questions that are in the same type as the example. Examples can help teachers foster adaptive, flexible transfer among learners (Atkinson et al., 2000). Bokosmaty et al. (2015) investigate the effectiveness of teaching learners to solve geometry problems by studying worked examples. Their findings reinforce the importance of problem solvers with little (or no) experience in studying mathematical rules and problem-solving steps. Paas and Van Gog (2006) published a review concerning experiments in investigating possible ways to increase the germane cognitive load in studying worked examples. A reason for improving students’ cognitive load during learning is for students’ attaining the transfer from examples to novel questions (Catrambone, 1998). However, Paas and Van Gog (2006) suggests that methods of practicing worked examples should try to reduce the extraneous load to enable learners to allocate more cognitive resources to relevant cognitive activities. Alexander (2014) stated that the main advantage of practicing worked examples within an ITS is that they reduce learning time while fostering conceptual understanding because examples relieve learners of problem-solving that, for novice learners, is slow, error-prone, and driven by superficial strategies.

The argument that encouraging learners to apply multiple solutions to a given problem improves learning efficiency has been discussed in education since 1989 (Große, 2014), with the majority of empirical evidence coming from mathematics education. Prior studies (Große, 2014; Levav-Waynberg and Leikin, 2012; Rittle-Johnson and Star, 2007) reported class experiment results of encouraging students to solve problems with multiple solutions. The results show students in the multiple-solution group performed better on the post-test than those in the control group. Encouraging multiple solutions problem solving is also employed by ITSs (Hu and Taylor, 2016; Tenison and MacLellan, 2014; Waalkens et al., 2013; Willenham, 2009), but most of the ITSs focuses on teaching mathematics (Tenison and MacLellan, 2014; Waalkens et al., 2013; Willenham, 2009). Only a few of them focus on another subject, such as (Hu and Taylor, 2016) teaching learners to draw 3D models in multiple ways. Most of the studies that encouraged learners to solve a problem in multiple
ways focus on young people, such as students in elementary school, high school, or college. To our knowledge, no published prior study has reported the effects of encouraging seniors to solve problems in multiple ways.
Chapter 3

FreeCAD ITS and teaching effectiveness

In this chapter, the author presents a design of a Computer-Aided-Design (CAD) Intelligent Tutoring System, namely, FreeCAD ITS. The author conducted an on-campus experiment to evaluate the teaching effectiveness of the system. This chapter includes the contents of the experiment results, the corresponding analysis, and conclusion. Section 3.1 includes details of design the system, section 3.2 includes the experiment method, section 3.3 includes the preliminary results and analysis, section 3.4 includes the conclusions draws from the the experiment.

3.1 FreeCAD Intelligent Tutoring System

The FreeCAD intelligent tutorial system aims to teach basic CAD skills. In this chapter, we focus on the skill, namely Boolean operation (union, intersect, and subtract), which is necessary for understanding solid modeling principles (Osakue, 2015). The tutorial system was developed for a parametric 3D software, FreeCAD, that allows solid models to be modified by changing parameters (Falck et al., 2012). The tutorial system can present three different
instruction approaches, which are described in section 3.2. The reason for choosing FreeCAD is because it is highly customizable, scriptable, and extensible. FreeCAD allows developers to build their own interfaces as well as functional components. The tutorial system is built as a workbench, a tool that with customized user interface. The program is developed in Python 2.7 from scratch. We build all basic primitives for users with python-based scripts. It is not necessary for learners to build basic objects by themselves, which allows them to focus on learning principles of solid modeling. We built “go back” and “restart” functions that allow users retrace their own steps. Boolean operation functions are imported from the FreeCAD library. For each 3D model that were set as a goal, we applied a search and planning algorithm, an AI approach, to compute all possible construction sequences under certain constraints.

The system consists of five parts: a introduction, a pre-test, a training section with three exercises, a post-test section with at most four tests, and a post-survey. A flow chart of the tutorial process is shown in Figure 3.1.

At the project introduction stage, the tutorial briefly introduces the overall tutorial process and user interface. Participants were required to enter age, major and group number. There are three groups that correlate to the three instructional methods. Figure 3.2 shows an overview of the user interface of the tutorial system, Figure 3.3 shows the detail on the top, Figure 3.4a shows the detail of the right-hand side components in the interface. Learners can select primitives, boolean operations, and task submission from the top toolbar. On the right-hand-side, there is a task description, an animation that demonstrates the goal model, and available options. A timer is embedded at the bottom on right-hand-side, which begins a count-down when the task starts. The center area is where the solid model were constructed. FreeCAD supports navigation of solid models through rotation view (FreeCAD contributors, 2016), which is a general function available for most CAD software. The tutorial has a reminder in the interface to help learners learn this skill.

During the pre-test, a learner were given five minutes to construct a given model, shown
Figure 3.1 Flow Chart of FreeCAD Intelligent Tutorial System

Figure 3.2 An overall view of user interface
Figure 3.3 Details on the top of the user interface

Figure 3.4 (a) Details of the right-hand side of the user interface. (b) Goal model for pre-test, training section and post-test 1 and 2
in Figure 3.4b. The solid model is a famous one that can be constructed from primitives by a constructed solid geometry (CSG) tree. Nevertheless, there is no previous research to construct this model by using multiple ways of combining Boolean operations. Under the constraints of using four operations (1 intersect, 1 subtract and 2 unions) and five primitives (3 cylinders, 1 sphere, and 1 cube), our planning algorithm computed more than one hundred methods that can construct the CSG model successfully. We expected that most participants would be unable to solve the pre-test successfully if they had no prior CAD experience.

The content of the training section for each group is different. Group 2 is the textbook mode and is considered to be the control group. Learners in group 2 were given a paper handout we designed to teach three methods to construct the model. We choose three fundamentally distinct solutions, allowing the learner to compare them during the training. A demonstration of the three methods is shown in Figure 3.5, the static description of their sequences is shown in Table 3.1. An example of the handout is shown in Figure 3.6, which demonstrates a sequence of Boolean operations to the learner: union the three cylinders first (step 1 to step 4, Figure 3.6a) and then subtract the union from the sphere (step 5 to step 7, Figure 3.6b), and then intersect the cube with the common part from last Boolean operation (step 8 to step 9, Figure 3.6c).

Group 1 and Group 3 are exploration mode and partial guidance mode, respectively, and their training is guided by the tutorial system. An example of the user interface is shown in Figure 3.7. It consists of three components at the right-hand-side of the screen. The top is a comprehensive icon introduction for the tutorial, which shows a primitive’s name and a Boolean operation’s name. The suggestion window is located under the icon instruction.

The content of the suggestion window depends on the subject group and the selected Boolean operation. The training section of group 1 is in exploration mode; the suggestion window will not give any guidance at the beginning but will encourage learners to try whatever he or she thinks is correct. When an incorrect operation is made, the tutorial will give advice regarding the correct operation. The training section of group 3 is in partial guidance
**Figure 3.5** Three methods that employed in the textbook

**Table 3.1** Three methods instructed in textbook

<table>
<thead>
<tr>
<th>Method #</th>
<th>Step</th>
<th>Boolean Operation</th>
<th>Primitive 1</th>
<th>Primitive 2</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method 1</td>
<td>1</td>
<td>Union</td>
<td>cylinder x</td>
<td>cylinder 1</td>
<td>fusion 1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Union</td>
<td>fusion 1</td>
<td>cylinder z</td>
<td>fusion 2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Subtract</td>
<td>sphere</td>
<td>fusion 2</td>
<td>cut</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Intersect</td>
<td>cut</td>
<td>cube</td>
<td>goal model</td>
</tr>
<tr>
<td>Method 2</td>
<td>1</td>
<td>Intersect</td>
<td>cube</td>
<td>sphere</td>
<td>common</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Union</td>
<td>cylinder x</td>
<td>cylinder 1</td>
<td>fusion 1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Union</td>
<td>fusion 1</td>
<td>cylinder z</td>
<td>fusion 2</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Subtract</td>
<td>common</td>
<td>fusion 2</td>
<td>goal model</td>
</tr>
<tr>
<td>Method 3</td>
<td>1</td>
<td>Union</td>
<td>cylinder x</td>
<td>cylinder 1</td>
<td>fusion 1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Intersect</td>
<td>cube</td>
<td>sphere</td>
<td>common</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Union</td>
<td>fusion 1</td>
<td>cylinder z</td>
<td>fusion 2</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Subtract</td>
<td>common</td>
<td>fusion 2</td>
<td>goal model</td>
</tr>
</tbody>
</table>
Figure 3.6 Example of a method to construct solid model using FreeCAD tutorial
The tutorial system knows all possible operation sequence paths to achieve the goal. In the training section, if it finds that the present step deviates from the correct path, it will give all possible paths that direct the learner in the right direction. For example, an incorrect step is shown in Figure 3.9; the learner has subtracted a cylinder from the sphere, which is incorrect. The tutorial will warn that the present step is incorrect and will offer 2 possible ways foward step. Figure 3.9 shows the incorrect step and suggestions from the tutorial.

For all three groups, the training section lasts at most 20 minutes. Once the learner constructs the solid model in three different ways he or she can end this section. A timer is
Figure 3.8 An example of guidance to construct the goal model

Figure 3.9 An example of tutorial suggests ways to alert the user after an incorrect step
embedded at the bottom at right-hand-side. The timer begins countdown when the section starts.

Post-test 1 and post-test 2 requires the learner to construct the solid models that have practiced in the training section, but the learner has to use two different sequences of steps. The two tests must be finished within five minutes, they serve as review tests for the contents that had been instructed in the training section.

The task in post-test 3 is to construct a new model with only four steps (1 intersect, 1 subtract and 2 unions) and five primitives (4 spheres and 1 cube). The model is shown in Figure 3.10a. It can be constructed using Boolean operation sequences similar to those in the training section. This task serves as a check of the learner’s mastery of solid modeling skills. Demonstrating a single solution sequence is sufficient to pass the test.

Post-test 4 is optional and was added as a more difficult challenge. The goal model of this test is shown in Figure 3.10b. To pass this test, learners need to use a Boolean operation sequence with four steps (1 union, 1 subtract, 2 intersect) and five primitives (2 spheres, 1 cube, and 1 cylinder). We employed the same AI algorithm to compute all possible solutions, from which we obtained more than 300 sequences to successfully construct the model. The solution pool covers methods that participants had applied in post-tests.

### 3.2 Teaching effectiveness experiment

The research is motivated by our desire to explore novel instruction methods for basic solid modeling skills in CAD software, as well as to evaluate the effectiveness of employing strategic flexibility in CAD sketch skills. It aims to validate three hypotheses:

1) Teaching multiple strategies with an ITS is more effective than a textbook

2) Content taught by textbooks could be straightforward for students to follow, but may not help solve new problems
3) Combining guidance and practice is more effective than only using a textbook or following an exercise step-by-step

In our preliminary experiment, we will test three teaching strategies: 1) *exploration mode*, in which participants have to figure out three strategies by themselves; 2) *textbook mode*, in which participants will follow three strategies from a textbook; and 3) *partial guidance mode*, in which participants were instructed on how to perform two solutions at the beginning and then are requested to figure out a third solution on their own.

Each group will have four post-tests. The first and the second post-test require the participant sketch previous models, using two different methods. The purpose of the two tests is to validate hypotheses 1 and 2. The third post-test asks participants to construct a new model. The fourth post-test is optional and its purpose is to both challenge the student and to make the experiment more interesting. The purpose of the third and fourth post-tests is to validate hypothesis 3.

### 3.3 Results

We carried out an experiment using our FreeCAD tutorial, which has been approved by institutional review board (IRB), and recruited 14 students by means of campus advertisement. The 14 students were from a variety of majors: 4 are in Mechanical Engineering, 5 are in Civil Engineering, 2 are in Computer Science, 1 is from Chemistry, 1 is from Education Psychology, and 1 is from Material Science. Each student has been rewarded $15 as compensation. They had been randomly divided into three groups. There are four in group 1, five in group 2 and five in group 3. All collected data are stored anonymously. The duration of the experiment is approximately 45 minutes.

Percentages of students who successfully passed tests are shown in Figure 3.11 (i.e., those participants who could construct the goal model within the time limit). The pass rate of group 1 and group 3 are higher than group 2, suggesting that using an ITS during training
is more effective than following written instructions in the textbook mode.

Students who pass post-test 3 indicate they mastered the skill of using Boolean operations by constructing a model they were not explicitly trained on. Students who pass post-test 4 indicate an even higher level of understanding. The pass rate of post-test 3 in group 1 and group 3 are higher than in group 2. The pass rate of post-test 4 in group 3 is higher than in both group 1 and group 2, suggesting that a combination of guidance and practice is more effective in solving new problems than only using a textbook or guided exercises. Note that the guidance in the textbook was designed to have the same overall quality as the guidance given to group 3.

The average time spent on each section is shown in Figure 3.12. As the FreeCAD ITS moves to the next task right after the participants finished the current one. The length of time reflects the efforts participants had put in solving a task. Therefore, the length of time indicates the difficulty of that section, which is valuable to validate our hypothesis 2. The training section of group 1 has the longest time, and group 2 has the shortest. We compared textbook group participants’ activity logs in training with the methods in the textbook and found out that participants in the textbook group followed all the steps. The results show the length of time spent in the three training sections suggests that the training task of textbook group is more comfortable (or at least faster) to follow than interacting with the intelligent tutorial system. The textbook group has the lowest pass rates on post-test 3, suggesting that content taught by textbooks may be straightforward for students to follow but not be as helpful in solving new problems.

The post-survey is collected at the end of the experiment. It is a questionnaire asking participants about their usage experience. The questionnaire has 15 statements, Question 1 to Question 4 are about gender, previous experiences of CAD software and the level of interest in geometry math class, respectively. Table 3.2 lists Question 5 to Question 14 and the average rating of responses to the statements in the questionnaire is given in the last column. Because of the sample size limitation, the conventional statistical analysis is of limited value.
Figure 3.10 Goal model for (a) pre-test 3 and (b) post-test 4.

Figure 3.11 Percentage of participants test satisfaction
Question 15 is about how comfortable the participant is with computers. Recruited students are requested to rate these statements with 1: strongly disagree, 2: disagree, 3: undecided, 4: agree, 5: strongly agree. 14 students responded to the questionnaire. The minimum average response is 2.86 (Question 10) and one may conclude that the students believe the pre-test model was not easy, which helps to explain why the average time spent on pre-test is higher than it spent on post-tests. The average response of Question 11 shows that students largely agree that after training the solid models, post-test 1 and post-test 2 are easy to pass. The maximum average response is 4.21 (Question 7), which deals with subjective feelings to instruction with multiple solutions. The average rating to this statement demonstrates that students believe teaching multiple solutions helps them to understand solid modeling construction principles. Based on the overall responses in Table 3.2, and data from our preliminary experiment, we believe that using our FreeCAD ITS holds promise.
<table>
<thead>
<tr>
<th>Question #</th>
<th>Statement</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Do you agree with the following statement: I enjoyed using this method to learn to sketch a CAD model.</td>
<td>3.79</td>
</tr>
<tr>
<td>6</td>
<td>Do you agree with the following statement: It makes completing sketching easier than I expected.</td>
<td>3.57</td>
</tr>
<tr>
<td>7</td>
<td>Do you agree with the following statement: Multiple solutions is an effective teaching method</td>
<td>4.21</td>
</tr>
<tr>
<td>8</td>
<td>Do you agree with the following statement: Completing training was not frustrating</td>
<td>3.71</td>
</tr>
<tr>
<td>9</td>
<td>Do you agree with the following statement: I did not have to work hard to complete training.</td>
<td>3.43</td>
</tr>
<tr>
<td>10</td>
<td>Do you agree with the following statement: The pre-test was easy.</td>
<td>2.86</td>
</tr>
<tr>
<td>11</td>
<td>Do you agree with the following statement: After training, making the first two objects was easy.</td>
<td>3.79</td>
</tr>
<tr>
<td>12</td>
<td>Do you agree with the following statement: After training, making the final (novel) object was easy.</td>
<td>3.00</td>
</tr>
<tr>
<td>13</td>
<td>Do you agree with the following statement: Learn multiple ways to construct the same object was easy.</td>
<td>3.86</td>
</tr>
<tr>
<td>14</td>
<td>Do you agree with the following statement: I understand Boolean Operations better than I did before.</td>
<td>3.86</td>
</tr>
</tbody>
</table>
3.4 Conclusion

We have assessed a novel instruction method to teach basic solid modeling skills for CAD software. The tutorial aims to teach Boolean operation (union, intersect, and subtract) skills, employing multiple strategies as teaching method during instruction. We investigate three instruction methods: exploration mode, textbook mode, and partial guidance mode.

We carried out a preliminary experiment to validate our hypotheses with 14 participants. Results of our tests provide preliminary data indicating that using an ITS is more effective than using textbook (hypothesis 1), a combination of guidance and practice will be more effective in solving new problems than only use textbook or exercise (hypothesis 3), and the textbook is easier to follow than an ITS but is not as helpful in solving new problems (hypothesis 2).

The contribution of this research includes the following. First, we designed and implemented an intelligent tutoring system for the FreeCAD software system. Second, our ITS was able to automatically discover and demonstrate multiple solution methods for a single goal model. Third, we performed a preliminary evaluation of teaching effectiveness for three different ways of learning CAD software. Fourth, our results suggest that a combination of suggestions and guided exploration allow students to best apply their learned knowledge to novel tasks.

The primary limitation of this research is sample size: we expect to recruit more students to the experiments in the future. To enlarge the sample size, crowdsourcing would be an option to recruit a more diverse set of people to evaluate teaching effectiveness. Future work could also focus on improving the training section through a machine-learning algorithm that could discover the optimal time to interrupt a participant and to give explicit guidance towards the exact issue the participant is working on. An additional method to provide guidance to a participant could be via animation, rather than static text, as was suggested by some of the participants in this study. Finally, further study could develop techniques to
automatically generate video demonstrations of all possible ways to construct a given model.
Chapter 4

SHiB ITS and teaching effectiveness

In this chapter, the author presents a study that examined the teaching effectiveness of three strategies (trial-and-error, textbook, and combination) via a smart-home installation intelligent tutoring system (SHiB ITS). One hundred and twenty-five participants were randomly assigned to one of the strategies and tested with the SHiB ITS. ANOVA analyses revealed that participants in the combination condition performed significantly ($p < 0.01$) better than those in the other two groups with medium effect-size. Senior participants in the trial-and-error group performed significantly ($p < 0.01$) better than those in the control condition with large effect-size. Younger participants in the combination condition performed significantly ($p < 0.05$) better than those in the control condition with medium effect-size. Results suggest that the teaching strategies had differential effects due to age groups.

Results, data analysis, and conclusion are included in this chapter. Section 4.1 states the purpose of this study, section 4.2 includes details of design the system, section 4.3 includes the experiment method, section 4.4 includes the experiment results and analysis, and section 4.5 includes the conclusion draws from the results.
4.1 Purpose of the study

In this chapter, we present an online intelligent tutoring system, SHiB ITS, to teach seniors how to install a smart-home system in their own house. This study aims to address the following questions:

1. What are the appropriate teaching strategies for seniors?

2. What are the effects of encouraging seniors to solve problems with multiple solutions?

3. How to extend the content of ITS for seniors to fields other than learning to use new technologies?

For the first question, we examined three teaching strategies: *exploration*, *textbook*, and *combination*. The *textbook* is the instructive teaching strategy, which is considered the control condition. *Exploration* is a trial-and-error teaching strategy; *combination* combines trial-and-error and instructive. *Exploration* and *Combination* are considered the experimental conditions. In the current study, we investigated which teaching strategy improves senior participants’ post-tests performance the most, and observed and compared the number of steps taken by senior participants to accomplish post-tests. We recruited young participants to have a general view of the effectiveness of teaching strategies that are employed by the SHiB ITS.

For the second question, we examined senior participants’ behaviors under the training exercises and their performance under the post-tests that require multiple solutions. We recruited young participants to investigate if there exist significant differences between the performances of age groups. We also experimented with the effects of answer verification during the training stage to examine if immediate feedback will help improve the post-test performance.

For the third question, we evaluate the design of SHiB ITS by a post-survey that is answered by each participant after the post-test’s end. The survey asks participants questions
about aspects of ease of use, functional complexity, and learning in solving problems in multiple ways.

Our study used an online learning environment, which means participants have to take the experiment on their own without any help from the experiment conductor. All the participants were recruited from an online crowd-sourcing platform (Amazon’s Mechanical Turk) and included different nationalities.

4.2 SHiB Intelligent Tutoring System

Gerontechnology is an interdisciplinary field combining gerontology and ubiquitous computing, aiming to allow senior participants to live alone longer and to have an improved quality of life. It often focuses on matching a living environment to the health, housing, mobility, communication, leisure, and work needs of seniors (Fernández-Caballero et al., 2017). Smart-Home in a Box (SHiB) is a smart home project that aims to recognize seniors’ activities and compare their present activities to their past regular routines. SHiB is a kit containing motion-aware sensors, temperature-aware sensors, relays, and an Ethernet server (Cook et al., 2013). The long-term purpose of the SHiB project is to produce a smart home kit that can be easily self-installed and then use it to provide a large amount of data related to user activity.

The SHiB ITS\textsuperscript{1} is a web ITS. It was designed to teach seniors how to install devices shipped in a kit into different types of dwellings, and then evaluate a participant’s ability to do so. Based on the SHiB kit installation instructions, there are requirements for selecting the position for a device. As the house layouts are different, it is impractical to give explicit instructions for all houses. Positions of devices in the SHiB ITS are relative. That is, there is no fixed positioning plan for devices. SHiB ITS judges submitted answers by reasoning the devices’ locations with the SHiB kit installation instructions, and then, if in the training

\textsuperscript{1}http://shib-test.byethost5.com/pre-tests/layout_tests/layout.php?MID=publictry& GROUP=3
stage, the ITS gives feedback in respect to the answer. SHiB ITS employed multiple-solutions teaching method in training and requires multiple solutions in post-tests sections. At the tasks requiring multiple solutions, SHiB ITS reasons the submitted answers as well as to recognize if they are distinct or not.

In Figure 4.1, we present a user interface example. The SHiB ITS adopts recommendations discussed in Wolfson et al. (2014). Participants can spend as much time as they want to complete the task. The difficulty level of tasks is raised gradually to avoid unnecessary confusion. Each task is composed of multiple small tasks, and at the beginning of each task, the ITS informs participants of how many devices they will place for the current task. Placement requirements for each of the device types are available to participants. The interface of the system consists of three parts: a function button area, a layout area, and an instruction area (see Figure 4.1). The three parts are designed to provide sufficient information without showing unnecessary information. The different areas of the screen are self-describing. The SHiB ITS runs in a browser window, which we developed in Javascript from scratch. While this allows us to more efficiently collect data from many participants, relative to using in-person studies, we have less control over the size of the application window and the size of the user’s display. Figure 4.2 shows the settings for the three training, Figure 4.3 shows post-tests settings, and Figure 4.4 shows an example of the exploration teaching strategy. Figure 4.4 illustrates how the hint changes according to participants’ selections. Figure 4.5 illustrates multiple solutions in the textbook teaching strategy.

4.3 Teaching effectiveness experiment

4.3.1 Participants

We recruited one hundred and sixty-three participants on Amazon’s Mechanical Turk (MTurk). Seventy-five participants are fifty and older, and eighty-eight are younger than fifty. As 50
years old is the threshold that population age studies (FBIC Global Retail and Technology, 2015) applied to describe the older adults. Moreover, previous older adults ITSs (Bruder et al., 2014; Ribeiro and de Barros, 2014; Toyota et al., 2014) used different age thresholds, start from 54 through 61. On purpose to recruit a bigger population, in this study 50 years old is chosen as the threshold.

The senior participants’ ages range from 50 to 70 ($mean = 56.84$). The sample includes 47 females and 28 males. The young participants’ ages range from 18 to 48 ($mean = 31.20$). The sample includes 42 females and 46 males. Of all the participants, 80.1% are from North
Figure 4.3 Settings for (a) post-test 1, (b) post-tests 2 and 3, and (c) post-test 4

Figure 4.4 Exploration strategy in training-exercise 2

America, 10.4% are from Asia, 4.3% are from South America, and 3.7% are from Europe, and 1.5% are from Africa. The majority of the participants (84.7%) are native speakers of English.

The tasks posted on MTurk are known as a Human Intelligence Tasks (HITs). In our setting, a participant was limited to executing a HIT once. We recorded each participant’s MTurk ID to prevent them from executing the task more than once. We set age constraints for the experiments: the age limit for the senior group is 50 to 120, and 18 to 49 is the limit for the young group. MTurk workers who agreed to participate in a HIT linked to a website asked to enter their age and gender before the experiment began. If the age of a worker were unacceptable, the website would not allow the worker to take the experiment. Participants
were unacquainted with the experimental condition of a HIT. The HIT was not specified to recruit any individuals in advance. Participants were (anonymously) compensated after the experiment. Those who finished the experiment earned $2. If they achieved over 90% accuracy during the post-tests, they received an extra $5 bonus (for a total $7). Participants could stop before completing the experiment (any participant could close their web browser window at any time and quit).

4.3.2 Measurement method

Participants’ background

Backgrounds of participants were assessed prior to beginning the experiment. Participants were asked to respond to a survey which included questions about their age, gender, marital status, education level, native language, geographical region, race, and computer experience.

Participants’ performance

We measured participants’ performance by the number of devices placed in the correct positions over the total number of devices given in a post-test. The SHiB ITS defines correct positions are those meet the SHiB kit installation principles. The SHiB ITS records participants’ steps at each task as logs that are used to compare the step length of each
participant representing the time they spent on each task.

**User experience evaluation**

We collected user experience at the end of the experiments on the purpose of answering the question that is it possible the content of ITSs for seniors extends to other areas. A post-survey was displayed and asked about their viewpoints on using the ITS. The questionnaire has twelve questions. Each question can be answering by choosing a value from an array associated with the question, with a rating of 1 through 5, from strongly disagree (corresponds to 1) to strongly agree (corresponds to 5). The questionnaire includes questions about confidence in using the ITS, the complexity of the ITS, utility of the ITS, and how much they understand about the content taught by the ITS. The result section presents the details of the questionnaire and the evaluation scores.

### 4.3.3 Procedure

The SHiB ITS consists of three parts: pre-test, training, and post-test. Figure 4.6 shows an illustration of the procedure. In the pre-test, a participant needs to place three room-sensors in the places she thinks are correct. The ITS does not give prompts nor feedback in the pre-test. In the training session, the participant needs to have three training exercises. The first exercise is a simple task, which has three different types of device. In the second exercise, the participant needs to place ten devices of three types in a layout. In the third exercise, the setting is identical to the second exercise, but the participant should place the devices in different places from where she put them in the second exercise. The SHiB ITS decides whether a device is placed at a correct position in accordance with the SHiB kit installation principles. The purpose of the second and third exercises is to have participants practice resolving the same problem with multiple solutions. In the post-test session, the participant has four tasks to take. The first post-test is a simple task, identical to the pre-test, to
measure whether participants improved from the pre-test. The second and third post-tests were moderately tricky, in identical settings. The two post-tests have identical types of devices, the same number of devices, and the same layout. Participants in the third-test should place the same devices in alternative places, to examine whether they can solve the same problem in two ways. The fourth post-test is more complicated than the other tests; it has nineteen devices to place, and almost all rooms in the layout should have devices.

Figure 4.6 An illustration of the procedure

4.4 Results

We examine the results in three age groups, a combined age group, a senior group, and a young group. The purpose of having a combined age group is to have a general view.

4.4.1 Teaching strategy effectiveness analysis

We conducted an experiment to examine the teaching strategies’ effectiveness via participants’ performance in the four post-tests. In this experiment, forty-eight out of seventy-five senior participants finished the pre-test through post-test 4. Seventeen participants were in
the *exploration* group, sixteen were in the *combination* group, and fifteen were in the *textbook* group. In the young participant group, forty-eight out of sixty-one participants completed the tasks. Seventeen young participants were in the *exploration* group, sixteen were in the *combination* group, and fifteen were in the *textbook* group.

We conclude that a participant passed a test if she had placed all the devices in the correct positions. Table 1 shows the statistical results of the success rate of both young and senior groups. The results include the mean values of the success rates of the corresponding test and the standard deviations. The success rate of each post-test is the fraction of the number of correctly placed devices over the number of total devices. The symbol $std$ is the standard deviation and $v_s$ is the coefficient variance, which is defined as the ratio of the standard deviation to the mean and shows the extent of variability about the mean values of each group.

We performed Fisher's exact test for testing the significance of performance in the pre-test and each post-test among the two age groups taught by the three teaching strategies. Results of the Fisher's exact test showed that in combined age groups, participants' performance in post-test 1 improved significantly from the pre-test. The performance of the *exploration* group is significantly ($p < 0.05$) better than that of the *textbook* group in post-test 1 and post-test 2. The performance of the *combination* group is significantly ($p < 0.01$) better than that of the *textbook* group in post-test 2. The success rate of the *exploration* group is better than that of the *combination* group in post-test 1 but is not significant. The results indicate that, overall, after training, most participants can correctly place the sensors shown in the pre-test. *Exploration* and *combination* have significant effects on participants' performance in the post-test.

In the senior group, participants in the *exploration* group performed significantly ($p < 0.05$) better in post-test 1 and post-test 2 than those in the *textbook* group. They also performed better than those in the *combination* group in three post-tests (post-test 1, 2, and 4), but Fisher's exact test did not show results that are significant. In the young
group, participants in the \textit{combination} group performed significantly ($p < 0.05$) better than those in the \textit{textbook} group in post-test 2. The success rates show the \textit{combination} group in the other three post-tests (post-test 1, 3, and 4) were better than the \textit{textbook} group, but Fisher’s exact test did not show those were significant. Success rates of the \textit{exploration} group are higher than those in the \textit{textbook} group, but not significantly. Participants in the \textit{combination} group performed significantly ($p < 0.01$) better than those in the \textit{exploration} group in post-test 3. The results indicate that the \textit{exploration} teaching strategy has a significant effect on improving senior participants’ performance in post-tests; on the other hand, \textit{combination} teaching strategy has a significant effect on improving young participants’ performance in post-tests. Especially in the young group, the effect of the \textit{combination} teaching strategy on improving their performance in post-tests is significantly higher than that of the \textit{exploration} teaching strategy. Compared with participants in \textit{exploration} and \textit{combination}, fewer participants in \textit{textbook} had improved performance in post-tests.

An ANOVA was performed to analyze the differences in the participants’ performances according to the three strategies. Instead of using success or failure we normalized each participant’s performance over the four post-tests. In each post-test, if a participant correctly placed all devices in that test, she received a +25%. If a participant passed all the post-tests, then she received 100% in total. If she passed three post-tests, she received 75% in total.

### Table 4.1 Participant performance in pre-test and post-tests

<table>
<thead>
<tr>
<th></th>
<th>pre-test</th>
<th>post-test 1</th>
<th>post-test 2</th>
<th>post-test 3</th>
<th>post-test 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$N$ mean std</td>
<td>$N$ mean std</td>
<td>$N$ mean std</td>
<td>$N$ mean std</td>
<td>$N$ mean std</td>
</tr>
<tr>
<td>combined age group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>explore</td>
<td>34 0.324 0.278 46.0</td>
<td>34 0.990 0.0572 7.77</td>
<td>34 0.959 0.0957 9.98</td>
<td>34 0.777 0.298 38.8</td>
<td>34 0.947 0.0777 8.21</td>
</tr>
<tr>
<td>textbook</td>
<td>30 0.344 0.270 78.3</td>
<td>30 0.969 0.199 22.1</td>
<td>30 0.870 0.188 21.6</td>
<td>30 0.818 0.247 30.5</td>
<td>30 0.895 0.203 22.7</td>
</tr>
<tr>
<td>combine</td>
<td>32 0.385 0.269 69.9</td>
<td>32 0.969 0.130 13.4</td>
<td>32 0.969 0.128 13.2</td>
<td>32 0.844 0.250 20.6</td>
<td>32 0.938 0.129 13.8</td>
</tr>
<tr>
<td>≥ 50 group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>explore</td>
<td>17 0.431 0.283 65.6</td>
<td>17 1.000 0.000 0.00</td>
<td>17 0.965 0.106 11.1</td>
<td>17 0.771 0.297 38.6</td>
<td>17 0.944 0.0882 9.34</td>
</tr>
<tr>
<td>textbook</td>
<td>15 0.409 0.287 71.8</td>
<td>15 0.869 0.206 23.2</td>
<td>15 0.860 0.203 23.8</td>
<td>15 0.749 0.277 37.5</td>
<td>15 0.849 0.221 26.1</td>
</tr>
<tr>
<td>combine</td>
<td>16 0.306 0.304 76.7</td>
<td>16 0.938 0.184 19.5</td>
<td>16 0.950 0.175 18.4</td>
<td>16 0.784 0.276 35.4</td>
<td>16 0.924 0.161 9.04</td>
</tr>
<tr>
<td>&lt; 50 group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>explore</td>
<td>17 0.216 0.234 109</td>
<td>17 0.980 0.0868 8.25</td>
<td>17 0.953 0.0674 9.18</td>
<td>17 0.782 0.367 39.2</td>
<td>17 0.951 0.0983 7.19</td>
</tr>
<tr>
<td>textbook</td>
<td>15 0.299 0.248 85.8</td>
<td>15 0.911 0.198 21.7</td>
<td>15 0.896 0.178 20.2</td>
<td>15 0.880 0.197 22.8</td>
<td>15 0.940 0.178 18.9</td>
</tr>
<tr>
<td>combine</td>
<td>16 0.375 0.240 61.9</td>
<td>16 1.000 0.000 0.00</td>
<td>16 0.988 0.0509 5.06</td>
<td>16 0.900 0.211 23.3</td>
<td>16 0.951 0.0991 9.58</td>
</tr>
</tbody>
</table>

An ANOVA was performed to analyze the differences in the participants’ performances according to the three strategies. Instead of using success or failure we normalized each participant’s performance over the four post-tests. In each post-test, if a participant correctly placed all devices in that test, she received a +25%. If a participant passed all the post-tests, then she received 100% in total. If she passed three post-tests, she received 75% in total.
We evaluated the degree of effectiveness of each teaching strategy as effect-size by Hedges’ $g$. Table 4.2 reports the ANOVA results.

In the combined age group, participants in the *exploration* group performed significantly ($p < 0.05$) better than those in the *textbook* group, with small to medium effect-size ($g = 0.438$). Participants in the *combination* group performed significantly ($p < 0.01$) better than those in the *textbook* group, with a medium effect-size ($g = 0.704$). The performance comparison between *exploration* and *combination* group was not significant in an ANOVA result.

In the senior group, participants in the *exploration* group performed significantly ($p < 0.01$) better than those in the *textbook* group, with a large effect-size ($g = 0.892$). Participants in the *combination* group performed significantly ($p < 0.05$) better than those in the *textbook* group. Performance comparisons between the *exploration* group and the *combination* group were not significant.

In the young group, participants in the *combination* group performed significantly ($p < 0.05$) better than those in the *textbook* group, with a medium effect-size ($g = 0.701$). Participants in the *combination* group performed significantly better than those in the *exploration* group, with a medium effect-size ($g = 0.798$). Performance comparisons between the *exploration* group and *textbook* group were not significant. Fisher’s exact test and an ANOVA test performed a performance comparison of the seniors. Both test results did not show significant differences.

We used an ANOVA test to analyze the difference in gender performance in each teaching strategy group was also analyzed by an ANOVA test. The results show that the normalized performance of each participant in the *textbook* group in post-test 2 and post-test 4 was significantly different ($p < 0.05$). The mean value of the normalized performance of female participants in post-test 2 is 93.3% with a variance of 0.011, whereas the mean value of male participants is 77.5% with a variance of 0.060. In post-test 4, the mean value of the normalized performance of female participants is 96.5% with a variance of 0.005, and
Table 4.2 Results of ANOVA test and Effect Size for participant performances in the four-posts

<table>
<thead>
<tr>
<th>Teaching Strategies</th>
<th>Exploration vs. Textbook</th>
<th>Combination vs. Textbook</th>
<th>Exploration vs. Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>combined age group</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>*</td>
<td>**</td>
<td>-</td>
</tr>
<tr>
<td>ES(g)</td>
<td>0.438 (small)</td>
<td>0.704 (medium)</td>
<td>-0.318 (small)</td>
</tr>
<tr>
<td>95% CI lower</td>
<td>-0.0767</td>
<td>0.171</td>
<td>-0.821</td>
</tr>
<tr>
<td>95% CI upper</td>
<td>0.953</td>
<td>1.24</td>
<td>0.185</td>
</tr>
<tr>
<td>≥ 50 group</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>**</td>
<td>*</td>
<td>-</td>
</tr>
<tr>
<td>ES(g)</td>
<td>0.892 (large)</td>
<td>0.716 (medium)</td>
<td>0.121 (negligible)</td>
</tr>
<tr>
<td>95% CI lower</td>
<td>0.107</td>
<td>-0.8693</td>
<td>-0.613</td>
</tr>
<tr>
<td>95% CI upper</td>
<td>1.68</td>
<td>1.50</td>
<td>0.854</td>
</tr>
<tr>
<td>&lt; 50 group</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>-</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>ES(g)</td>
<td>0.0272 (negligible)</td>
<td>0.701 (medium)</td>
<td>-0.798 (medium)</td>
</tr>
<tr>
<td>95% CI lower</td>
<td>-0.720</td>
<td>-0.8838</td>
<td>-1.56</td>
</tr>
<tr>
<td>95% CI upper</td>
<td>0.774</td>
<td>1.49</td>
<td>-0.0345</td>
</tr>
</tbody>
</table>

* p < 0.05 , ** p < 0.01
- no significant difference

those values of male participants are 78.9% and 0.081, respectively. In both situations, the female participants showed better performance than the male participants, which suggests that providing worked examples in training would have a larger positive impact on female participants than on male participants.

4.4.2 Participants’ step length analysis

Recording the time taken in an experiment can be difficult when the participant is not physically present. As a proxy for time, effort, and experiment difficulty, we recorded the number of actions each participant executed in each of the four post-tests. As the tasks of SHiB ITS continue to the next one right after a participant placed all the devices correctly, the number of steps a participant spent in a post-test indicate how many moves the participant had tried to find the correct positions. Participants can force the SHiB ITS continue to the next task if he or she does not want to fix mistakes. The mean value of the number of actions
(which we refer to as the step length) and the standard deviations are shown in Table 4.3. In general, an increased competency correlates with fewer steps when finding a correct solution (Rafferty et al., 2016). We examined the impact of the three teaching strategies on the participants’ step length using an ANOVA test. The data in Table 4.3 show the statistical results of participants who passed each post-test. \( \nu_s \) is coefficient variance of step lengths in the data.

Table 4.3 Results of participants’ performance in the four post-tests, including the number (N), the mean step length, and the standard deviation of the step length

<table>
<thead>
<tr>
<th></th>
<th>post-test 1</th>
<th>post-test 2</th>
<th>post-test 3</th>
<th>post-test 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>mean</td>
<td>std</td>
<td>( \nu_s )</td>
</tr>
<tr>
<td>combined age group</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>exploration</td>
<td>32</td>
<td>177</td>
<td>29.1</td>
<td>0.165</td>
</tr>
<tr>
<td>textbook</td>
<td>23</td>
<td>298</td>
<td>188</td>
<td>0.631</td>
</tr>
<tr>
<td>combination</td>
<td>30</td>
<td>207</td>
<td>54.9</td>
<td>0.265</td>
</tr>
<tr>
<td>( \geq 50 ) group</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>exploration</td>
<td>17</td>
<td>174</td>
<td>32.8</td>
<td>0.188</td>
</tr>
<tr>
<td>textbook</td>
<td>11</td>
<td>253</td>
<td>142</td>
<td>0.562</td>
</tr>
<tr>
<td>combination</td>
<td>14</td>
<td>201</td>
<td>43.3</td>
<td>0.216</td>
</tr>
<tr>
<td>( &lt; 50 ) group</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>exploration</td>
<td>15</td>
<td>180</td>
<td>24.9</td>
<td>0.139</td>
</tr>
<tr>
<td>textbook</td>
<td>12</td>
<td>340</td>
<td>220</td>
<td>0.649</td>
</tr>
<tr>
<td>combination</td>
<td>16</td>
<td>213</td>
<td>64.8</td>
<td>0.304</td>
</tr>
</tbody>
</table>

Results of the ANOVA test show in the combined age group, participants in the exploration group took significantly \( (p < 0.001) \) fewer steps than those in the the textbook group in post-test 1; participants in the combination group took significantly \( (p < 0.05) \) fewer steps than those in the the textbook group in post-test 1; participants in the exploration group took significantly \( (p < 0.01) \) fewer steps than those in the combination group in post-test 1. In the senior group, participants took significantly \( (p < 0.01) \) fewer steps than those in the textbook group in post-test 1 and post-test 2. Participants in the exploration group took significantly \( (p < 0.05) \) fewer steps than those in the combination group in post-test 1. No significant difference was shown in the comparison between the combination and textbook groups. In the young group, in post-test 1, participants in the exploration group took sig-
nificantly \((p < 0.01)\) fewer steps than those in the *textbook* group, and significant \((p < 0.05)\) fewer steps than those in the *combination* group. Participants in the *combination* group took significantly \((p < 0.05)\) fewer steps than those in the *textbook* group in post-test 1. As shown in Table 4.3, the mean values of step length in post-test 3 of the three teaching strategies increase significantly from post-test 2, which indicates that the objective difficulty of post-test 3 is higher than that of post-test 2. Step length increased in post-test 4 from post-test 3, suggesting that post-test 4 has higher complexity than post-test 3, which is consistent with the difficulty design of the tutoring system.

### 4.4.3 Effects of training validation analysis of Experiment II

To examine the effects of applying validation processes in training for *textbook* and *combination* teaching strategies, we conducted another experiment to compare the four post-tests' performance among five teaching strategies. The first three teaching strategy groups were those in Experiment I, *exploration, textbook*, and *combination*. In this experiment, we added two teaching strategy groups to the existing ones. They are *textbook II* and *combination II*. In the *textbook II* group, participants were given the same guidance as the regular *textbook* group. In each training exercise, the system examined their answers and required them to make corrections if mistakes were found. In the *combination II* group, as in the regular *combination* group, participants had three training exercises with guidance and feedback given in the third exercise, and the system skipped the answer validation process in the first two exercises.

Twenty-nine participants younger than fifty took part in the experiment with two new groups. The *combination II* group has fourteen participants and the *textbook II* group has fifteen participants. Since there was no significant difference in performance on the post-tests between senior and young participants, in this experiment we compared performance of young participants in the five teaching strategy groups. Data of the first three teaching
strategy groups were collected from young participants in the last experiment. Statistical results from the five groups’ performance in the four post-tests are shown in Table 4.4. Mean values of the success rates of the four post-tests of each group and their standard deviations are shown in the table. \( v_s \) is the coefficient variance.

<table>
<thead>
<tr>
<th>Group</th>
<th>post-test 1</th>
<th>post-test 2</th>
<th>post-test 3</th>
<th>post-test 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>mean</td>
<td>std</td>
<td>( v_s )</td>
</tr>
<tr>
<td>Exploration</td>
<td>17</td>
<td>0.980</td>
<td>0.0808</td>
<td>8.25</td>
</tr>
<tr>
<td>Textbook 1</td>
<td>15</td>
<td>0.911</td>
<td>0.198</td>
<td>21.7</td>
</tr>
<tr>
<td>Combination 1</td>
<td>16</td>
<td>1.00</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Textbook 2</td>
<td>15</td>
<td>0.956</td>
<td>0.172</td>
<td>18.0</td>
</tr>
<tr>
<td>Combination 2</td>
<td>14</td>
<td>1.00</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Fisher’s exact tests were performed on the collected data for measuring the significance of participants’ performance. Participants in the combination group performed significantly \( (p < 0.05) \) better in post-test 2 and post-test 3 than those in the textbook II. Participants in the combination II group performed significantly \( (p < 0.05) \) better in post-test 4 than those in exploration group.

ANOVA tests were performed to analyze the differences in the accuracy rates (measured as percentage of sensors correctly placed) in each post-test. The results showed participants in the textbook group performed significantly \( (p < 0.001) \) better in post-test 3 than those in the textbook II group. Textbook had a mean value of 88.0%, whereas the value for the textbook II group was 58.7%. Participants in the combination II group performed significantly \( (p < 0.05) \) better in post-test 4 than those in the exploration group.

4.4.4 Results of post-survey

The questionnaire asked questions in five aspects: participants’ likeliness of using the tutorial system, the ease use of the system, the complexity of the functional design of the tutorial system, and the likeliness of learning by multiple solutions. To answer the question can the
contents of ITSs for seniors extend to other fields, we will present the results of questions regarding their likeliness of using the tutorial system and the ease use of the system. There are five questions regarding these aspects. Question 1 is about the likeliness of using the tutorial system. Questions 3, 4, 7, 8, 9, and 10 are about the ease use of the system. Table 4.5 shows the results of these questions.

The purpose of presenting the results of the combined group is to have a view of participants’ responses in general. The score distribution in questions 3, 4, 7, 8, 9, and 10 show the participants, in general, do not need extra assistance or knowledge to use the tutorial (items 4 and 10). Scores for question 3 and 8 show participants have a balanced view on the ease of using the tutorial. From their comments, the researchers found many participants thought of using the mouse to control the avatar rather than the keyboard (arrow keys, “p” to pick, and “o” to place) to control the avatar. The score of Question 1 shows the participants, in general, have an unbiased opinion of the likeliness of using the tutorial system to learn to place the sensors. Senior participants in the textbook group has the highest mean score for the question, which implies senior participants like the practicing worked examples step-by-step more than the trial-and-error.

Table 4.5 Results of post-survey

<table>
<thead>
<tr>
<th>Question#</th>
<th>Question content</th>
<th>Combined Age</th>
<th>≥ 50</th>
<th>&lt; 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Do you agree with the following statement: ‘I think that I would like to use this tutorial to learn how to place sensors.’</td>
<td>3.49</td>
<td>3.45</td>
<td>3.52</td>
</tr>
<tr>
<td>3</td>
<td>Do you agree with the following statement: ‘I thought the tutorial was easy to use.’</td>
<td>3.13</td>
<td>2.96</td>
<td>3.29</td>
</tr>
<tr>
<td>4</td>
<td>Do you agree with the following statement: ‘I think that I would need assistance to be able to use this tutorial.’</td>
<td>2.52</td>
<td>2.55</td>
<td>2.48</td>
</tr>
<tr>
<td>7</td>
<td>Do you agree with the following statement: ‘I would imagine that most people would learn to use this tutorial very quickly.’</td>
<td>3.20</td>
<td>3.09</td>
<td>3.32</td>
</tr>
<tr>
<td>8</td>
<td>Do you agree with the following statement: ‘I found the tutorial very cumbersome to use.’</td>
<td>3.15</td>
<td>3.18</td>
<td>3.13</td>
</tr>
<tr>
<td>9</td>
<td>Do you agree with the following statement: ‘I felt very confident using this tutorial.’</td>
<td>3.33</td>
<td>3.23</td>
<td>3.44</td>
</tr>
<tr>
<td>10</td>
<td>Do you agree with the following statement: ‘I needed to learn a lot of things before I could get going with this tutorial.’</td>
<td>2.69</td>
<td>2.69</td>
<td>2.70</td>
</tr>
</tbody>
</table>
4.5 Discussion and Conclusion

This chapter presents an Intelligent Tutoring System (ITS) that is designed for seniors to teach them the principles of installing a smart-home kit (SHiB). The chapter aims to address three research questions: 1) What are the appropriate teaching strategies for seniors? 2) What are the effects of encouraging seniors to solve problems with multiple solutions? 3) Can the content of ITS for seniors extend to a new field?

For the first research question, we examined the teaching effectiveness of three teaching strategies, exploration, textbook, and combination, via the SHiB ITS, with senior (fifty years or older) and younger adults (younger than fifty years). For the second problem, learners are required to solve a problem with two solutions in every teaching strategy condition. For the third problem, we designed a post-survey to collect participants’ opinions on five aspects regarding the ITS. The survey asks questions about the likeliness of using the ITS as well as the ease of use it.

The ITS was designed to teach seniors to learn the installation principles of a smart-home kit. We designed the user interface of the ITS with the consideration for the declining cognitive capacity and sensory capacity of seniors. The learning program consists of a pre-test, three training exercises, and four post-tests. On average, adults completed the study in 45 minutes.

In Experiment I, the effectiveness of the three teaching strategies is investigated and assessed by participants’ performances in four post-tests. The results indicate that the learning outcome of a participant is affected by teaching strategy regarding age. Participants who are fifty years or older gained the most benefits from the exploration (trial-and-error) teaching strategy, and those in the combination (a combination of trial-and-error and practicing worked examples) did better than those in the textbook (practicing worked examples). Younger participants learned best from the combination teaching strategy. The result is in line with the previous chapter, which shows that college students benefit the most from a
teaching strategy that combines practicing worked examples and trial-and-error, and they can apply the knowledge to solve new problems. Our results show that younger participants who were trained by the combination teaching strategy performed significantly better than those trained by exploration and textbook. This finding is in line with other research (Rittle-Johnson et al., 2009) that shows having elementary students or college students learn worked examples step-by-step can maximize the teaching effectiveness.

In both age groups, the performance of the textbook group perform worse than the other two groups. Chi and Wylie (2014) gives a precise definition of passive learning and active learning. Participants in the textbook group were practicing worked examples of correctly placed devices, and such learning process is a type of passive learning; while participants in the exploration group and combination group were exploring solutions on themselves, which is a type of active learning. Based on the study (Chi and Wylie, 2014) the post-test performance of participants in the active learning condition is better than those in the passive learning condition. The cognitive outcome of passive learning is to recall the knowledge to the same context, while participants in the active learning condition are to apply the knowledge to a similar but not identical context. The apply knowledge cognitive activity leads to a deeper understanding than the recall activity. Therefore, a reason behind the performance of the textbook group is the teaching strategy leads to passive learning for the participants regardless of age while trial-and-error teaching strategy leads to active learning.

The result of the senior group suggests having senior adults explore solutions and compare two solutions for one problem during the training section will positively affect their performance in post-tests. Although senior adults, in general, are more comfortable in learning by following instructions step-by-step than by trial-and-error (Leung et al., 2012). The latter one, based on our experimental results, will be more helpful for them in learning knowledge and then apply it to solve problems. The post-survey results show seniors like to learn by exploring and comparing multiple solutions. A study (Rittle-Johnson et al., 2009) mentioned
that prior knowledge in a domain is essential for a novice to benefit from comparing solutions. The study shows that comparing solutions is more useful for middle school or college students who have prior knowledge about algebra than those who do not. However, when the new learners are seniors, knowledge gained via trial-and-error in solving a problem with two solutions is more effective than directly practicing the two solutions.

The step length of a participant who passed a post-test reflects the extent to which the participant understands the contents taught by the ITS. An assumption of using step length to examine the degree of understanding is that a well-trained participant will take fewer steps to pass a post-test than those who are not. As the SHiB ITS tasks continue to the next one once a participant placed all the devices correctly. A long step log indicates that there exists confusion or the participant may make repeated errors, and a short step log, in combination with the task accuracy rate, shows the participant understands the problem. The results of the comparison of the step length of the three teaching strategies show that post-test 3 is very challenging for both senior and younger participants. However, how to best identify participant confusion or the reasons behind their repeated errors, and when to intervene to provide advice is a crucial problem for future research. The mean accuracy rate of senior participants in post-test 3 is 76%, whereas the accuracy of younger participants is 85%. Results of participants' performance in post-test 2 show the mean accuracy rate of the senior participants is 92%, and that for younger participants is 94%. Since post-test 2 and post-test 3 are in identical settings, and post-test 3 asks for a solution that is different from what the participant has submitted in post-test 2, the decreased accuracy rate in post-test 3 shows that post-test 3 is more difficult for seniors than for younger adults. The difference could be caused by the declining capacity of senior participants' working memory. It would be an interesting study in the future to test if and how post-test 3 overloaded senior participants' working memory.

In Experiment II, the effectiveness of answer validation on participants' performance in the post-tests is investigated. The results indicate that answer validation has a negative
impact on participants' post-test 1 performance in the textbook teaching strategy. Participants trained by practicing worked examples with answer checking at each exercise performed significantly worse than other groups in post-test 2 and post-test 4. The results indicate that answer validation is unnecessary when learners are learning by practicing worked examples, but it is necessary when learners are learning by trial-and-error teaching strategy. Future studies could investigate the reason behind the observation. One reason could be that answer validation misleads participants to focus on whether they follow the example perfectly rather than on gaining the knowledge by practicing worked examples. It also indicates that when learners practice a worked solution, it is unnecessary to check their answers.

The post-survey results answer the research question of the possibility of extending the contents taught by ITSs for seniors to other fields. The results suggest that ITS designers can extend the teaching contents to novel fields for seniors as long as the user interface is self-explained, has no time constraint in learning, and has necessary written instructions. The results also give clues to the ITS design. User inputs could be simplified; the SHiB ITS requires user input from the keyboard, which could upgrade to using the mouse to move the devices.

There are multiple interesting directions for future work. First, future work should investigate a robust approach to identify when the participant is confused and to assist at the most advantageous time. Second, data mining technology may allow us to analyze how the actions performed by the participant during the training section could predict his or her performance in the post-tests. Third, we will study how well participants are trained in the ITS and then install an actual smart-home kit in an apartment. Fourth, it is possible that our ITS could learn from expert examples, allowing it to better provide physical, realistic training. Fifth, cognitive modeling could be used to better understand a participant’s learning ability.
Chapter 5

SHiB ITS in actual smart-home kit installation

A fact about baby boomers is that, by 2035, one in five people in the United States will be aged 65 and older (Joint Center for Housing Studies of Harvard University, 2014). The Smart-Home in a Box (SHiB) (Cook et al., 2013) is a ubiquitous system that intends to improve life quality of older adults. The SHiB requires a self-installation before in use. Our previous utility study (Hu et al., 2016) of SHiB found that it is not easy for seniors to install correctly. The SHiB Intelligent Tutoring System (SHiB ITS), which is introduced in Chapter 5, is designed to help individuals install a SHiB kit. The ITS uses a house layout with different types of SHiB devices as training exercises, and different house layouts in four post-tests to provide a range of examples. Ideally, the ITS would allow individuals to understand the SHiB installation principles in a more intuitive way.

In this chapter, the author will presents an experiment and its results of comparing the effectiveness of prior knowledge gained by using the SHiB ITS versus using a static document when participants install a SHiB kit. Before start the experiment, the author developed three hypotheses:

1. Participants’ performance of the real installation in the experimental group will be
better than those in the control group;

2. Participants in the experimental group will spend less time in the real installation than those in the control group;

3. Participants in the experimental group will be less confused about the installation than those in the control group.

The setting of the experiment is described in section 5.1, results of the experiment and relevant discussions are in section 5.2, conclusions and suggestions for the future work that extends from the current study are in section 5.3.

5.1 Methods

Knowledge about the SHiB installation can come from multiple types of sources: an observation of a demonstration, an oral description of the procedure, training in a virtual environment, or a brief introduction booklet. In this study, regardless of participants’ previous experience, two ways of teaching are compared. One method is through the tutoring system, SHiB ITS, which has three training exercises and four post-tests. The other method is a booklet containing three examples of placing sensors in a layout. Participants in both groups were given installation instructions, which has been included in the SHiB kit for two years and used by over eighty participants. The examples in the booklets are identical to the three training exercises in the SHiB ITS for the sake of having participants in the two groups have the same information. The installation of the SHiB kit is in an empty on-campus apartment and the floor plan of the apartment is unknown to participants before they begin the physical installation.
5.1.1 Participants

Participants were enrolled from the Pullman, Washington area and students from Washington State University. Twenty participants took part in the study. 12 of them were males and 8 of them were females. The age of participants ranged from 18 to 55 years old, mean value is 29.40 and the standard deviation is 11.06. 8 participants were married, 12 participants were single. 10 participants were Caucasian, 4 were Asian, 2 were Native American, 1 was African American, and three did not answer this question. 12 participants spoke English as their native language. Native languages spoken by the other 8 participants include Chinese, Hindi, Spanish, and others. The lowest level of education was completion of high school and the highest education level obtained was Ph.D. Each participant is identified by as a two digit number. Participants voluntarily joined in this study. Participants were assigned randomly to two groups; each group has 10 participants.

5.1.2 Procedure

The experiment was conducted in an apartment with three bedrooms, one bathroom, one living room, and one kitchen, for a total of 1106 square feet. In the experiment, participants needed to install sensors in two bedrooms. Before beginning, each participant is surveyed about their background, including age, gender, native language, and so on. In the training part of the experiment, participants in the control group were given a booklet with three examples of a correct smart home sensor arrangement on a floorplan layout. An example of the layout is shown in Figure 5.1a. Details of the training contents of the control group is shown in section 5.1.3.

Participants in the experimental group trained on the SHiB ITS on a provided laptop. An example of the interface of the SHiB ITS is shown in Figure 5.1b. The SHiB ITS has a pre-test, which helps participants become familiar with the interface. There are three exercises in the training section and four tests in the post-test section. Details of the training contents
(a) An example of a correct sensor placement for an apartment layout

(b) An example of SHiB ITS user interface

**Figure 5.1** Example of training contents for experimental group and control group
of the experimental group is stated in section 5.1.3.

During the physical installation part of the experiment, each participant was given a box containing 25 smart home devices. The devices include 14 motion-sensors, 2 relays, 2 temperature sensors, 1 door-sensor, and 1 server. Among those devices, the temperature sensor and the door sensors were not included in the training section. Besides the devices, each participant was given a handbook having the details for the installation of each type of sensor. For instance, the motion-sensors should be placed on the ceiling or as high as possible in a room via adhesive strips. A specific port of the server should be connected to the Ethernet cable. Relays should be placed on walls via adhesive strips near power plugs. The door sensor has two magnetic components: one should be placed on the door frame and the other should be on the door. The two door sensor components should be aligned and close together when the door is closed. The handbook contains general information about the placement of each type of sensor, which becomes a reminder for participants in the actual installations.

5.1.3 Training Contents

The teaching contents of the control group are printed in a three-page booklet. The format of each page is same as Figure 5.1a. The contents of the experimental group include training exercises and post-tests. The contents of the training exercises of the experimental group are similar to that of the control group, but they are delivered differently. In the first and second training exercises in the ITS, participants needed to follow a worked example. If an error was made, the ITS will ask the participant to fix the error. In the third training exercise, the content is the same as the third worked example in the booklet, but ITS participants were not initially provided the solution — they had to discover it on their own. The ITS gives hints to participants if there are errors.

In this section, section 5.1.3 discusses the three worked examples printed in the three-
page booklet. Section 5.1.3 discusses details of the ITS and covers the third training exercise and the four post-tests.

Control Group

Worked examples shown in the booklet are in the sequence of from simple to complex. The example on page 3 is in the same setting as the one on page 2. Page 3 shows an alternative solution for the same problem. The purpose of showing two solutions is to highlight alternate acceptable solutions. The three worked examples of the booklet are shown in Figure 5.2. It can be seen that the house layouts of the three examples are identical. In the example from page 1 (Figure 5.2a) there are three devices: a room sensor, a relay, and a server. On pages 2 and 3, (Figures 5.2b and 5.2c), there are three room sensors, three relays, three kitchen sensors, and a server. The example in Figure 5.2c is an alternative solution of the example in Figure 5.2b. For example, in Figure 5.2b, the three room sensors are placed in the left bedroom, while in Figure 5.2c, they are placed in the right bedroom. Positions of relays and kitchen sensors in Figure 5.2c are changed from that in Figure 5.2b.

![Figure 5.2](image-url)  
(a) page 1  
(b) page 2  
(c) page 3  

**Figure 5.2** Worked examples in the control group’s instruction booklet
Experimental Group

As the first and the second training exercises are similar to those in the booklet, this section starts at the third training exercise in the ITS, and the four post-tests. Four examples of the third training exercise are shown in Figures 5.3a–5.3d. When the ITS detects a device is at a wrong place, it shows an X on the device. The ITS has three teaching strategies. The strategy that was used in this study does not tolerate incorrect device placement at training stage. Every time a participant presses the “Submit” button, the ITS verifies the devices placement. The participant moves the red avatar and uses it to pick up a device that has been incorrectly placed and then click the “Advice” button at the bottom right. The ITS will then highlight all the areas that the device can be placed. As shown in Figure 5.3a, the name of the device is “Bedroom1_Sensor_3,” which is shown in red at the right. A bedroom sensor can be correctly placed in three types of spots: above the bed, at the door, or in the area in a bedroom such that no one can see it from outside the room. As in the third exercise, the location bedroom sensors are placed in should also be different from where the devices were placed in the second exercise. In this example, only two types of spots are highlighted in the left bedroom because the device was placed in the third type of spot in the second exercise, which is the area that no one can see from outside the bedroom.

A Kitchen sensor example is shown in Figure 5.3b. The name of the example kitchen sensor is “KitchenSensor_1,” which is in red on the right side of the layout. The example shows two places in the kitchen where the device can be placed: the spot in front of the sink and in the corner behind the doorway to the kitchen. A kitchen sensor can be placed in three spots in a kitchen — the requirements for the sensor are shown on the right side of the layout. In this case, the spot in front of the oven is not highlighted, which indicates that the device was placed in that spot in the second training exercise.

A relay placement example is shown in Figure 5.3c. Relays should be placed next to outlets. Five outlet symbols are in the layout and four of them are highlighted. The one in
(a) An example of hints for a room sensor

(b) An example of hints for a kitchen sensor

(c) An example of hints for a relay

(d) An example of hint for the server

Figure 5.3 An example of Training 3 exercise of the ITS
the kitchen is not highlighted because the relay, “relay_2,” had been placed near this outlet in the second training exercise.

A server placement example is shown in Figure 5.3d. The server should be placed near the internet connection, which is highlighted in the figure. The server has only one spot as there is normally a single router in a house.

The four post-tests use three different house layouts. Post-test 1 tests three room sensors. Post-test 2 and post-test 3 use the same layout and both test six sensors, three relays, and a server. Post-test 3 requires an alternative solution from the solution submitted in post-test 2. Post-test 4 has another layout and tests twelve sensors, three relays, and a server. Examples of the post-tests are shown in Figure 5.4. In post-tests, the ITS will verify submitted answers if the participant wants to check her answers. The ITS allows the user to continue even if sensors are incorrectly placed.

![Figure 5.4 Examples of post-tests](image)

5.1.4 Evaluation Metrics

Participants were evaluated in terms of how accurately they installed sensors in the physical apartment. The researchers marked the spots that the participant has chosen to install each device. A spot is marked in the format of “x-a”, where “x” is the location and the letter “a” will be changed to the initial letter of the sensor type. For instance, “R” is the initial letter for relays, “M” is the initial letter for motion sensor, “A” is the initial letter for area sensor,
and “T” is the initial letter for temperature sensor. An example is shown in Figure 5.5. In the figure, correctly placed devices are circled in green and incorrectly placed devices are circled in red. No experiment was conducted in the area where is covered by a big “X”.

Another evaluation metric is the time each participant spends in the training section and installation section of the experiment. We assume that having a better understanding of the installation requirements correlates with a shorter installation time. We believe that the installation time is more important than the time spent training — shorter installation time saves energy for senior residents and will be helpful in reducing the risk of indoor accidents. A decision change in the SHiB kit installation means the participant must re-install a device, which could include climbing up and down a ladder (the risk of injury due to from falling from ladders increases with age (Con et al., 2014)). In contrast, time spent in training has minimal risk. In addition, each participant was surveyed after the installation. In the questionnaire, their opinions about the installation were collected and will become another metric for the experiment. Surveyed questions are shown in the section 5.4.

Figure 5.5 An example of evaluating an installation
5.2 Results and Discussion

Results of the experiment consist of three parts: the participants’ performance in the physical installation, the time spent by participants in training and the physical installation, and the participants’ responses to the post-survey.

5.2.1 Performance of Installation

Results of the experiment are presented in this section. Results of the actual installation are shown in Table 5.1. Each device was installed 10 times in each group. Types of devices are shown in Table 5.1, and there are different numbers of each type of device. The table also records “failures,” which is the number of times that type of devices had been installed at an incorrect position. A server failed to be installed when it was not connected correctly to the Ethernet. The average accuracy rate represents the mean value of the number of correctly installed devices over the total number of devices in each group.

The results of the physical installation performance does not support hypothesis (1), which is that the experimental group will perform better than the control group. The experimental group performed better at installing motion sensors and area sensors than the control group, and the control group performed better in server installation than the experimental group. Installing temperature sensors, relays, and door sensors are similar in the two groups. A reason for high failure rate in installing the server in the experimental group is that participants in the experimental group were trained that the server should be connected to the Ethernet, but the training does not have instructions for how to connect to the Ethernet. The server has two ports, but only one of them works for the connection. Participants have to read the installation manual carefully to know which port to connect to. Temperature sensor installation has specific requirements. An oven temperature sensor should be placed right above the oven — many participants placed the sensor on the ceiling, which was too far from the oven to be effective. Similarly, the temperature sensor in the
bathroom was difficult to place — it had to be placed close enough to the shower so that it could detect hot water, but far enough that it did not become damp from steam.

Table 5.1 Results of real installation

<table>
<thead>
<tr>
<th></th>
<th>Motion</th>
<th>Area</th>
<th>Temperature</th>
<th>Relay</th>
<th>Door</th>
<th>Server</th>
<th>Average Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>total</td>
<td>failure</td>
<td>total</td>
<td>failure</td>
<td>total</td>
<td>failure</td>
<td>total</td>
</tr>
<tr>
<td>Experimental Accuracy</td>
<td>80.7%</td>
<td>80.0%</td>
<td>50.0%</td>
<td>90.0%</td>
<td>60.0%</td>
<td>78.0%</td>
<td></td>
</tr>
<tr>
<td>Control Accuracy</td>
<td>77.1%</td>
<td>76.0%</td>
<td>55.0%</td>
<td>95.0%</td>
<td>90.0%</td>
<td>77.6%</td>
<td></td>
</tr>
</tbody>
</table>

5.2.2 Time Spent

The researcher timed how long it took for each participant to finish each section of the experiment. Since the two training methods were designed to have different training times, the results reflect the purpose of the training methods. The mean time of control group participants spent in training is 6.4 minutes, while the mean time spent by the experimental group is 59.4 minutes. Shorter training time in the control group is due to the brevity of the 3-page booklet given to each participant. Each page has an intuitive layout with sensors marked at correct places. Participants can spend as much time as they want in the booklet. Most participants spent less than 10 minutes reading through the material. On the other hand, the SHiB ITS training program has eight tasks in total. Each task requires the participants to work on each sensor in a virtual environment. The program is designed to provide 45 to 60 minutes of training for learners.

Results of the time spent by each participant installing the SHiB and the responses of participants to the post-survey are shown in Table 5.2. An ANOVA test is conducted on the installation time. The result implies that participants in the experimental group spent significantly ($p < 0.05$) less time (with a mean of 64.2 minutes) than participants in the control group (with a mean of 80.6 minutes). The result supports hypothesis (2) of the
study. The result indicates that it is possible for participants to spend less time in installing the smart home kit in their own house if they had a training in the SHiB ITS program as opposed only following static directions.

**Table 5.2** Comparison of time spent in installation

<table>
<thead>
<tr>
<th>No.</th>
<th>(min)</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>73</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>56</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>84</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>58</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>75</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>11</td>
<td>67</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>105</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>13</td>
<td>40</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>108</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>N/A</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>108</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>N/A</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>67</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>67</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>21</td>
<td>75</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>22</td>
<td>67</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>23</td>
<td>58</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>40</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>25</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>26</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>27</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No.</th>
<th>(min)</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>73</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>56</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>84</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>58</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>75</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>11</td>
<td>67</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>105</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>13</td>
<td>40</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>108</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>N/A</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>108</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>N/A</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>67</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>67</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>21</td>
<td>75</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>22</td>
<td>67</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>23</td>
<td>58</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>24</td>
<td>40</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>25</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>26</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>27</td>
<td>105</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mean</th>
<th>64.2(*)</th>
<th>4.1(**)</th>
<th>4.4(**)</th>
<th>4.5(**)</th>
<th>4.3(**)</th>
<th>4.1(**)</th>
<th>4.0(**)</th>
<th>3.7</th>
<th>3.3</th>
<th>2.3</th>
<th>3.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Std.dev</td>
<td>40</td>
<td>0.9</td>
<td>0.5</td>
<td>0.7</td>
<td>0.5</td>
<td>0.5</td>
<td>0.6</td>
<td>0.5</td>
<td>1.1</td>
<td>0.8</td>
<td>0.9</td>
</tr>
<tr>
<td>Min</td>
<td>84</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Max</td>
<td>115</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

\[(*) = p < 0.05, (***) = p < 0.01\]

**5.2.3 Post-survey Results**

Numeric questions of the post-survey are classified into two categories: questions 1 through 6 relate to the ease of actual installation of smart-home devices, while questions 7 through 10 relate to the training and overall installation. The numeric value of each question represents the degree of agreement to the question, “5” stands for strongly agree or very easy while “1” stands for strongly disagree or very difficult, and “N/A” means that the participant did not answer that question. The analyses of the results do not include those responses with “N/A.” Questions are listed in section 5.4.

Results for the first six questions of the post-survey are shown in Table 5.2. An ANOVA test is conducted to examine the differences between the two groups. The two groups’ average rating values for the first six questions implies that the responses from the experimental
group are significantly more positive \((p < 0.01)\) than those from the control group. The six questions focus on participants’ subjective response to the actual installation. Questions 1, 3, and 6 are designed to collect participants’ viewpoints of the ease of the physical sensor and server installations. Questions 2 and 4 are designed to examine participants’ viewpoints on how easy the installation manual and the names of the sensors are to understand. Question 5 is designed to evaluate participants’ opinions about the ease of installation for other people. The responses to the first six questions from both groups suggest that participants believe the physical installation is easier for participants in the experimental group than those in the control group. This result implies that participants in the experimental group had higher confidence in the actual installation than those in the control group.

Results of questions 7 through 10 are shown in Table 5.2. An ANOVA test is conducted on the average values of questions 7 through 10 in the post-survey to examine the significance between the two groups. No significance is observed. Questions 7 and 10 evaluate how easy participants thought it was to understand the training material and how useful the training was for the physical installation. The results imply that training is easy (mean = 4.0) for participants in the control group, but neutral (mean = 3.7) for those in the experimental group. However, participants in the experimental group think the training is helpful for the installation (mean = 3.9) while the responses from participants in the control group are neutral (mean = 3.2). The length of the whole installation process is neutral for participants in both groups. Participants in both groups disagree with the statement that the installation was not frustrating.

Question 11 asks, “Would you recommend installing a Smart Home in a Box to others?” and was responded to by 18 out of 20 participants with “yes”; two participants did not answer this question. Question 12 asks “Did you receive any help with installing the sensors?”, is answered with 6 “No”s and 4 “Yes”s in the experimental group, and 8 “No”s and 2 “Yes”s in the control group. Participants could receive some types of help in the installation; for example, researchers answered questions about the apartment floor plan, or helped to hold
the ladder while the participant was placing devices in high places. Questions relating to the correct position for placing the devices are not included in the help, since the researchers refused to answer these.

Question 13 asks participants to comment on what part of the training seemed the most helpful. The responses are shown in Table 5.3. “N/A” means the participant did not answer the question. 10 participants in the experimental group answered the question, while 8 participants in the control group left responses. In both groups, participants mentioned that the training for the motion sensors and the area sensors was helpful. Participants in the experimental group mentioned that the training for the relays’ position was helpful, while one participant in the control group mentioned the material for relays was confusing. Practicing on the SHiB ITS was mentioned as helpful for participants in the experimental group. Participants in the experimental group also commented that the similarity between the SHiB ITS and the actual installation was helpful.

Table 5.3 Responses to Q13 of the experimental group and the control group

<table>
<thead>
<tr>
<th>Participant No.</th>
<th>Response on Q13</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>instructing every time and showing proper directions to set up the sensors</td>
</tr>
<tr>
<td>2</td>
<td>confused with relays</td>
</tr>
<tr>
<td>4</td>
<td>It help me understand where the ‘area’ sensor should be placed</td>
</tr>
<tr>
<td>3</td>
<td>like enforce best places for sensors</td>
</tr>
<tr>
<td>6</td>
<td>place sensors and relays in the specific area</td>
</tr>
<tr>
<td>5</td>
<td>textbook is helpful for area sensors</td>
</tr>
<tr>
<td>7</td>
<td>the similarity between the software and manual instructions</td>
</tr>
<tr>
<td>8</td>
<td>textbook gives a general understanding of different sensor locations</td>
</tr>
<tr>
<td>10</td>
<td>knowing the placement of the sensors</td>
</tr>
<tr>
<td>9</td>
<td>N/A</td>
</tr>
<tr>
<td>11</td>
<td>locations of sensors</td>
</tr>
<tr>
<td>12</td>
<td>Floor plans with pictures is okay</td>
</tr>
<tr>
<td>14</td>
<td>Material is enough</td>
</tr>
<tr>
<td>13</td>
<td>N/A</td>
</tr>
<tr>
<td>15</td>
<td>The figures showing positions of sensors might help</td>
</tr>
<tr>
<td>16</td>
<td>practice, different steps</td>
</tr>
<tr>
<td>17</td>
<td>all</td>
</tr>
<tr>
<td>19</td>
<td>The visual displays translated over to the actual installation</td>
</tr>
<tr>
<td>18</td>
<td>The map (textbook) is useful</td>
</tr>
<tr>
<td>20</td>
<td>going through the process of placing the items, the descriptions of how and where to place the items</td>
</tr>
</tbody>
</table>

5.3 Conclusion

This article reports on a study that tests the effectiveness of teaching participants how to install a smart home. An experiment compared two types of teaching methods. One method uses a booklet with three pages of floor plans and with sensors marked at their correct positions. The other method uses an intelligent tutoring system, SHiB ITS, which includes
three training exercises and four post-tests. The effectiveness of each teaching method is evaluated terms of 1) the accuracy rate of a physical smart-home installation, 2) the time spent in the installation, and 3) participants’ subjective viewpoints about the installation.

20 participants took part in the experiment. They were randomly assigned to two groups: a control group and an experimental group, with 10 participants in each group. Each participant received training first, and then started the physical smart-home installation in an apartment that was rented for the experiment. After the installation, participants were surveyed by a questionnaire about the installation. The performance of each participant was evaluated by the authors.

The results of the experiment show that the accuracy rate of the two groups is similar, 77% and 78%, with no significant difference. Time used was significantly different between the two groups: time spent by participants in the experimental group is significantly \((p < 0.05)\) shorter than those in the control group. The result suggests that training with the SHiB ITS will help to reduce actual installation time, which can be explained as reducing unnecessary work during the installation, which in turn will reduce the risk of injury for senior installers. Results of the post-survey imply that participants training with SHiB ITS before the installation have significantly \((p < 0.01)\) more confidence than those training via a booklet. Participants in the experimental group think that the SHiB ITS helps them in determining sensor positions, getting familiar with the installation manual, and translating the content learned in the ITS to the actual installation.

There are some limitations of this study. The sample size of participants is small when compared to the number of potential senior installers. Most participants of this study are young, with a mean age 29.4, and may not accurately represent a senior population. The SHiB ITS helped in reducing time spent in the actual installation, but needs improvement in increasing the accuracy rate of actual installation. For example, details related to installing the door sensor and server could be included in the SHiB ITS. A larger scale experiment could examine the knowledge translation of the SHiB ITS virtual environment to physical
installations.

## 5.4 Additional experiment material

Table 5.4 Questionnaire of post-survey

<table>
<thead>
<tr>
<th>Question No.</th>
<th>Question</th>
<th>Very Difficult (1)</th>
<th>Difficult (2)</th>
<th>Neutral (3)</th>
<th>Easy (4)</th>
<th>Very Easy (5)</th>
<th>N/A</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>How easy was it to install the sensors?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>How easy was it to install the names and terms for items?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>How easy was it to install the server box?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>How easy was it to follow the instruction manual?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>How easy would it be for other people to install the sensors and server box?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Overall, how easy was the entire installation process?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>How easy was it to follow the installation software?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Would you say the installation process was long?</td>
<td>Strongly Disagree (1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Would you say that you were frustrated or anxious while installing the sensors?</td>
<td>Strongly Disagree (1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Would you say the installation tutorial software was helpful for the actual installation?</td>
<td>Strongly Disagree (1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Would you recommend installing a Smart Home in a Box to others?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Did you receive any help with installing the sensors?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 6

Additional Analysis of user data collected from the SHiB ITS experiment

In this chapter, the author will present work that has done for processing the SHiB ITS online experiment data. The work includes applying Principal Components Analysis (PCA) and Hidden Markov Models (HMMs) to analyze the data.

6.1 Purpose of the study

The primary purpose of this study is to improve the training features of a tutoring system by analyzing the data obtained from the SHiB teaching effectiveness experiment. The author recruited participants from Amazon’s Mechanical Turk. The experiment tested three teaching strategies (trial-and-error, textbook, and combination) in two age groups (younger than 50 year’s old and older than 50 year’s old). Results of the experiment can be found in Chapter 4. The results show participants’ post-tests performance are affected by the teaching strategies, and the strategies affect differently regarding age.
The improvement the author would like to do is having the system identify participants’ plan in real-time by observing the actions they have made. If a participant is going in the direction to a wrong answer, the system will intervene and give suggestions. A possible way to start is to know users’ plans in advance: either they are right or wrong. In the context of Hidden Markov Models (HMMs), the plans can be represented as hidden stats. Types of the state-dependent distributions depend on the type of data.

6.2 Data type

The ITS system collected data when participants finished a task. The data were decisions, or steps participants made when they were doing the tasks assigned by the tutoring system. An example of the user interface of the tutoring system can be found in Figure 4.1 in Chapter 4. Participants controlled the red avatar by pressing keyboard. The avatar can perform six actions: pick up devices, move (up, down, left, and right), and place devices. The avatar moves a grid space at a time. A list of requirements for the devices is at the right to the layout. The list is available for participants for all eight tasks. The author selected data of 96 participants who finished all the tasks in the experiment. The participants include both senior and young adults, and were randomly assigned by the website to one of three groups. Their post-test performance were analyzed in the Chapter 4. 48 of them are older than 50 years old, 48 of them are younger than 50 years old. The author collected their background at the beginning of the experiment, such as gender, marriage, nation, race, native language, education, and so on.

A data file contains a participant’s steps in the pre-test, three training exercises, and four post-tests. The data does not have the time stamp. The original records are actions the participant had made to move devices, and the actions include left, right, up, and down. Each data file contains the device placement verifications of all eight tasks. In a task, if the participant placed a device at a right spot, it is labeled as “true”; otherwise, it is “false.”
When the author was processing the data, she converted participants’ actions (left, right, up, down) to a grid of (x,y) for the location of the avatar.

The author calculates the accuracy of the device placement of a participant in a task by counting the number of “true”s and the number of “false”s. The author obtained the accuracy of the teaching strategies and the accuracy of placements of types of devices in the same way.

The number of actions each participant spent on each device in each task reflects the degree of struggle when placing the device in the task. If a participant knows where the right spot is, he or she will spend the minimum number of actions to move the device to that spot. Otherwise, the number of actions is greater than the number of actions in the shortest path. Spending more steps in a task is caused by changing the device’s position multiple times to find the right position for the device. In some extreme case, the number of actions is almost five times greater than the minimum. Analysis shows participants in the textbook group practicing worked examples spent the minimum number of actions in training, but they spent more actions than the other two groups in post-tests. Besides, the post-tests performance of participants in practicing worked example group is also worse than that of the other two groups.

Participants in the trial-and-error group had the most struggles in training. Because they were not given hints in advance, therefore they have to try different localities to see which is the correct one. The trajectory of a participant’s sequential actions reflect how he or she changed the plan or idea of placing devices in a layout regarding the suggestions given by the system after a solution submission. The tutoring system suggestions are highlighted areas that are right for the device. Figure 6.1 shows an example of suggestion areas for a device.

In this chapter, the author will present data analysis of the sequences of actions each participant made for each device in each task. For this type of data, the author has some ideas about dealing with them. One thought is to use the spots in which participants had
Figure 6.1 An example of tutoring system interface

placed devices. Because under a teaching mode, if a participant had placed a device in a wrong spot, the system will show the participant the right area (as shown in Figure 6.1.) Data of spots that participants had tried to reflect their understanding about the list of requirements for each device. Statistical analysis about the spots that all participants had tried for a device should be helpful in classifying different types of understanding about the requirements. A case study mentioned in Zucchini et al. (2016) that if the data is comprised a series of locations, a method is to use the mixed HMMs to analyze them.

6.3 Related works

Zucchini et al. (2016) showed an example that applies HMMs to analyze the effects of climate change regarding animal movements with an assumption that the measurement error in the recorded locations is negligible. The original data are recorded geographic locations by GPS. The authors first convert the locations into step lengths and turning angles. The two are considered independent of each other in the example. The step lengths have a gamma distribution, while the turning angles have a von Mises distribution in the example. The study considered the states of animals as “encamped” / “resting” or “exploring” / “migrating.” The study included three examples.

The first case study is analyzing movements of fruit flies by HMMs. They compared two-state HMMs and three-state HMMs for both wild and mutant fruit flies. They applied
Akaike information criterion (AIC) and the Bayesian information criterion (BIC) to select the number of states of HMMs. Based on the definitions, the lower the AIC and BIC values are, the suitable the number of the corresponding states. Their results of log-likelihood, AIC, and BIC show for both flies that three-state HMMs are more suitable for the dataset. In the second case study, they present the application of HMMs on movements of bison. They applied a two-state HMM and a two-state HSMM (Hidden Semi-Markov Model) to the data. The results show that the parameters of the state-dependent distribution in the two models vary similarly. So does the AIC values of the two models. In the third case study, they applied mixed HMMs for 97 family groups of woodpecker movement data. Woodpeckers moved from tree to tree, the shortest series of those considered comprises only 15 locations, and the most extended series 125 locations. The authors applied two-state HMM to the problem. They implemented three different specifications of the random effects, and the results show an independent beta-distributed random effect has the optimal AIC value.

Ye et al. (2013) conducted a study that predicts the location that is most likely to be visited by a location-based social networks user. The researchers use everyday life divisions of locations, such as entertainment, shopping mall, and so on, to label the data which were in geographical longitude and altitude. The authors use the check-in category data as an observation to model the underlying user movement pattern by an HMM. They assumed that the time is homogeneous and the initial state distribution $\pi$ for the HMM is stationary. They then mixed the HMM with temporal and spatial covariates. They followed the multivariate logit model to formulate the state-dependent probability. BIC was employed to find the optimal number of states for both HMM and mixed HMM. Markov chain Monte Carlo (MCMC) Bayes Estimation was employed to estimate the parameters of the mixed HMM.
6.4 An overview of the pattern of data - PCA

Principal components analysis (PCA) is a technique that can bring out a pattern in a dataset. In the current study, the dataset consists of sequences of locations that participants had selected for devices in tasks. The author performed data processing before applying PCA to the dataset. This step is necessary as participants had made a various length of series of locations for each device. Since PCA analyzes data as a matrix and the lengths of the location series of each device are different, the first step in data processing is to unify the lengths of the device location series. The author set the most extended length as the standard length. Sequences of locations whose lengths are shorter than the standard length have additional locations added to the ends until the lengths are equal to the unified one. The additional locations are the last location of the original sequences. In such way, the number of steps for each participant is the same.

As the PCA analyzes data as a matrix with each element is a single number, the second step is to change the location data format from 2-Dimension to 1-Dimension. Each original location data is comprised of two components: $x$ and $y$. As the relationship between $x$ and $y$ is not linear, the two components will affect the result of PCA. A method the author applied is to project locations to a unit vector $\vec{u}$, which is

$$\vec{u} = [\sin\left(\frac{\pi}{4}\right), \cos\left(\frac{\pi}{4}\right)]^T$$

based on the feature of the unit vector, the values of its $x$ and $y$ are identical. In such way, the 2-Dimension data reduce to 1-Dimension.

The third step is to apply PCA in R to the data of 65 devices. Devices are independent of one another. Therefore, the author applies PCA to each device. Results show Principal Component 1 (PC1) and Principal Component 2 (PC2) covered over 90% of the variance of the data of each device. We use the PC1 and the PC2 as the main features of each device. By observing the coefficients of the PCs, we found that PC1 covers most of the steps except the first step. However, PC2 covers only the first step. We are using the PCA package
that applies Singular Value Decomposition (SVD) for the analysis. Therefore, the values of each PC are the co-variances of devices that were rotated by the principal axes or the singular vectors. Table 6.1 shows statistical results of PC1 and PC2 of the devices over 96 participants.

The author performed a hierarchical clustering on the standard deviation of PC1, and the result is shown in Figure 6.2. Smaller standard deviation values indicate most participants are consistent with a specific area of placing the device. Higher standard deviation values indicate the spots for that devices vary over participants. In Figure 6.2, standard deviation values of the servers are minimum so that they at the left end of the clusters; standard deviation values of relays and room sensors are maximum, and they are at the right end of the clusters. All the servers are under a cluster means the standard deviation of the co-variances of the servers is close. That indicates most participants agree with the same location for each server. That is due to the fact that the server is the only device in the SHiB ITS that participants do not have to choose from multiple spots. In the block at the right end of the cluster, participants have a variety of viewpoints on the placement of relays. That is caused by in each task that there are several spots for the relays, and the placement of a relay in a task should be different from the placement in the next task. In this case, the standard deviation values can be an indicator of examining how multiple solutions are applied when participants are using the tutoring system.

The author also performed a hierarchical cluster on the standard deviation of PC2, Figure 6.3 shows the result. Since PC2 covers the variance of the first step of participants, the cluster shows most participants have consistent ideas of the at their first decision made for the servers, room sensor 1 in post-test 1, and the kitchen sensor 2 in post-test 4. On the other hand, participants have different ideas regarding the locations for relays, room sensors, and kitchen sensors in the training exercises, the living room sensors, and room sensor 1 in post-test 4.
<table>
<thead>
<tr>
<th>Device</th>
<th>st.dev</th>
<th>mean</th>
<th>max</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>p2.l1</td>
<td>1.04E-01</td>
<td>5.31E-07</td>
<td>2.36E+00</td>
<td>-9.77E-07</td>
</tr>
<tr>
<td>p2.l2</td>
<td>1.34E-01</td>
<td>1.00E+07</td>
<td>2.10E+01</td>
<td>-2.08E+01</td>
</tr>
<tr>
<td>p2.k1</td>
<td>1.12E-01</td>
<td>7.35E-07</td>
<td>2.64E+00</td>
<td>-1.90E+01</td>
</tr>
<tr>
<td>p2.r1</td>
<td>1.28E+00</td>
<td>1.35E+01</td>
<td>1.58E+01</td>
<td>-9.75E+00</td>
</tr>
<tr>
<td>p2.r2</td>
<td>7.05E+00</td>
<td>1.40E+00</td>
<td>2.00E+01</td>
<td>-4.25E+00</td>
</tr>
<tr>
<td>p2.r3</td>
<td>1.09E+00</td>
<td>2.02E+04</td>
<td>1.80E+01</td>
<td>-1.80E+01</td>
</tr>
<tr>
<td>p3.l2</td>
<td>4.07E+00</td>
<td>8.98E+00</td>
<td>3.00E+01</td>
<td>-5.00E-05</td>
</tr>
<tr>
<td>p3.l1</td>
<td>8.03E+00</td>
<td>5.41E-07</td>
<td>1.24E+01</td>
<td>-9.60E+00</td>
</tr>
<tr>
<td>p3.k1</td>
<td>8.72E-01</td>
<td>1.02E+00</td>
<td>7.28E+00</td>
<td>-3.35E+00</td>
</tr>
<tr>
<td>p3.r1</td>
<td>8.31E-01</td>
<td>1.02E+00</td>
<td>4.82E+00</td>
<td>-9.33E+00</td>
</tr>
<tr>
<td>p3.r2</td>
<td>4.80E-01</td>
<td>8.45E-08</td>
<td>5.65E+00</td>
<td>-5.55E+00</td>
</tr>
<tr>
<td>p3.r3</td>
<td>1.35E+00</td>
<td>1.40E+00</td>
<td>2.40E+00</td>
<td>-3.60E+00</td>
</tr>
<tr>
<td>p4.l2</td>
<td>1.50E+01</td>
<td>7.28E+00</td>
<td>2.13E+01</td>
<td>-5.00E-05</td>
</tr>
<tr>
<td>p4.k1</td>
<td>1.05E+01</td>
<td>1.13E+00</td>
<td>8.00E+00</td>
<td>-7.80E+00</td>
</tr>
<tr>
<td>p4.r1</td>
<td>1.28E+00</td>
<td>2.04E+00</td>
<td>2.11E+01</td>
<td>-5.80E+00</td>
</tr>
<tr>
<td>p4.r2</td>
<td>3.01E+00</td>
<td>1.01E+00</td>
<td>7.97E+00</td>
<td>-1.59E+01</td>
</tr>
<tr>
<td>p4.r3</td>
<td>1.29E+00</td>
<td>1.21E+00</td>
<td>8.42E+00</td>
<td>-4.00E+00</td>
</tr>
<tr>
<td>p4.k2</td>
<td>1.29E+00</td>
<td>1.21E+00</td>
<td>8.42E+00</td>
<td>-4.00E+00</td>
</tr>
<tr>
<td>p4.r4</td>
<td>1.29E+00</td>
<td>1.21E+00</td>
<td>8.42E+00</td>
<td>-4.00E+00</td>
</tr>
</tbody>
</table>
Figure 6.2 Hierarchical cluster on the standard deviation of PC1
Figure 6.3 Hierarchical cluster on the standard deviation of PC2
6.5 Apply mixed HMM to the data

The function of applying two or more distributions to a mixed Hidden Markov Model (HMM) is shown below:

\[ L_T = \delta P(x_1) \Gamma P(x_2) \Gamma P(x_3) \ldots \Gamma P(x_T) \mathbf{1}' \]

where \( \delta \) is the initial distribution, a \( 1 \times n \) matrix, with \( n \) states. \( P \) is a \( n \times n \) matrix and elements of the matrix are probabilities. \( \Gamma \) is the state transition probability matrix (t.p.m), which is a \( n \times n \) matrix. \( x_i \) is the \( i \)th value in a time series. \( T \) is the length of the time series.

Studies applied statistic distributions to mixed HMMs to analyze data. For example, in Zucchini et al. (2016), the author uses a Poisson distributions to analyze the observations of significant earthquakes in 1900 through 2006. McKellar et al. (2014) applied a Gamma distribution and a Von Mise distribution to the motion data of 97 groups of woodpeckers. The authors convert GPS location data into step lengths and turning angles. Step length was the gap of steps when animals were moving. Turning angle is the circular radian between every two steps. Ye et al. (2013) applied the softmax function and the multivariate logit model to obtain the probability of categories in hidden states to learn a model from the data collected from a local social network.

In the case of SHiB ITS user data, the three distributions, the Poisson distribution, the mixed Gamma distribution, and the Von Mise distribution mentioned above are not suitable. As the SHiB user data are a series of decisions, therefore, the Poisson distribution (Zucchini et al., 2016), which is applicable for the number of events occur in a certain time interval, such as the times of earthquakes in ten years. However, participants spent different time length to accomplish the task; therefore, the Poisson distribution is not applicable to the SHiB user data.

The data conversion method mentioned in McKellar et al. (2014) is not applicable either. Many SHiB users have one step length, which means they move and place the devices only once, and only a few of them changed their decision later. The Gamma distribution HMM
requires the number of step lengths for each data to be at least two. Therefore, the SHiB user data is not suitable for using the Gamma distribution and Von Mise HMMs.

The method mentioned in Ye et al. (2013) classifies the location data, and then uses the class as input data to model the HMMs. They use the locations’ feature to classify, such as if a location in the map is a mall, then they classify the location as a shopping place. If a location in the map is a cinema, then they classify the location as entertainment. The SHiB user data is a series of locations in a house layout: locations in a house can be bedrooms, restrooms, kitchen, and living room. However, the classification is too broad for the SHiB user data. Even though a user is in a bedroom, there are correct locations and incorrect locations.

The author applied a Reinforcement Learning (RL) algorithm, Sarsa, to let an RL agent learn a task of placing a SHiB device. The agent starts at different places of the house layout in each episode. The training continues until all the states’ Q-values of the placing SHiB device task converge. Then the author uses the Q-values to match locations of SHiB user data. As the Q-values are continuous, the author decided to use the Normal distribution HMMs to analyze the data.

The author uses the converted user data generated three multiple state Normal distribution HMMs. Each Normal distribution HMM has two settings, stationary and homogeneous. Figure 6.4 displays the AIC and BIC values of three multiple state Normal distribution HMMs. The number in each notation represents the number of states, and the “s” represents stationary while the “h” homogeneous. In the figure, AIC and BIC values of two states and three states are lower than four states, which indicates to use two states Normal distribution HMMs and three states Normal distribution HMMs to make the behavior prediction will have better results than using four states HMMs.

Three Normal distribution Hidden Markov Models (Normal-HMMs), both stationary and homogeneous, performed predictions of the state. Each Normal-HMM has ten tests. The tests consist of data of participants from three groups. The Normal-HMMs have the first two
steps of the participants, and then the HMMs have to predict all the rest steps. The accuracy of each participant is the fraction of correct predictions over the total number of predictions. Mean values of the accuracy of the Normal-HMMs were computed by averaging the values of the accuracy of participants. Table 6.2 shows the prediction results of the Normal-HMM. The mean accuracy is the results of cross-validations. The results show that the highest accuracy lies in three states, which is 58.68%, then the two states, which is 58.13%. The four states have the lowest accuracy. A hidden state represents a normal distribution. The results show the accuracy of three normal distributions is higher than two or four normal distributions.

Table 6.2 Mean accuracy of predictions of Normal-HMMs

<table>
<thead>
<tr>
<th>state #</th>
<th>Mean Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>58.13%</td>
</tr>
<tr>
<td>3</td>
<td>58.68%</td>
</tr>
<tr>
<td>4</td>
<td>34.51%</td>
</tr>
</tbody>
</table>
6.6 Conclusion and limitation

This study performed Principal Component Analysis (PCA), and Hidden Markov Models (HMMs) mixed with Normal distribution analyze the SHiB ITS users’ data. The results of the cluster of the PCA show participants have similar viewpoints of the placements of servers but have a variety viewpoint of placements of relays. The standard deviation of the PC1 is helpful to give an intuitive view of how the multiple solution requirements and the task for each device affect participants’ behavior.

The time series decisions prediction by the Normal distribution mixed with HMMs shows with three states, in both stationary and homogeneous, the accuracy rate is 58.68%, which indicates the data as a whole could have three normal distributions. There are three groups in the user data. The three normal distributions indicate the Q-values that converged from the RL agent reflects users’ behaviors that were affected by the teaching strategies.

The limitation of applying the Normal distribution mixed HMMs to tutorial system lies in it is an offline prediction, whereas the tutorial system needs a method that can use online prediction for the user behaviors. Another limitation is the size of the dataset. From the prediction results of the two-state Normal-HMMs and the three-state Normal-HMMs, the two accuracy rates are close. That shows the method can either differentiate according to the age of the participants or make differences between groups. The behavior of participants of various ages in each group cannot be distinguished. A better way to deploy suggestions for the tutorial systems is to have a teaching strategy that does not wait for users to act. The strategy should also know which mistakes it needs to address and which not.
Chapter 7

Learn to Teach an agent to be a teacher

In this chapter, the author presents four algorithms that help a Reinforcement Learning (RL) agent to decide when to advise an RL agent so that to help RL students complete a task. The chapter includes the purpose of the study, algorithms, corresponding results, conclusions, and limitations.

7.1 Purpose of the study

As an extension to the previous work of Smart Home in a Box (SHiB) ITS (Chapter 4, 5, and 6), an improvement can be made to help the ITS users is to advise them before they start next actions. A piece of good advice should be delivered at the right time, especially when a user is going to put a device to an incorrect position. Also, the suggestion should point to a direction but will not tell the user where the correct position is. The pre-defined sequential activity suggestion setting is not applicable in such a case because users have a variety of ideas about spots to place a device (as stated in the conclusion of Chapter 6).

A good teacher usually has adequate experience in teaching certain subjects. When she
observes a student doing homework, at a particular step, before the student writes down the wrong answer, the teacher will be able to tell if the student will make mistakes in a few steps or not. Such ability is from experience accumulated by teaching and answering students’ questions.

A new teacher grows to become experienced by teaching students frequently. Sometimes the process takes years. An analogous setting is where an ITS can teach people specific tasks by learning from teaching other agents (human or RL agents) to finish tasks to become an “experienced” teacher. An advantage of gaining teaching experience by tutoring people is that the complexity of the training tasks is similar to that of final teaching tasks. The disadvantage of performing such training is that it is time-consuming.

On the other hand, an RL agent can learn to perform a task as human learners. The agent will not act as intelligently as humans at the very beginning. However, the learning process is similar, in that the agent will accumulate experience performing the task. The advantage of using RL agents as students to train an RL teacher is that the training task can generate as many RL students as needed.

The purpose of this chapter is to explore teaching methods and RL teacher training strategies that an RL teacher can perform similar teaching activities as human teachers. Training an RL teacher does not focus on the teaching content. However, the training tasks focus on students’ performance after receiving advice from the RL teacher. Therefore, the teaching contents can be any tasks. In this chapter, the author chose a Pac-Man game as the teaching contents. Previous studies (Fachantidis et al., 2018; Taylor et al., 2014; Torrey and Taylor, 2013; Zhan et al., 2014) had worked on exploring suitable feature sets for regular RL agents to play a Pac-Man game (Torrey and Taylor, 2013), exploring variant factors on teaching methods (Taylor et al., 2014; Torrey and Taylor, 2013; Zhan et al., 2014), and on and offline Q-learning methods (Sutton and Barto, 2018) to training an RL teacher (Fachantidis et al., 2018). Torrey and Taylor (2013) introduced the feature set an approximate RL agent can learn to play Pac-Man game, the work also introduced concepts of state importance, and
using SVM as assistance to predict students’ future activities. Zhan et al. (2014) proposed teaching methods with a suggestion-budget, advising after the student performed certain steps, using state importance as a threshold on making advising decisions, and gradually decreasing the threshold of state importance. Fachantidis et al. (2018) introduces factors of a feature set for the Q-learning methods, the features they considered include advising decision, remaining budgets, students’ learning progress, and students’ intended activities. The reward function they use to update the Q-values of teaching activities is the Q-value difference between the action from the teacher’s perspective and the action performed by the student.

In this chapter, the author addresses the problem in three types of methods. The first one is to investigate the effects of parameters on RL students’ learning curves. The second one is to propose an approximate Reinforcement Learning method to train an RL teacher to teach. Finally, the author suggests a way of using the Artificial Neural Network (ANN) to predict the time of giving suggestions.

7.2 Methods

In this section, the author presents three methods that are implemented to train an RL teacher. The first method is to investigate the effects of thresholds of state importance on teaching outcome; the second method is to apply an approximated State-Action-Reward-State-Action (SARSA) to train an RL teacher to teach. The third method is to train an Artificial Neural Network (ANN) to model the behavior of a mistake correcting teacher (Taylor et al., 2014) with a fixed threshold.

7.2.1 Parameter Effects

As mentioned in (Taylor et al., 2014), the state importance from an RL teacher’s perspective is critical in choosing when to advise on RL student. State importance is defined as the
difference between the maximum Q value and the minimum Q value of a state that the student encounters. The value of state importance implies how big the gap between the two Q values is. The more significant the difference is, the more critical the state is. It could directly lead to a rewarding state or a cliff. It is not practical to advise RL students at any state importance value. With the consideration of saving a suggestion budget, an RL teacher chooses to advise an RL student at high state importance states can save on the budget from suggesting at low state importance states. That is because the frequency of top importance states is rare relative to that of low importance states.

Taylor et al. (2014) has extended the state importance to two more representations. One of them represents the state importance by the variance of an RL teacher’s Q values. So the other one describes the state importance by the absolute deviation-based importance of an RL teacher’s Q values. They investigate different values of the threshold for state importance. In this section, the author extends the work from (Taylor et al., 2014) to examine the effects of importance threshold on teaching outcomes, which in this section, is the RL student’s performance.

As stated in (Taylor et al., 2014), an RL teacher that corrects an RL student after the student announced the next action outperformed the other proposed teaching approaches. To compare the teaching outcome, the author adopts the mistake correcting teaching approach as the upper bound of the teaching outcome. The author then proposed two methods in this section on RL teacher teaching RL students. The first approach is named Dice Advise Importance and the other one is named Memory Advise Importance. Both approaches require threshold τ inputs. The procedure of Dice Advise Importance is shown in Algorithm 1. The method of Memory Advise Importance is shown in Algorithm 2.

In Algorithm 1, the BUDGET represents the number of available suggestions the RL teacher can deliver τ represents the threshold for state importance, and the dice represent a threshold to control the chance for an RL teacher can provide the advice. With the author’s
observation, a Pac-Man RL agent encounters critical state more often in later episodes. If the RL teacher used up the budget in early episodes, then the teacher cannot help the student when it encounters such critical states. By using dice to control suggestion delivery, it can extend the time of availability of the RL teacher.

\begin{algorithm}
\textbf{input :} BUDGET, τ, π, dice
\textbf{output:} Action
\textbf{for each student state }s \textbf{ do}
\hspace{1em} dp=\text{random number};
\hspace{1em} \textbf{if } BUDGET > 0 \text{ and Importance}(s) \geq τ \text{ and } dp \geq \text{dice } \text{ then}
\hspace{2em} \text{Advise } π(s);
\hspace{2em} \text{BUDGET} -= 1;
\hspace{1em} \textbf{else}
\hspace{2em} \text{Advise } = \text{None};
\hspace{1em} \textbf{end}
\textbf{end}
\textbf{Algorithm 1:} Dice Advise Importance
\end{algorithm}

From Algorithm 1, the author extends the means of deciding the value of dice to a statistical way in Algorithm 2. In this approach, the RL teacher records all the state importance values that are greater than the threshold τ and their frequencies. Then the decision of giving suggestion is determined by 1.0 minus the halved importance frequency. The purpose of the equation is to reduce the chance of delivering the same advice at the same importance states. Through such an approach we extend the length of the available period of the RL teacher.
input : BUDGET, τ, π, dice, ImportanceFreq-list

output: Action

for each student state s do
  if BUDGET > 0 and Importance(s) ≥ τ then
    ImportanceFreq = ImportanceFreq-list(Importance(s));
    dice = 1.0 − \frac{1}{2} \cdot ImportanceFreq;
    dp = random number;
    if dp ≥ dice then
      Advise π(s);
      BUDGET-=1;
    else
      Advise = None;
    end
  else
    Advise = None;
  end
end

Algorithm 2: Memory Advise Importance
7.2.2 Approximate SARSA

SARSA is an RL algorithm for on-policy control (Sutton and Barto, 2018). On-policy methods try to evaluate or improve the policy that is used to make decisions. A general equation of computing the Q-value of each action at each state is

\[ Q(S_t, A_t) = Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)] \]  \hspace{1cm} (7.1)

The Q-value update is performed after every transition from a state \( S_t \) to state \( S_{t+1} \). If state \( S_{t+1} \) is terminal state, then \( Q(S_{t+1}, A_{t+1}) \) is defined as zero. The \( \alpha \) is denoted as learning rate, and \( \gamma \) is denoted as a discount. In the static environment case, where the size of state space is limited, the world is represented as a table. Each grid in the table is considered as a state, each state has an action set, \( A_t \in \text{Action} \), with \( | \text{Action} | > 1 \). In a tabular RL method, the table contains the Q-values of all the actions.

For problems with arbitrarily large state spaces, such as continuous states, most encountered states will never have been seen before. The tabular RL methods become inapplicable in such cases as the large state spaces will need greater memory size for the table, as well as the time and data, are required to fill the table accurately. Approximate RL methods are using the generalization of previously encountered states and make similar decisions in similar situations. For each task, the encountered states are considered as examples of some desired function and the approximate methods attempts to generalize from them and then to construct an approximation of the desired function. Therefore, the approximate RL method is an instance of supervised learning. The q-values of actions are the product of two vectors, and a weight vector \( \mathbf{w} \) and a feature vector \( \mathbf{x} \). At each state, after receiving a reward from the environment, the agent updates its weight vector with the q-values of the current state and the rewards it received from the environment. The update at a time \( t \) only considers the current values of weight vectors and an approximation of state value, which implies the target output will be biased and that makes the method not a true gradient descent method. Therefore, the method is called a semi-gradient method. Equation 7.2 shows the equation.
of the weight updating of semi-gradient one-step Sarsa algorithm.

\[
\mathbf{w}_{t+1} = \mathbf{w}_t + \alpha [R_{t+1} + \gamma \hat{q}(S_{t+1}, A_{t+1}, \mathbf{w}_t) - \hat{q}(S_t, A_t, \mathbf{w}_t)] \nabla \hat{q}(S_t, A_t, \mathbf{w}_t) \tag{7.2}
\]

Most of the symbols of Equation 7.2 are identical to that of Equation 7.1. The \( \hat{q} \) represent the approximate q-values. The term \( \nabla \hat{q}(S_t, A_t, \mathbf{w}_t) \) is the feature vector \( \mathbf{x}_t \). Equation 7.3 shows method to compute the \( \hat{q}(s, a, \mathbf{w}) \) in equation 7.2. The estimation occurs at each pair of state, \( s \), and action, \( a \).

\[
\hat{q}(s, a, \mathbf{w}) \doteq \mathbf{w}^T \mathbf{x}(s, a) = \sum_{i=1}^{d} w_i \cdot x_i(s, a) \tag{7.3}
\]

The state space of training a Pac-Man RL teacher task is greater than the tabular case. A reason is that the state space of the Pac-Man task is continuous. In the task, the states of an RL teacher are the steps where a student may or may not make a mistake. The teacher’s action set is binary, with two actions: to give a suggestion or not to give a piece of advice. If a teacher provides an advice, but the student made the right decision, then it turns out the teacher take the wrong action because the advice was wasted. As the student is an RL agent, it also keeps learning and updating its Q function. Even at the same state, the student may make a different decision at each time, as the student may explore.

Analogous to a human teacher, the RL teacher observes the RL student’s behavior. The observation could be the task progress where the student is at, or the relative performance compare with the student made in the past. Also, it should not include the student’s Q function as a human teacher cannot see what is in a student’s mind. The target of the teacher training task is to let the teaching experience as many RL students as possible and to learn to give suggestions at the proper time.

**Feature Construction**

Each step of the training teacher task has two pairs of states and actions. The state is represented by five features, \( | \mathbf{x}_t(s, a) | = 5 \). \( x_1(s, a) \) represents the difficulty level at state,
A critical state tends to have a higher value of state importance. For example, when a student is solving a linear equation, the student has a calculator as a helper so that the math computation will not affect the final answer as long as the student inputs the correct number. However, if the student put a wrong sign on the number, then that will directly affect the final answer. In this case, putting the correct sign on the numbers are more important than the math calculation. In the Pac-Man case, a state with high state importance indicates the correct action will receive a reward, such as eat a bean or encounter a ghost, which will end the game immediately. At the teacher level, choosing a threshold is critical. If a teacher is always telling a student what to do at the current step, it will annoy the student a lot. However, if a teacher only gives suggestions at the crucial step, which will prevent the student from making critical mistakes, it will help the student improve quickly. To limit the number of suggestions, a threshold for the state importance is necessary. In the last section, the author conducted multiple preliminary experiments on choosing the threshold for importance. The results show a threshold at the center of the range from the minimum through the maximum of the teacher’s q-values has decent effects on improving the student’s performance. Therefore, the author uses a mean value of all the old states’ q-values to adjust the feature \( x_1(s,a) \).

\( x_2(s,a) \) represents the perspective of teacher on the student’s situation. The idea comes from the place where a student is at in performing a task. \( x_3(s,a) \) is the accuracy rate of prediction of a SVM (Torrey and Taylor, 2013) trained by the sequences of student’s actions in previous episodes. \( x_4(s,a) \) is the squared value of \( x_1(s,a) \), and \( x_5(s,a) \) enables interactions between \( x_1(s,a) \) and \( x_2(s,a) \).

The action set, \( A \), has two actions, one is to suggest, the other is to not, with \( A = \{ a_1, a_2 \} \). To implement approximate SARSA methods the author represents the feature vector \( x(s,a) \) as below:
\[ x_1 = \text{Sigmoid}(\text{TSIM} - \text{midgap}) \]
\[ x_2 = \text{Sigmoid}(\text{AppStuGap} - \text{midAppStuGap}) \]
\[ x_3 = \text{Accuracy}(	ext{SVM}) \]
\[ x_4 = x_1^2 \]
\[ x_5 = x_1 \cdot x_2 \]

where, the \text{Sigmoid} is the logistic function, \( \text{Sigmoid}(x) = \frac{1}{1+\exp(x)} \), which returns a value in the range of \([0, 1]\). Teacher state importance (TSIM) is defined as below:

\[
\text{TSIM} = \begin{cases} 
\max Q_{\text{teacher}}(s_{\text{teacher}}, a_{\text{teacher}})_t - \min Q_{\text{teacher}}(s_{\text{teacher}}, a_{\text{teacher}})_t, & \text{if } a_t = 1 \\
\text{mean} Q_{\text{teacher}}(s_{\text{teacher}}, a_{\text{teacher}})_t - \min Q_{\text{teacher}}(s_{\text{teacher}}, a_{\text{teacher}})_t, & \text{if } a_t = 0
\end{cases}
\tag{7.4}
\]

In the training teacher task, \( a_t = 1 \) represents giving a suggestion at step \( t \), and \( a_t = 0 \) represents not giving a suggestion. \text{midgap} denotes the mean value of the differences between the maximum and minimum q-values from the first step, \( \text{midgap} = \frac{1}{n} \sum_{i=1}^{n} TSMI_{t=i} \). It represents the middle level of all the states the student has encountered. \text{AppStuGap} is the approximate student gap in the teacher’s perspective. The equation for \text{AppStuGap} is shown below:

\[
\text{AppStuGap} = \frac{\text{TSIM} + \text{AppStuGap} \ast \text{Steps}}{1 + \text{Steps}}
\tag{7.5}
\]

\[ \text{Steps}^+ = 1 \]

where the \text{Steps} represents the number of total steps the student has taken, and has an initial of zero. And the mid value of the approximate student gap is computed as: \( \text{midAppStuGap} = \frac{1}{\text{steps}} \sum_{i=1}^{\text{steps}} \text{AppStuGap} \).

At each time the SVM gives a prediction about the student’s next action after the RL teacher decided on giving or not giving a suggestion. The teacher will validate the SVM prediction and the actual RL student’s action. The RL teacher keeps on tracking the correct
and incorrect SVM predictions. The value of $Accuracy(CVM)$ is computed as below:

$$
Accuracy(SVM) = \begin{cases} 
\frac{\text{correct suggestions}}{\text{total suggestions}} & \text{if } a_t = 1 \\
1 - \frac{\text{correct suggestions}}{\text{total suggestions}} & \text{if } a_t = 0 
\end{cases}
$$

(7.6)

when the action, $a_t = 0$, the result of $Accuracy(SVM)$ represents the ratio of correct predictions over all the predictions made by the SVM.

Q-value evaluation

After the training task has obtained two Q-values, $Q_{a=1}$ and $Q_{a=0}$ from equation 7.4 through equation 7.6, the author performed an additional evaluation for the two Q values. The purpose of the additional evaluation is to use the percentile, $p_t$, of the difference, $d_t$, of the two Q values to all the difference values, $D_{1,t-1}$, from the first step until the current step to reevaluate the Q value of action $a = 1$. As stated in the last section, the threshold used in the feature value evaluation is an intermediate value of the importance of states that students have encountered.

To find out the percentile, $p_t$, of the current step difference value of the two Q-values, a set, $D_{1,t-1}$, of all the different values of Q-values is maintained from the first step until the step before the current one. The mean value, $\bar{D}$, of the difference values is obtained from the list. From the mean value $\bar{D}$, the variance, $v_D$, is obtained by the variance equation. And the corresponding probability density function, $PDF$, is estimated by the mean, $\bar{D}$, and variance $v_D$. The percentile, $p_t$, is the result of an integration with interval between last step different Q-value, $d_{t-1}$, and the current difference Q-values, $d_t$. A new Q-value, $Q_{a=1}^*$ is obtained by multiplying the original $Q_{a=1}$ and $1 - p_t$. The smaller the value of $p_t$, the more significant $Q_{a=1}$ is when compared to $Q_{a=0}$. If $Q_{a=1}^*$ is greater than $Q_{a=0}$, it implies $Q_{a=1}$ is significantly greater than $Q_{a=0}$. If it is not, it implies the $Q_{a=1}$ is greater than $Q_{a=0}$ but not significantly. The methods and equations are shown in Algorithm 3.

As shown in Algorithm 3, the percentile $p_t$ estimation happens when $d_t > \bar{D}$, which
guarantees that the difference value between $Q_{a=1}$ and $Q_{a=0}$ is higher than the mean value of all the difference values between the two $Q$ values in the current episode.

**Reward Shaping**

Regular reinforcement learning tasks often give the reward to the RL agent whenever it achieves a final state, such as finding the exit of a maze, or a state with prizes, such as eating beans in Pan-Man. To train a teacher the definition for a good teaching action is vague. For instance, an RL teacher helps an RL student to achieve a prize, the reward for the RL teacher could be the prize the RL student receives, or it could be the knowledge improvement of the student. If an RL teacher has a budget of suggestions to give, it would be better for the teacher to save the budget at the beginning steps where the student usually will not receive prizes. On the contrary, if an RL teacher quickly gives out all the suggestions, then the student will not have any help at the steps where the situation becomes complicated.

With the consideration of complexity in rating a teaching activity, the author proposes a reward which comprises multiple aspects of teaching action rating. The reward is shaped by four aspects: credits($r_1$) on giving suggestions different from student’s action, credits($r_2$) on saving suggestion budget, credits($r_3$) on giving suggestion at proper time, and credits($r_4$) on student’s post-activity reward. Each type of credits takes 25% of the total reward $R_{t+1}$ that the RL teacher will receive after the teaching activity is taken. The methods of computing each of the credits are listed below:

$$r_1 = \begin{cases} 
0.04 & \text{if } a_{teacher} \neq a_{student}, a = 1, \text{TSIM} \geq \text{midgap} \\
0.00 & \text{otherwise}
\end{cases}$$  \hspace{1cm} (7.7)

where $\text{TSIM}$ represents the difference between the maximum and the minimum of $Q$ values for state $s_t$. $a_{teacher}$ is the action that the teacher will take if the teacher performs the same task. $a_{student}$ is the action the student will take on the task. $\text{midgap}$ is the mean value of all the $\text{TSIM}$ of states the student had encountered in one episode. $a$ is the teaching activity,
input : $d_t, d_{t-1}, D_1,t-1, Q_{a=1}, Q_{a=0}$

output: Final action

if $d_t > 0$ then

$$\bar{D} = \frac{1}{t-1} \sum_{i=1}^{t-1} D_i$$

if $d_t > \bar{D}$ then

$$v_D = \frac{1}{t-2} \sum_{i=1}^{t-1} (d_i - \bar{D}), d_i \in D_{1,t-1}$$

$$PDF = \frac{1}{\sqrt{2 \cdot \pi \cdot v_D}} \cdot \exp\left(-\frac{(d_t - \bar{D})^2}{2v_D}\right)$$

$$p_t = \left| \int_{x=d_{t-1}}^{x=d_t} PDF \, dx \right| = | d_t \cdot PDF - d_{t-1} \cdot PDF |$$

if $Q_{a=1}^* > Q_{a=0}$ then

Final action $a = 1$;

else

Final action $a = 0$;

end

else

Final action $a = 0$;

end

else

Final action $a = 0$;

end

Algorithm 3: Q value re-evaluation
to give a suggestion or not to give a piece of advice.

\[
\begin{align*}
  r_2 &= \begin{cases} 
    0.00 & \text{if } a = 1 \\
    0.04 & \text{if } a = 0
  \end{cases} \\
  r_3 &= \begin{cases} 
    0.00 & \text{if } a_{\text{teacher}} = a_{\text{student}} \\
    0.04 & \text{if } a_{\text{teacher}} \neq a_{\text{student}}
  \end{cases}
\end{align*}
\]  

(7.8)  

(7.9)

\[r_4 = R_{t+1, \text{student}}\]

(7.10)

And the weight updating with multiple reward objects is shown below:

\[
\begin{align*}
  w_{t+1} &= w_t + \alpha \left[ \frac{1}{4} \sum_{i=1}^{4} r_i + \gamma \hat{q}(s_{t+1}, a_{t+1}, w_t) - \hat{q}(s_t, a_t, w_t) \right] \cdot x
\end{align*}
\]

(7.11)

7.2.3 RL teacher behavior modeling by ANN

Behaviors of a mistake correcting RL teacher with a threshold are determined by three factors: the threshold, which is state importance in this study; the action of the RL student; and the environment, which is the Pac-Man game in this case. The threshold determines the states at which the RL teacher will be aware of the students’ announced actions. The RL teacher’s advise decisions are determined by the announced actions of the RL students. If the student announces an action that is distinct from the RL teacher’s action, then the teacher will advise the student, and the student will perform the suggested action, and vice versa. The Pac-Man game environment indirectly affects the RL teacher’s behavior via RL student-environment interactions. An RL student learns to perform a task well by updating the feature weights for its approximate function in a stochastic environment. Before the feature weights of an RL student converge to an optimum, the weight values vary within wide ranges as the environment changes successively. As the particular value of feature weights and their changes are not observable to the RL teacher, they can be viewed as hidden units when modeling the behaviors of a mistake correcting RL teacher.

96
An Artificial Neural Network (ANN) is a network that consists of interconnected components that represent different properties of target models. An ANN has an input layer and an output layer, and in between are the hidden layers. Except for the input layer (the first layer), each node in a layer is computed as a weighted sum of the input values, which is the output of the previous layer, and then the result is applied to a nonlinear function (Sutton and Barto, 2018).

In this subsection, the author proposes an approach of applying an ANN that takes inputs of features that are observable to the RL teacher and then outputs suggestion delivery decisions. The purpose of the ANN application is to model the behaviors of a mistake correcting RL teacher. The inputs for the ANN are five-fold: teacher state importance (TSIM\textsubscript{ANN}), the approximate student mid-state importance (AppStuGap), the score of the previous step (step reward), square of the TSIM\textsubscript{ANN}, and the result of multiplication of TSIM\textsubscript{ANN} and AppStuGap. The value of TSIM\textsubscript{ANN} is defined as the difference between the maximum of RL teacher’s Q value and the minimum of RL teacher’s Q value at state s. The ANN has four layers: an input layer, two hidden layers, and an output layer. Each hidden layer consists of a 2D weight matrix and a 1D bias vectors, and both are initialized as random values. Activation function of each layer is ReLu, which stands for Rectified Linear Unit, which is defined as ReLu(x) = \max(0, x). The reasons for choosing ReLu as activation functions include its cheap computation, fast converge, and sparsity.

Activities of a mistake correcting RL teacher with a threshold \( \tau = 375 \) and a suggestion budget of 100 for each RL student tutoring 20 RL students were collected as a training dataset. The dataset is in two folds: one is the features that the RL teacher can observe at each step, and the validation part is the suggestion delivery decisions (\( a = 1 \) or 0) made by mistake correcting RL teacher. Within the entire dataset, 80\% are randomly selected as the training dataset, and the remaining 20\% were the test dataset.

The author experimented with several dropout rates; finally, the dropout rates of 75\% and 50\% were chosen for layer 2 and 3. The purpose of using dropout is to regularize the
ANN and avoid overfitting. With the chosen dropout rates, at each iteration, 25\% of nodes in layer 2 (\# node=16) are randomly picked as parameters to compute $Z_3$ and $A_3$ for layer 3 (\# node = 4). Similarly, in layer 4 (output layer), the number of nodes is half the number of nodes in layer 3, therefore, at each iteration, 50\% of the nodes are randomly selected for the $Z_3$ computation. To measure the performance of the ANN, the author used the accuracy metric, sensitivity metric, and specificity metric. The sensitivity metric is defined as $\frac{\text{true positive}}{\text{true positive} + \text{false negative}}$, while the specificity metric is defined as $\frac{\text{true negative}}{\text{false positive} + \text{true negative}}$.

**Imbalanced data**

The imbalanced data of the activities of a mistake correcting RL teacher comes from the imbalance between suggestion budgets and the number of possible steps an RL student can perform in all its training trials. The suggestion budget the RL teacher can give for an RL student is set as 100 in this case. On the other hand, if an RL student can avoid all the ghosts in an episode, the maximum number of steps the game allows an RL agent to perform is 2,000. Each RL student can learn 400 episodes. If a student performs well at most tasks, the RL teacher will have suggestion budget left at around the 200th episodes.

The data set has two classes, one is to give a suggestion, and the other is not to. A mistake correcting RL teacher only give suggestions once the state importance satisfies the threshold ($\tau = 375$) and the RL student announced an incorrect action. The ratio between the class of suggesting to the other class is 1:360, which is higher than the proportion defined as extreme cases in (Zhou and Liu, 2006).

Previous studies handle the imbalanced data problem by two types of methods. One is augmenting the number of minority class samples by re-sampling the training data, which is also known as oversampling. The other is to reduce the amount of data in the majority class, which is known as undersampling. Zhou and Liu (2006) compared methods of the two types, and found out the oversampling and undersampling methods work well for regular imbalanced two-class data, and the range ratio of the number of samples of the two-class is
between 1:4 to 1:1.24. Then Zhou and Liu (2006) proposed another method named moving threshold, which in their study can handle extreme imbalanced data. The most extreme case in the study is 1:20.

Instead of manipulation of the dataset before performing training, the moving threshold modifies outputs of the trained neural network. The algorithm returns the index of the maximum output after a class-wise normalization. The equation of the threshold moving algorithm is shown in equation 7.12 (Zhou and Liu, 2006). The symbol $O_i$ denotes the output of a neural network. $Cost[i,c]$ denotes the cost of misclassifying an example of the $i$th class to the $c$th class ($Cost[i,i] = 0$). $\eta$ is a normalization term such that $\sum_{i=1}^{C} O_i^* = 1$ and $0 \leq O_i^* \leq 1$. In the current study, the $Cost[i,c]$ is defined by the ratio of the two classes in the dataset. That is, if an output is misclassified to give suggestion $Cost[i,give] = 360$. On the contrary, the cost is 1 if the output is misclassified as not to give the suggestion class.

$$O_i^* = \eta \sum_{c=1}^{C} O_iCost[i,c]$$

(7.12)

Implementation of the moving threshold algorithm requires a real numbered neural network output, which is different from the method the author proposed in the last subsection. A modification is made to have a real number output $Z_3$; the sigmoid operation on $Z_3$ is removed. The mean, and variance of each pair of the outputs are estimated, and then they are used to obtain the normalization value $O_i^*$.

The loss function of the ANN in this case also considers the imbalanced data effects. The equation of the loss function is defined by Abadi et al. (2015) as shown in equation 7.13.

$$Loss = w_{class} \cdot Y \cdot (- \log(sigmoid(\hat{Y}))) + (1 - Y) \cdot (- \log(1 - sigmoid(\hat{Y})))$$

(7.13)

where $w_{class}$ is the class weight vector, $Y$ is the vector of validation data, and $\hat{Y}$ is the prediction output vector, which equivalent to $Z_3$ in this study. The author applies Adam optimization (Kingma and Ba, 2014) method to minimize the value of $Loss$. 

99
7.3 Results

In this section, the author presents the results of the three methods that investigate strategies of training the RL teacher. The approaches are importance effect investigation method, approximate function method, and modeling behaviors of mistake correcting RL teacher by artificial neural networks.

7.3.1 Results of Parameter Effects

To investigate the effects of parameters (threshold of state importance, the threshold for Algorithm 1, and Algorithm 2) on an RL teacher’s teaching outcome, the author experimented with different state importance values and dice threshold values. Teaching outcome is represented by the average score of 30 RL students. The Dice Advice Importance teaching method (Algorithm 1) and the Memory Advise Importance method (Algorithm 2) are compared with mistake correcting method (Taylor et al., 2014).

The author first examined the effects of thresholds dice for dice advice algorithm. Figure 7.1 shows the results of comparison two threshold values, 0.5 and 0.75 over nine importance thresholds (100 through 450). The value of 0.5 represents the RL teacher has a 50% chance to suggest once the importance threshold is satisfied. On the other hand, 0.75 indicates that the RL teacher has 25% chance to suggest once the importance threshold is satisfied. Figure 7.1 demonstrates that once the threshold for random numbers is set to 0.75, the teaching outcome decreases when compared with the threshold of 0.50. The results indicate that when the RL teacher has a low chance (25%) to suggest, at most time, it does not improve RL students’ performance as much as the RL teacher suggesting with a high chance.

To identify a suitable state importance threshold, the author compared the teaching outcomes of the dice advice algorithm with 50% chance suggesting with the mistake correcting method. The proper state importance is defined as the one that reduces the gap between
the two teaching strategies. As students’ performance taught by mistake correcting method is considered as upper bounds, the author wants to find a teaching strategy under which the students’ performance is close to the upper bound. Figure 7.2 shows the comparison of the two approaches across nine importance thresholds, from 100 through 450. The gap between the two methods decreases as the importance threshold increases. The closest value gap appears around the value of 375 through 450, which means the suitable state importance lies in this range. Considering the scores of threshold = 375 and threshold = 425 are similar, 2535 and 2554, respectively, and a lower threshold will encourage the RL teacher to suggest more frequently, the author decided to choose 375 as the fixed importance threshold.

With the fixed importance threshold, the author then compared the teaching outcomes of the three algorithms: the dice advise algorithm, memory advice algorithm, and the mistake correcting algorithm. Comparison metrics is the mean scores of RL students at test episodes, in which the teacher is not available, and the student is not updating its weight vectors. The reason for using the mean score as metrics is that if a student is typically outperformed by other students, the mean scores of that student should also be higher than the rest students. The author experimented with 375 as the threshold of state importance in episodes from 0 through 405. Figure 7.3 shows the comparison results. Condition Memory 375 means the
Figure 7.2 Comparison of teaching outcome between dice advice algorithm and mistake correcting

teacher reduces the chance to advise students at the same states. The condition of Correct 375 means the teacher advises students only if the student makes a mistake. Under Noteacher condition no teacher is advising the student in the whole time. The Dice 37550 condition indicates that the teacher advises students with a 50% chance if the state importance is satisfied. The result suggests that the performance of students taught by Memory 375 is close to students taught by Correct 375. Students taught by Dice 375 is barely better than study alone students.

The results indicate that state importance thresholds of teaching methods affect RL students’ performance. The results of Figure 7.2 and 7.3 indicate using a threshold of 375 for methods with a decreasing chance in suggesting on same states can reduce the gap between the proposed algorithms. The memory advice algorithm with a state importance threshold of 375 outperformed the mistake correcting method with the same threshold. The results also show that when RL students encountered in similar states, reduce the chance in suggesting will save the suggestion-budget.
7.3.2 Results of Approximate function

In this subsection, the author presents the results of experimenting parameters with the approximate SARSA algorithm in training an RL teacher. The metrics of the experiments is the mean score of students under test mode, in which the teacher is not allowed to suggest, and the students do not update their weight vectors. The author tested effects under four conditions. *Singlereward* means the reward for teaching activities contains one aspect, in this case, which is the reward the student received after taking actions. *Singlereward-five* means the SVM takes the five beginning episodes of each student as training dataset. The author then tested reward shapes from multiple aspects. In the 3MOOne-five-0.1e-5 and 4MOOne-five-0.1e-5 conditions, the rewards are formed by multiple aspects. The number 3 and 4 indicates the number of aspects. Learning rates of 3MOOne-five-0.1e-5 and 4MOOne-five-0.1e-5 are $10^{-5}$, while the learning rate of other conditions are $10^{-2}$.

The results of 7.4 indicate that students taught by *singlereward* barely performed better than the students trained without teacher. And students taught by *singlereward-five* performs better than students taught by *singlereward*. A reason could be the training dataset
for the singlereward is less than the singlereward-five, which is students’ activities in one episode. Students taught by the 3MOOne-five-0.1e-5 and the 4MOOne-five-0.1e-5 performs better than students taught by singlereward-five. A reason is that rewards considers multiple aspects help the teacher to learn better. Similarly, 4MOOne-five-0.1e-5 performs better than 3MOOne-five-0.1e-5 because 4MOOne-five-0.1e-5 has one more aspect.

Figure 7.4 Comparison of RL teacher training strategies

The author then experimented teaching effects under decay learning rates. In this case, the author experimented decay rate of 0.01 and 10⁻⁵. Learning rate with a decay rate decreases a fraction, which is the decay rate, of the learning rate itself each time when RL teacher updates its learning to teach weight vectors. Both conditions use students’ activities in the previous five episodes to train the SVM to predict students’ next action. The reward function of the two conditions is identical to the definition of 4MOOne-five-0.1e-5. The condition with a decay learning rate of 0.01 is named as 4MOOne-five-drop-0.01, and the other one is named as 4MOOne-five-drop-0.1e-5. The author then compares the two conditions with 4MOOne-five-0.1e-5, Memory Advice 375, Correct 375, and Noteacher.
conditions. Figure 7.5 shows the comparison of the six conditions. The results indicate students’ performance taught by 4MOOne-five-drop-0.1e-5 teacher is similar to the ones taught by 4MOOne-five-0.1e-5 teacher. Performance of students’ taught by RL teacher with 4MOOne-five-drop-0.01 barely better than Noteacher condition.

![Figure 7.5 Comparison of RL teacher training strategies with different learning rates](image)

**Figure 7.5** Comparison of RL teacher training strategies with different learning rates

A conclusion one can draw from the results is that an RL teacher can learn teaching strategy via training different RL students and can perform teaching activity without manual threshold setting. The teaching strategy considers state importance, the average state importance level of students, and predictions of an SVM. Through a re-evaluation, the algorithm compares the significance of the difference between suggesting to not suggesting decisions with a record of all the previous ones. The re-evaluation method considers the
significance of each advice decision, and only advise student when the two q-values are significantly different. An advantage of the re-evaluation is that it helps to allow the RL teacher to stay more extended episodes with RL students. The disadvantage of the method is that as the records accumulate the RL teacher tends to suggest at higher importance states as the teacher stays longer with the students. Since a teacher stays alive longer tends to help the student perform better (Fachantidis et al., 2018). So that if an RL student explores some lower state importance states, which the RL teacher may advise at early episodes, but will not do that at later episodes.

7.3.3 Results of ANN

In this subsection, the author presents the experimental results of comparing an RL teacher teaching outcomes under six conditions. Teaching outcome here is represented by the average score of RL students advised by the predictions of ANNs. The conditions are different settings on training artificial neural networks (ANN) by using the teaching activities of mistake correcting teacher (Taylor et al., 2014) at state importance of 375 (Correct 375). The author trained ANNs with various sizes of training datasets. Two ANNs were trained with the dataset from five RL students that were advised by Correct 375. Four ANNs were trained with dataset collected from twenty RL students were suggested by Correct 375. Each RL student performs 400 episodes and each episode has 100 trials. The author compared the teaching outcome of ANNs that were trained with regularization and without regularization. The author also examined the teaching outcome of ANNs trained with a moving threshold. Predictions of SVM is not used in training the ANNs. Condition 20stu represents the ANN is trained by 20 RL students without regularization, and 20stu-reg means the ANN is regularized. Condition 20stu-mvth means the output is re-estimated by the moving threshold method (Zhou and Liu, 2006) without regularization, while 20stu-mvth-reg means the ANN is regularized. 5student means the ANN is trained by a dataset of five RL students’ activities.
And $5_{student-reg}$ means the ANN is regularized. Regularization is considered as a condition because the training dataset is extremely imbalanced.

Results of experiments are shown in Figure 7.6. The results indicate that, in the 20 RL student as training dataset case, RL students advised by the ANNs trained with regularization perform better than those advised by ANNs trained without regularization. For instance, students advised by $20stu-reg$ perform better than those advised by $20-stu$, and students advised by $20stu-mvth-reg$ perform better than those suggested by $20-stu-mvth$. The regularization advantage does not show in the five students as training dataset case. Students advised by $5_{student-reg}$ perform similar to those advised by $5_{student}$. Students advised by $20stu-reg$ perform barely better than those advised by $20stu-mvth-reg$. Students suggested by $20stu-mvth-reg$ achieve higher scores than those advised by $20stu-reg$.

Conclusions the author can draw here is that as the training dataset is extremely imbalanced, the size of the dataset does not affect the outcome very much. Regularization, in this case, is a dropout rate of 0.75 and 0.5 at the second and third hidden layers, changes more on the big dataset (20 RL students) than small dataset (5 RL students), even though the imbalance ratio of the sizes of the two classes are similar. The author experimented ANN with deeper hidden layers, which leads to poor performance that was caused by a vanishing gradient as the size of the input feature is very small. The moving threshold method (Zhou and Liu, 2006) performs a normalization to the output of linear function results of the second last hidden layer multiply with the cost of misclassification, in other words, it replaces the sigmoid function with a weighted normalization function. Zhou and Liu (2006) reported the moving threshold method outperformed other methods in dealing with high imbalanced training datasets. However, the moving threshold method performs similarly to regular sigmoid functions. Sample augmentation is another way many studies put efforts in to solve imbalanced dataset. However, as the training data used in this case is generated by nonlinear teaching activities (Correct 375 only suggests when the student announced a wrong action and the state importance is greater or equal to 375). Besides, RL students’ activities
are determined by linear functions (Q-function is a sum product of its weight vector and feature vector). Therefore, traditional augmentation method is not a good fit in this case.

![Comparison of learning curves of RL teacher training strategies](image)

**Figure 7.6** Comparison of learning curves of RL teacher training strategies

### 7.4 Conclusion and limits

In this chapter, the author presents three types of methods for teaching an RL agent to become a teacher. The first type has two approaches, one is named Dice Advice Algorithm, and the other is titled Memory Advice Algorithm. The Dice Advice method intuitively decides when to suggest by the setting of a fixed state importance threshold and a fixed threshold for randomly generated dice values. The memory advice method makes suggestion decisions by a configuration of a stationary state importance value and a dynamic threshold...
for randomly generated dice values. The history of students encountered states decides the
dynamic threshold. If the RL student meets a stat frequently, the threshold value drifts
up to lower the chance to suggest an action. With this type method, the author examined
parameter effects on RL students’ learning curves.

The second type of method is approximate SARSA. The algorithm takes five features
every time to make the suggestion delivery decisions. The feature set consists of the impor-
tance of states the student encountered, the approximate importance of stats of the student,
the accuracy rate of the student activity prediction of an SVM, the square of the importance
of states, and the results of multiplication of the importance of stats and the approximate
importance of stats. The reward for each teaching action comprises four aspects: the credits
on giving suggestions differ from student’s action, the credits on saving suggestion budget,
the credits on suggesting at the proper time, and the credit on student’s post-activity reward.
The reward assigned to update Q function is the mean value of the four aspects.

The differences of the current work on approximate SARSA from studies conducted
by Fachantidis et al. (2018) are in threefold. They are the configuration of the feature set on
computing actions of making teaching decisions, the reward function, and Q-value evaluation
functions. In the current study, the author does not consider the features proposed by
Fachantidis et al. (2018): they are the teacher’s intended action on performing the task as
a feature on making teaching decision, the suggestion budgets, and the student’s proposed
activities. Instead, the author uses state importance as one features. The author also uses the
accuracy rate of an SVM’s prediction (Torrey and Taylor, 2013) as a feature, since it assists
the teacher to predict student’s future activities. Then the author uses value ($AppStuGap$)
to approximate the student’s progress. The author then uses a function to re-evaluate the
decision of giving suggestions. In the reward function part, the author considers four aspects:
the difference between RL teacher’s action and student’s intended action, saving suggestion
budgets, advising at a proper time, and student’s post-activity reward. The reward shape
is different from (Fachantidis et al., 2018) in that they use the Q-value difference between
the teacher’s intended action and student performed action as a reward for the teacher’s teaching activities.

The third type method is to train an artificial neural network (ANN) to learn to make decisions on advising. The author trained the ANN with features at each step of RL students, and the class label is the decision made by mistake correcting teacher (Taylor et al., 2014). The feature set includes the importance of states the student encountered, the approximate importance of stats of the student, rewards the student received from the last step, square of the importance of stats, and results of multiplication of the importance of stats and the approximate importance of stats. Training data was collected when a mistake correcting teacher is teaching RL students with a threshold of state importance as 375. During training the ANN, the author encountered imbalanced data problem, the author applied moving threshold method (Zhou and Liu, 2006) to handle the problem.

In the approximate SARSA teacher training experiment and the ANN experiment, the author observed the instability of the accuracy rate of the SVM in predicting future steps of the RL students. The author compared results with and without the SVM prediction feature and found out that the results are similar, which indicate the SVM prediction can be ignored. A reason behind the unstable prediction rate can be the nonlinearity of students’ activities as an RL student can perform either the action with the highest Q-value or random action. In the experiment, the RL student has 1% chance to perform a random activity, but that affects the SVM model training as well as its prediction.

The results of this chapter are threefold. The first part is the parameter effect examination on RL student learning curves. The author aims to find a threshold that minimizes the gap between the proposed methods and the mistake correction method. The results show the smallest gap between a dice advice teacher and the mistake correcting teacher is around state importance of 375 through 425 — the author selected 375 as the importance threshold to have the RL teacher to advise more frequently. Learning curves of 40 RL students taught by the memory advice teacher with importance threshold perform similarly to the learning
curves of 40 RL students taught by mistake correcting teacher.

The results of 40 RL students taught by approximate the SARSA teacher show that with a learning rate dropping to $10^{-5}$ and predictions of RL students’ activity from an SVM, the average learning curve of the 40 RL students outperforms the students taught by approximate SARSA teacher with other settings. A reason is that a decreased learning rate tends to promote Q function coverage. In this case, the drop rate is compatible with the teacher’s pace in learning how to teach. The learning curves of RL students taught by the approximate SARSA teacher is similar to the students taught by the dice advice teacher with state importance threshold equal to 375.

The teaching outcomes of trained ANNs show RL students trained by an ANN trained with a mistake correcting teacher advises 20 RL students and with regularization during training outperformed students trained by other ANN teachers. When compared with the other teachers, the students’ performance is similar to the students tutored by approximate SARSA teacher and by dice advice teacher with state importance threshold equals 375.

Contributions the author made in this chapter include 1) examined state importance threshold for RL students’ performance, 2) proposed of two intuitive methods to advise RL students, 3) investigated using approximate SARSA method to train RL agent to learn to teach, and 4) using an artificial neural network (ANN) to learn from mistake correcting teacher to advise RL students.

There are limitations to the work presented in this chapter. The author selected state importance threshold as 375 and experimented it with memory advice method. The experiment could to using other state importance values as the threshold for the memory advice method. The reward of approximate SARSA could consider the improvement made by RL students, such as the one proposed by Omidshafiei et al. (2018) that use the change of students’ q-values to increase or decrease the value of rewards. The work of teaching with ANN can be extended to train another RL agent to learn how to use the output of an ANN to teach RL students. The training dataset for the ANN can consider other state importance,
as the author’s observation, the state importance the ANN decides to advise is fluctuate around 375 in this study.
In this chapter, the author presents the work of training an artificial intelligent (AI) teacher by the Reinforcement Learning (RL) method proposed in the previous chapter (Chapter 7). The author then adds it to the SHiB ITS as a tutor and evaluate its teaching effectiveness. To train such a teacher, the author uses RL students to help the teacher to gain teaching experiences. The teaching contents are the training exercise 3 of the Smart Home in a Box (SHiB) ITS. Fifteen participants were recruited from Amazon’s Mechanical Turk. The author evaluated the RL teacher’s teaching effectiveness via participants’ post-test performance and their feedbacks in the post-survey.

The author described the reasons for performing this study for the purpose of the study section. Training the RL teacher and recruiting participants are included in the method section. The results section include the RL students’ performance and the participants’ performance. The final part consists of the conclusion and limits of the chapter.
8.1 Purpose of study

According to the results of Chapter 5, ITSs that apply trial-and-error and combination teaching strategies are suitable for elderly (≥ 50 years old) and young (< 50 year old) adults. Both teaching strategies require the system to respond to mistakes made by users. The current tutoring system takes actions after users had placed all the devices. If a learner has little experience, the process of correcting errors may make the learner frustrated in adjusting minor mistakes.

An experienced human teacher who has taught many students can generalize critical problems that most students will have trouble with. Similarly, if an ITS can simulate such feature, it will be able to 1) only suggest a specific time to avoid making learners annoying and 2) help learners to understand critical problems and learn more about them. To fulfill these features, an ITS should be able to 1) advise the user before making mistakes and 2) know when to be quiet and allow the student to experience some trials.

Any experienced teacher needs a process of gaining experiences. Human teachers obtained experience through teaching activities. Similarly, an AI agent can learn teaching experiences through interactions with human students. However, it requires a large number of participants. Instead of recruiting participants to teach a computer to become a teacher, the computer can learn how to teach by itself with machine learning methods. Students can be computers that have no experience in specific tasks, and the teacher is another computer who has learned the task in advance. The teacher can teach as many students as possible to develop a teaching strategy so that the agent knows when to suggest.

The author has investigated three types of methods to train an agent to become a teacher in Chapter 7. Considering the limits of computation power of web applications, and different state importance values, since the SHiB ITS training exercise 3 is different from Pac-Man game, the author decides to apply the approximate SARSA as the training method to prepare the SHiB ITS teacher. In the next section, the author presents the details of parameter
settings and participants recruitment.

8.2 Methods

In this section, the author presents methods of training an RL agent to be a teacher of the SHiB ITS training exercise 3. Then the author presents methods of recruiting participants, and evaluating the RL agent teaching outcomes.

8.2.1 Feature construction, Q-value re-evaluation, and reward shaping

Approximate SARSA (Sutton and Barto, 2018) is an online reinforcement learning methods. When an agent is trained, a Q function is maintained and updated with a weight vector, a feature vector, and a bias. The corresponding equations are Equation 7.1 and Equation 7.2. As an extension work of chapter 7, the author adopts feature settings and rewards from the methods described in chapter 7. Considering the computation power limit of web applications and unstable accuracy rate in prediction, the author decides to remove the feature of prediction accuracy of the support vector machine (SVM) that predicts students’ activities. The features that are adopted in this study are:

\[ x_1 = \text{Sigmoid}(TSIM - \text{midgap}) \]

\[ x_2 = \text{Sigmoid}(\text{AppStuGap} - \text{midAppStuGap}) \]

\[ x_3 = x_1^2 \]

\[ x_4 = x_1 \cdot x_2 \]

where, the \text{Sigmoid} is the logistic function, \text{Sigmoid}(x) = \frac{1}{1+\exp(x)}, which return a value in \([0, 1]\). Teacher state importance (TSIM) is defined in Equation 7.4. \text{AppStuGap} is the approximate student gap in the teacher’s perspective. The equation for \text{AppStuGap} is defined in Equation 7.5. Teaching activity re-evaluation method (Algorithm 3) is adopted in this section. An adjustment the author makes in this chapter is that the threshold of making decision
from using the $Q_{a=1}^* = Q_{a=1} \cdot (1.0 - p_t)$, is changed to using the “three sigma” rules (Kriegel et al., 2009) where $\sigma(68.2\%)$ as threshold ($Action = 1$ if $prob > 68.2\%$, Otherwise $Action = 0$). The reason for making such adjustment is that as the author observes the teaching activities of the RL teacher, the original setting hindered almost all suggesting decisions. The comparison also changed from using current Q-value difference ($d_t$) comparing with previous Q-values difference ($d_{t-1}$) to compares with the median ($\bar{D}$) of the Q-value difference records.

The reward function of this section is similar to the one presented in chapter 7. In this study, the author replaced $r_4 = R_{t+1,\text{student}}$ with $r_4 = 1(\hat{V}_{\text{student}} > V_{\text{student}})$, the $\hat{V}$ represents the student’s Q value after adopting the RL teacher’s suggestion. If the Q value is greater than the old one, then the teacher will receive a reward. The method is similar to (Omidshafiei et al., 2018). They suggest using untutored student’s Q value as a threshold to compare with the student’s Q values when the student applies the teacher’s suggestion. The suggestion budget is set as 30 for each single learning task. RL student follows five steps of actions the RL teacher suggests. During test mode, the RL teacher is not allowed to suggest the RL student, and the RL student will not update its Q function in the test mode.

### 8.2.2 Recruiment and teaching outcome evaluation metrics

Participants are recruited from Amazon’s Mechanical Turk. Participants were not acquainted with the contents of the SHiB ITS. Participants were assigned to the RL Teacher group. Adult participants are qualified for the experiment; no age constraints set for this experiment. The entire process will take around 45 minutes. Participants who complete the whole process (a pre-test, three training exercises, and four post-tests) will receive a $2.00 compensation. If a participant placed 90% devices correctly in the post-tests, he or she would receive an additional $5.00 as a bonus, for a total of $7.00.

The RL teacher group has three training exercises. The training exercise 1 and 2 use the Combination 2 teaching strategy. The contents of the two training exercises are the same
as the ones in other groups. Chapter 4 contains detailed materials of training exercise 1 and 2. Layout and devices of training exercise 3 are similar to exercise 2. The teaching strategy of training exercise 3 is to use an RL agent as a teacher to give participants’ advice after they picked a device and before they put it down. Once the RL teacher provides a piece of advice, the layout of the training exercise will highlight the next five steps. An illustration is shown in Figure 8.1. The highlighted green blocks in the figure are the suggested path from the RL teacher. Participants have to take four post-tests to have their task completion. Performance of participants is measured as the percentage of their correctly placed devices over all devices are given in each post-test.

![Figure 8.1 Demo of Group 6 training exercise 3](image)

Teaching effectiveness of the RL teacher is evaluated by participants’ performance in post-tests and their feedbacks to the post-survey. Participants’ post-test performance include their accuracy rate of placing devices and the steps they took to put the devices. The contents of the post-survey are identical to the one introduced in Chapter 4. The survey asks questions on seven aspects of the SHiB ITS: complexity of the ITS (Aspect 1); ease of use of the ITS (Aspect 2); function integration of the ITS (Aspect 3); quick to learn to use of the ITS (Aspect 4); confidence of using the ITS (Aspect 5); ease of learning multiple solutions (Aspect 6); and participants subjective feeling about understanding the teaching contents.
(Aspect 7). Aspects 1 through 5 are combinations of the first ten survey questions, that is aspect 1 combined results of questions 1 and 2; aspect 2 combined results of questions 3 and 4, and so on. The even-numbered questions (2,4,6,8,10) were reversed coded.

8.3 Results

In this section, the author presents the results of training RL teacher for SHiB ITS training task 3 and participants performance in post-tests with an RL teacher available for the training task 3. Post-survey results include a comparison of six groups of young adult participants’ grades on the seven aspects of the post-survey. As a reminder, group 1 are taught by trial-and-error teaching strategy, group 2 are trained by the textbook I teaching strategy, the combination I teaching strategy teaches group 3, group 4 are taught by the textbook II teaching strategy, group 5 are guided by the combination II teaching strategy, and group 6 are taught by the RL teacher.

8.3.1 Performance of RL student

Approximate SARSA algorithm is used to train RL students in this chapter. The student has a feature set \( F = \{f_1, f_2\} \) with two features, \( f_1 \) is the portion of correct spots in the area around it, which is 25 grids in this chapter; the \( f_2 \) is the mean value reciprocals of Euclidean distance apart from the student to correct spots. \( f_1 \) simulates the visual function, while \( f_2 \) simulates the olfactory function. The knowledge of the RL teacher is obtained by using an on-policy TD(0) SARSA algorithm (Sutton and Barto, 2018) to learn each devices’ correct spots. The update Q function equation is shown in Equation 7.1.

Figure 8.2 shows the performance of 40 RL students in the SHiB ITS training exercise 3. The author presents three conditions: RL student’s test mode performance with RL teacher in training, RL student’s test mode performance with no RL teacher during the training process, and RL student’s performance in training with RL teacher. Under test mode, the
RL students do not update their Q function nor do they receive advice. The purpose of including the RL student’s training performance is to show the difference with RL teacher and without RL teacher. A one-way ANOVA is conducted to compare the difference between RL students’ performance in the test modes. The result shows that the RL students that had an RL teacher in training perform significantly \( p < 2.2e^{-16}, g = 12.84 \) better than the RL students trained by themselves. The training performance is better than the performance in test mode, which indicates that the RL teacher helps the RL student to find correct spots during training. However, the RL student performs similarly across 100 training episodes, which suggests that the student does not learn very much during the training episodes. One reason can be that the learning method the RL student uses is approximate SARSA, which is more suitable for tasks that have a dynamic environment, while the SHiB ITS training exercise 3 is a static task, which may hinder the RL student keeps on learning. The reward function of the SHiB ITS training exercise 3 does not always give reward signals, and it only gives a reward once the student can find a correct location.

![Figure 8.2 RL student performance in SHiB ITS training exercise 3](image)

The author then applied a tabular SARSA learning method to an agent to perform as
an RL student to learn how to place a device. The RL teacher has gained knowledge in advance. The RL teacher is under training to teach by the approaches the author proposed in this chapter. An RL teacher teaches 40 RL students. Each student performs 100 trials (episodes), in between the student learns by performing 10 training exercises. During the trial task, the student does not update its Q functions nor is the teacher available. The RL student updates its Q functions in each training exercise. The RL teacher is available in the training exercises. Suggestion budget is set as 20 for each trial after the teacher advises a suggestion, the student will follow 5 sequential steps of the teacher’s actions. The followed steps are counted as suggestions. If the RL teacher advises the RL student during an exercise, then a Q-values re-estimation is performed for the RL student when the exercise ends. The re-estimation is to re-calculate the Q values of the states the RL student has encountered during the training exercise, from the final state to its start state by SARSA updating method.

Performance of placing sensors in SHiB ITS training exercise 3 is compared between RL students studied alone and the ones who studied with an RL teacher who is learning to teach. At each trial, an RL student starts at a random place. The rewards are the mean value of 40 students’ rewards at each trial. One-way ANOVA is conducted to compare the difference. Figure 8.3 shows a comparison of the performance of 40 RL students with and without a teacher in placing a server in SHiB ITS training exercise 3. The results show that the RL students with an RL teacher’s advice significantly \( p < 0.01, g = 0.504 \) outperformed the students who learned alone.

Figure 8.4 show the performance comparisons of placing six room sensors. Room 1 is the left side bedroom, and room 2 is the bedroom at right. The results show that the RL students placing the room 1 door sensor (Figure 8.4c) with an RL teacher’s advice are significantly \( p < 0.01, g = 0.422 \) better than those learned alone. For the rest five room sensors, the RL students tutored by the RL teacher performed better than those with no RL teacher, but the results are not significant. In particular, the performances are similar
for placing sensors in room 2. One reason is that the location of room 2 in the layout is not easily accessible, RL agents have to find the entry of a corridor that connects to room 2, and then find the room’s entrance. That also causes room 2 learning curves are not increasing obviously, which require more episodes to learn the tasks better.

Figure 8.5 show performance comparisons of RL students in placing kitchen sensors. The shape of the kitchen is an irregular polygon, which increases the difficulty for the RL agents to explore, as there are locations only two actions will move to other places. It takes a long time for the RL agent to disregard the actions that stay at the same locations. The RL students received the RL teacher’s advice significantly outperformed the other group in placing area sensors in the kitchen ($p < 0.01, g = 0.376$).

Figure 8.6 show the comparisons of RL students placing relays in room 1, room 2, and the kitchen. The average reward of placing relays is the highest among all types of devices, which indicate that, for the RL agents, the tasks are easy to learn. RL students learned with an RL teacher performed better than those with no help during training, but the results are not significant.

A conclusion can be drawn from the results is that the RL teacher can learn how to teach
(a) Room 1 bed sensor   (b) Room 1 area sensor   (c) Room 1 door sensor

Figure 8.4 Tabular SARSA student performance comparison in placing room sensor in SHiB ITS training exercise 3

(d) Room 2 bed sensor   (e) Room 2 area sensor   (f) Room 2 door sensor

(a) Kitchen sink sensor   (b) Kitchen area sensor   (c) Kitchen refrigerator

Figure 8.5 Tabular SARSA student performance comparison in placing kitchen sensors in SHiB ITS training exercise 3
via an approximate SARSA approach. The learning algorithms of the RL teacher and the RL students do not have to be the same; however, the task should be similar. Although the RL teacher advises 30 suggestions in each training task, that is much less than the steps, which is set as 150 steps in each training task, the RL student can correctly place many devices in each task. The improvement the approximate SARSA RL students made in the test mode shows that the RL teacher helps the students to perform better.

The RL teacher also learns to helps tabular SARSA students perform 13 placing sensor tasks. As the teacher is also learning to teach, it is reasonable that at the start of the episodes, tabular SARSA students perform not as good as those with no help. As the teacher gains more experience better times, the teacher advises, so that the RL students perform better at later stages of the episodes in most of the tasks. The author notices that the teaching effectiveness is evident if the training task is in moderate difficulty, in which, the RL student can progress gradually in 100 episodes, such as the server task and the room 1 tasks. Hard tasks require the RL students to spend more time to learn, which makes the learning curve grow vaguely, such as the kitchen tasks and room 2 tasks. Tabular RL agents learn easy tasks (relays tasks) quickly, as the teacher is also under training; therefore, the RL students with a teacher in training perform worse than those studied alone at the beginning episodes.
8.3.2 Performance of participants

Fifteen participants (\(N = 15\)) were recruited from Amazon’s Mechanical Turk. Their age ranges from 25 years old to 45 years old (\(mean = 32.73\)). 13% of the participants are African Americans, 40% of them are Asians, and 47% of them are Caucasians. Geographically, 13% of them live in Europe, 40% of them live in Asia, and 47% of them live in North America. 80% of them speak English as a native language the rest of them speak other languages.

Performance of the recruited participants is compared with the performance of young adults in Exploration group, Textbook 1 group, and Textbook 2 group, and the performance of participants in Textbook 2 group and Combination 2 group. The SHiB ITS performs teaching strategy in the Exploration group. Then the ITS performs following worked example teaching strategy in the Textbook 1 and Textbook 2 group, Textbook 2 has answer validation function during training state. In Combination 1 and Combination 2 group, the ITS performs following worked example and trial-and-error teaching strategies as a combination, Combination 2 group does not have answer validation function. Finally, in the RL teacher group, the ITS advises the users before they were placing devices. Table 8.1 lists the statistics of their performance. A one-way ANOVA is conducted to compare the performance of RL teacher group with other groups. The performance of RL teacher groups significantly better than participants in the Textbook 2 group in post-test 2 (\(g = 0.732\)) and post-test 3 (\(g = 0.876\)).

<table>
<thead>
<tr>
<th></th>
<th>post-test 1</th>
<th>post-test 2</th>
<th>post-test 3</th>
<th>post-test 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(N) mean std</td>
<td>(N) mean std</td>
<td>(N) mean std</td>
<td>(N) mean std</td>
</tr>
<tr>
<td>Exploration</td>
<td>17 0.98 0.08 8.25</td>
<td>17 0.95 0.09 9.18</td>
<td>17 0.78 0.31 39.20</td>
<td>17 0.95 0.07 7.19</td>
</tr>
<tr>
<td>Textbook 1</td>
<td>15 0.91 0.20 21.70</td>
<td>15 0.88 0.18 20.20</td>
<td>15 0.88 0.20 22.40</td>
<td>15 0.94 0.18 18.90</td>
</tr>
<tr>
<td>Combination 1</td>
<td>16 1.00 0.00 0.00</td>
<td>16 0.99 0.05 5.06</td>
<td>16 0.91 0.21 23.30</td>
<td>16 0.95 0.09 9.58</td>
</tr>
<tr>
<td>Textbook 2</td>
<td>15 0.96 0.17 18.00</td>
<td>15 0.79 0.28 35.50</td>
<td>15 0.59 0.33 56.10</td>
<td>15 0.80 0.30 37.10</td>
</tr>
<tr>
<td>Combination 2</td>
<td>14 1.00 0.00 0.00</td>
<td>14 0.96 0.11 11.40</td>
<td>14 0.76 0.28 36.50</td>
<td>14 0.99 0.03 2.84</td>
</tr>
<tr>
<td>RL teacher</td>
<td>15 0.96 0.17 5.55</td>
<td>15 0.95 0.11 8.99</td>
<td>15 0.85 0.24 3.50</td>
<td>15 0.91 0.12 7.37</td>
</tr>
</tbody>
</table>
A purpose of having an RL teacher in the SHiB ITS is to help users to learn better. Therefore, step lengths which, in this study, represents time participants spent in each task is essential for the efficacy evaluation of the RL teacher. Table 8.2(a) lists the statistics of the number of steps participants consumed in each exercise, and 8.2(b) lists the count of steps in post-tests. The count of steps shows that the RL teacher group participants spent significantly more time than the other five groups in training exercise 3, where the RL teacher is available. As a result, the steps spent by the RL Teacher participants in the post-test 3 is much less than the participants in other groups.

Table 8.2 Step length comparison (a) training exercises, (b) post-tests

(a)

<table>
<thead>
<tr>
<th></th>
<th>Training exercise 1</th>
<th>Training exercise 2</th>
<th>Training exercise 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>mean</td>
<td>std</td>
</tr>
<tr>
<td>Exploration</td>
<td>17</td>
<td>212.24</td>
<td>69.41</td>
</tr>
<tr>
<td>Textbook 1</td>
<td>15</td>
<td>108.40</td>
<td>45.54</td>
</tr>
<tr>
<td>Combination 1</td>
<td>16</td>
<td>138.47</td>
<td>56.67</td>
</tr>
<tr>
<td>Textbook 2</td>
<td>15</td>
<td>149.13</td>
<td>53.00</td>
</tr>
<tr>
<td>Combination 2</td>
<td>14</td>
<td>114.27</td>
<td>58.06</td>
</tr>
<tr>
<td>RL Teacher</td>
<td>15</td>
<td>121.87</td>
<td>81.53</td>
</tr>
</tbody>
</table>

(b)

<table>
<thead>
<tr>
<th></th>
<th>post-test 1</th>
<th>post-test 2</th>
<th>post-test 3</th>
<th>post-test 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
<td>mean</td>
<td>std</td>
<td>vs</td>
</tr>
<tr>
<td>Exploration</td>
<td>17</td>
<td>171.00</td>
<td>23.08</td>
<td>7.41</td>
</tr>
<tr>
<td>Textbook 1</td>
<td>15</td>
<td>226.20</td>
<td>122.99</td>
<td>1.84</td>
</tr>
<tr>
<td>Combination 1</td>
<td>16</td>
<td>188.87</td>
<td>40.80</td>
<td>4.63</td>
</tr>
<tr>
<td>Textbook 2</td>
<td>15</td>
<td>204.00</td>
<td>68.81</td>
<td>2.96</td>
</tr>
<tr>
<td>Combination 2</td>
<td>14</td>
<td>197.40</td>
<td>48.00</td>
<td>4.11</td>
</tr>
<tr>
<td>RL Teacher</td>
<td>15</td>
<td>230.47</td>
<td>60.12</td>
<td>3.83</td>
</tr>
</tbody>
</table>

Figure 8.7 shows the mean steps participants spent in the three training exercises. Factors that affect step length include task complexity and the efficacy of teaching strategies. Training exercise 1 is the most straightforward task among the three training exercises, therefore, the time participants spent in it is much less than what they spent in training exercise 2 and 3. Exploration group spent the most time in training exercise 1, as the trial-and-error teach-
ing method force participants to figure out the correct spots themselves. The same factors affect the steps in training exercise 2. In training exercise 3, participants in the RL Teacher group spent the most time to finish the task. As the RL teacher does not tell where the correct spots are for the participants, and it only gives hints when it thinks the participants perform poorly. Therefore, participants need to spend more time to figure out the correct spots with limited information. Another reason is a participant spent over 4,000 steps in the task, which causes the rise of step length.

![Figure 8.7](image.jpg)

**Figure 8.7** Comparision of number of steps participants spent in each training exercises

Participants spent less time in post-tests due to two reasons: one is the participant understands the teaching contents very well so that they can finish the tests quicker, the other reason is that participants feel the task is complicated so that they give up early. Therefore, it is necessary to combine participants post-tests’ performance to analyze the reason behind step lengths in post-tests. The results indicate that the complexity of post-test 1 is simple for the participants. The post-test 2 is easy for participants in Exploration, Combination 1, and
Combination 2 groups. Participants in RL teacher group spent a long time to achieve decent performance. Participants in Textbook 2 do not want to spend too much efforts to improve their performance. Post-test 3 is the most complex task, participants in Combination 1 and Textbook 1 spent much time (mean=1027.13 steps and mean=950.40 steps, respectively) to achieve a decent performance, 91% and 88%, respectively. Participants in the RL teacher group spent less time (mean=709.33 steps) but achieved an 85%. Exploration group and Combination 2 group spent more efforts (mean=858.59 steps and mean=870.13 steps, respectively) than RL teacher group, but achieve 78% and 76%. Textbook 2 gives up on fixing their answer. Post-test 4 has 19 devices to place, which takes the longest step length when compared with other tests; the number of steps participants spent in the test is more than 1,500.

The author then presents the statistics of results of the post-survey in Table 8.3. A one-way ANOVA test was performed to analyze the differences across six groups. vs denotes the coefficient of variation (CV). Participants in the RL Teacher group rate significantly higher than Textbook 2 in aspect the aspect of the functional integration of the ITS (aspect 3) (g = 0.744). On the aspect of quickly learn to use the ITS (Aspect 4), participants in the RL Teacher group graded significantly higher score than group Exploration, group Combination 1 and Textbook 2 (g = 0.634, g = 0.581, and g = 0.630 respectively). Participants in the Combination 2 group had significantly higher scores on the functional integration of the ITS (Aspect 3) than the Combination 1 and Textbook 2 conditions (g = 0.593 and g = 0.669 respectively). Participants in the Combination 2 condition indicated they had significantly higher confidence (Aspect 5) in using the ITS to learn than participants in the exploration condition (g = 0.615). Finally, participants in the Exploration and Textbook 2 groups indicated they had a significantly better understanding of the teaching contents (Aspect 7) of the ITS than the Combination 2 group (g = 0.998 and g = 0.918).

The results indicate that teaching strategies affect participants’ subjective feeling with the ITS. Training exercises with no answer validation help participants have a positive attitude
Table 8.3 Results of post-survey of six groups (a) results of aspect 1 through 5, (b) results of aspect 6 and 7

(a)

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>mean</th>
<th>std</th>
<th>vs</th>
<th>N</th>
<th>mean</th>
<th>std</th>
<th>vs</th>
<th>N</th>
<th>mean</th>
<th>std</th>
<th>vs</th>
<th>N</th>
<th>mean</th>
<th>std</th>
<th>vs</th>
</tr>
</thead>
<tbody>
<tr>
<td>explore</td>
<td>16</td>
<td>3.143</td>
<td>1.380</td>
<td>0.439</td>
<td>16</td>
<td>3.179</td>
<td>1.188</td>
<td>0.374</td>
<td>16</td>
<td>3.143</td>
<td>1.208</td>
<td>0.385</td>
<td>16</td>
<td>2.964</td>
<td>1.261</td>
<td>0.426</td>
</tr>
<tr>
<td>textbook</td>
<td>13</td>
<td>3.538</td>
<td>1.174</td>
<td>0.332</td>
<td>13</td>
<td>3.692</td>
<td>1.258</td>
<td>0.341</td>
<td>13</td>
<td>3.423</td>
<td>1.391</td>
<td>0.406</td>
<td>13</td>
<td>3.269</td>
<td>1.373</td>
<td>0.42</td>
</tr>
<tr>
<td>combine</td>
<td>14</td>
<td>3.143</td>
<td>1.239</td>
<td>0.394</td>
<td>14</td>
<td>3.214</td>
<td>1.315</td>
<td>0.409</td>
<td>14</td>
<td>3.071</td>
<td>1.245</td>
<td>0.405</td>
<td>14</td>
<td>3.071</td>
<td>1.096</td>
<td>0.354</td>
</tr>
<tr>
<td>textbook II</td>
<td>14</td>
<td>3.115</td>
<td>1.336</td>
<td>0.429</td>
<td>14</td>
<td>3.308</td>
<td>1.436</td>
<td>0.434</td>
<td>14</td>
<td>3.285</td>
<td>1.336</td>
<td>0.463</td>
<td>14</td>
<td>3.077</td>
<td>1.440</td>
<td>0.468</td>
</tr>
<tr>
<td>combine II</td>
<td>15</td>
<td>3.600</td>
<td>1.037</td>
<td>0.288</td>
<td>15</td>
<td>3.533</td>
<td>1.167</td>
<td>0.330</td>
<td>15</td>
<td>3.700</td>
<td>1.088</td>
<td>0.294</td>
<td>15</td>
<td>3.467</td>
<td>1.332</td>
<td>0.384</td>
</tr>
<tr>
<td>RL Teacher</td>
<td>15</td>
<td>3.531</td>
<td>0.763</td>
<td>0.216</td>
<td>15</td>
<td>3.313</td>
<td>0.929</td>
<td>0.280</td>
<td>15</td>
<td>3.594</td>
<td>0.821</td>
<td>0.228</td>
<td>15</td>
<td>3.750</td>
<td>0.775</td>
<td>0.207</td>
</tr>
</tbody>
</table>

(b)

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>mean</th>
<th>std</th>
<th>vs</th>
<th>N</th>
<th>mean</th>
<th>std</th>
<th>vs</th>
</tr>
</thead>
<tbody>
<tr>
<td>explore</td>
<td>16</td>
<td>4.143</td>
<td>0.770</td>
<td>0.186</td>
<td>16</td>
<td>3.500</td>
<td>1.286</td>
<td>0.367</td>
</tr>
<tr>
<td>textbook</td>
<td>13</td>
<td>4.308</td>
<td>0.947</td>
<td>0.220</td>
<td>13</td>
<td>2.769</td>
<td>1.235</td>
<td>0.446</td>
</tr>
<tr>
<td>combine</td>
<td>14</td>
<td>3.571</td>
<td>1.453</td>
<td>0.407</td>
<td>14</td>
<td>2.786</td>
<td>1.424</td>
<td>0.511</td>
</tr>
<tr>
<td>textbook II</td>
<td>13</td>
<td>3.923</td>
<td>1.188</td>
<td>0.303</td>
<td>13</td>
<td>3.385</td>
<td>1.261</td>
<td>0.373</td>
</tr>
<tr>
<td>combine II</td>
<td>15</td>
<td>4.267</td>
<td>0.884</td>
<td>0.207</td>
<td>15</td>
<td>2.200</td>
<td>1.329</td>
<td>0.600</td>
</tr>
<tr>
<td>RL Teacher</td>
<td>15</td>
<td>3.938</td>
<td>0.998</td>
<td>0.253</td>
<td>15</td>
<td>3.375</td>
<td>1.147</td>
<td>0.340</td>
</tr>
</tbody>
</table>

in functional integration (Aspect 3) and have more confidence in using the ITS (Aspect 5). With the assistance of the RL teacher helps participants learn to use the ITS quickly (Aspect 4). Finally, answer validation affects participants feel they understand the teaching contents better (Aspect 7).

8.4 Conclusion and limitations

In this chapter, the author presents the work of training an RL teacher by teaching 40 RL students in performing the SHiB ITS training exercise 3. The author then embedded the RL teacher into the SHiB ITS training exercise 3 and recruited fifteen participants from Amazon’s Mechanical Turk to evaluate the teaching effectiveness of the RL teacher.

The RL teacher itself learned to perform the task by an on-policy TD(0) control (SARSA) RL method to determine the correct spots for each device. The RL student is trained by using an approximate SARSA method, at each state, the agent considers two features to represent
its actions' q-values. Then the author uses an approximate SARSA method considers four features to train the RL teacher to teach the RL students. The results show that RL students' performance improved when the RL teacher is available. The results indicate that even though the RL teacher and RL student trained in different ways, the RL teacher still can learn how to teach, and they can help the RL student to improve performance.

With the trained RL teacher, the author then experimented with the teaching effectiveness of the RL teacher. The results indicate that participants in the *RL teacher* group spent more extended time than the other groups in the training exercise 3. As a result in the post-tests, they outperformed than *Textbook 2* group in post-test 2, and 3. In post-test 3, participants in the *RL teacher* group spent less time but achieved a decent result. The results indicate that using an RL teacher at the training exercise 3, make participants spend more time to figure out the answer, but that benefits their performance in the post-tests. As post-test 3 requires an alternative solution, it requires users to understand how multiple-solutions work. The RL teacher helps participants in understanding the tricks in training exercises.

Contributions the author made in this chapter include: 1) the author presents work to train an RL teacher to teach the SHiB ITS training exercise 3; 2) RL students' performance shows that even though the RL teacher and students learn the knowledge differently, the RL teacher still helps the RL students improve their performance; 3) the author developed an intelligent tutoring system that uses an RL teacher to advise users before placing the devices, and the result show such teaching strategy helps users to understand how multiple-solution works.

Limitations in this chapter include the RL student learning method only considers two features, which make the student not keep on progressing as it experiences more. In the RL teacher training task, the author uses a suggestion budget in each task, which differs from having a suggestion budget setting in the entire training process. In the teaching effectiveness experiment, the author tested the same condition as *Combination 2* in training exercise 1.
and 2, which can explain participants’ performance in post-test 3 but cannot explain why they spent a longer time in post-test 2 than other groups. *RL Teacher* group participants spent a long time in the training exercise 3, which means the frequency the RL teacher advise does not help to ease the process of learning. In the user interface design of training exercise 3, the AI teacher existence declaration only shows at the beginning of the task. Participants may ignore it and are not aware that the highlighted places in the exercise are the hints from the RL teacher. Alternatively, the participants may play with the system try to let the RL teacher tell them where are the correct spots.
Chapter 9

Conclusion and Future work

The author presents a summary of the results and the results of the previous work in this chapter. The author also offers suggestions for future work. The overview is in section 9.1, and the future work suggestions are in section 9.2.

9.1 Summary of presented work

In this dissertation, the author has developed two intelligent tutoring systems that teach multiple solutions, one is FreeCAD ITS, and the other is SHiB ITS. The author examined three teaching strategies (trial-and-error, practicing worked examples, and combination) with the tutoring systems. The experiments gathered participants’ viewpoints of the tutoring systems after post-tests.

The author recruited over 15 college students in the FreeCAD ITS experiment (Chapter 3). The results show that young adults benefit more from the combination teaching strategy than the other two. The combination teaching strategy has learners practicing worked examples and trial-and-error at the training stage.

In the SHiB ITS experiment (Chapter 4), 125 participants were recruited by the author, including young and old adults, 96 of whom completed all the study tasks. The results
of the experiment of the SHiB ITS show elderly participants (over fifty years age) benefit more from the trial-and-error teaching strategy, while the younger adults benefit more from the combination teaching strategy. Older adults in the trial-and-error group performed significantly better than the other two teaching strategies groups. The result is a contribution to the teaching strategy comparison for older adults as no significant differences have been found in previous studies (Ribeiro and de Barros, 2014; Struve and Wandke, 2009). The results of younger participants in the SHiB ITS experiment are consistent with the results of the FreeCAD ITS test.

The post-tests survey of the two experiments found that older and younger adults were interested in studying multiple solutions during the training period. The elderly participants of the SHiB ITS ranked practicing worked example higher than the trial-and-error, which is in agreement with the results of Leung et al. (2012).

The author has investigated the actual installation taught by a written document and the SHiB ITS teaching strategy combination (Chapter 5). The author recruited 20 participants, 10 for each group. The participants were not familiar with the contents of the experiment. The contents of the written document are identical to the contents of the SHiB ITS teaching strategy practicing worked examples. The experimental results show that participants trained by the SHiB ITS have spent significantly shorter time installing. They also had significantly greater self-confidence in the performance of the installation. The results suggest that SHiB ITS is feasible for older adults, which saves their time and improves their confidence in installing.

The author has carried out a further analysis of the data gathered from the SHiB ITS experiment (Chapter 6). The study includes a Principal Component Analysis (PCA), and a Hidden Markov Model (HMM) mixed with the normal distribution. The PCA results show that participants have similar viewpoints of server positions, but the views of relays’ positions in the post-tests are different from one another. A factor that causes various relay placements is the requirement to use multiple solutions to relays’ installation locations in the
post-tests. Besides, the results of the HMM mixed with Normal Distribution show that the accuracy of the subsequent behavior prediction is 58.68% with three normal distributions. The number of normal distributions matches the experiment group settings. The use of mixed HMM to predict subsequent behavior is limited to an off-line prediction, which also requires a greater size of the dataset than that of the SHiB online experiment (Chapter 4) for model training.

The author then investigated methods to train a Reinforcement Learning (RL) agent to teach another RL agent (Chapter 7) to complete a task. The author explored three types of methods in generating teaching strategies. Two intuitive methods, Advice Dice and Memory Advice are developed to advise RL students in specific situations, named state importance. Through the experiment, the author found at a certain level of state importance, RL students advised by an intuitive RL teacher perform similarly to those tutored by correct mistake RL teacher (Taylor et al., 2014).

Then the author explored features that can be used to train an RL agent to generate teaching strategy via approximate SARSA algorithm (Sutton and Barto, 2018). Fachantidis et al. (2018) also studied features that can be used in on-policy and off-policy approximate RL algorithms, they characterized features as the teacher’s intended actions on performing the task, suggestion budgets, and the student’s proposed actions. Instead, the author portrays the feature vector as the state importance, the accuracy of SVM prediction (Torrey and Taylor, 2013) on students’ future activities and an approximation of students’ progress. The author then added a function using a probability density function (PDF) to re-evaluate teaching decisions. Fachantidis et al. (2018) uses the gap between the teacher’s action Q-value and the student’s action Q-value as a reward. The author considers four aspects to reward teaching activities, and they are advising a better action than the student’s action, saving suggestion budgets, advising at a proper time, and student’s post-activity reward.

The author then investigates using an artificial neural network (ANN) to learn the teaching activity of the correct mistake teacher (Taylor et al., 2014) at a particular state impor-
stance threshold. The author noticed the training dataset is extremely imbalanced so that the author experimented with the teaching effects of conditions. The author found that with an extremely imbalanced dataset, the size of the dataset does not affect the teaching outcome. However, the regularization affects more on the bigger dataset. A previous study (Zhou and Liu, 2006) suggests using a moving threshold to deal with imbalanced data, but it has limitations when dealing with extremely imbalanced data.

With consideration of the computational limits of the web application and the generality of the training methods, the author decided to use the approximate SARSA to train an RL agent to teach another RL agent to perform the SHiB ITS training exercise 3 (Chapter 8). The author trained the RL teacher by an on-policy TD(0) control method (Sutton and Barto, 2018), while the RL student is trained by an approximate SARSA method with two features. The teacher training method considers four features, and the SVM prediction is removed because of the computational limits of the web application. The results indicate that the RL teacher helps the RL students in improving their performance in test mode. Because of the learning ability limits of the RL students, the performance does not grow as much as under the RL teacher’s suggestions. The work indicates that even though two agents are trained differently, an RL agent can learn to teach another RL agent.

The author then embedded the RL teacher to the SHiB ITS and recruited fifteen participants to evaluate the teaching effectiveness of the RL teacher (Chapter 8). The results indicate that the RL teacher helps participants learn to find alternative solutions better than other groups. Except for the Textbook 2 group that gives up early in post-test 3, among the rest five groups, the RL teacher group spent the least time in post-test 3 and achieved an 85% accuracy rate. Textbook 1 and Combination 1 achieved higher accuracy rates (88% and 91%, respectively), but spent much longer time in the test. The post-survey results indicate that the RL teacher helps participants learn to use the ITS quickly and help participants to have a positive attitude on the ITS functional integration.
9.2 Future work

The author conducted experiments recruited participants older than 50 years as one group to examine the teaching effectiveness of three teaching strategies by the proposed SHiB ITS. However, certain cognitive abilities declines with the aging process, such as processing speed and memory, language, and visuospatial abilities (Harada et al., 2013). The experiment conducted by the author treats participants older than 50 years as one group which does not examine the effects on different elderly levels. Further work can focus on the teaching effects with various elderly level participants to see how the learning outcome relates to a varied aging level.

The author experimented with the teaching effectiveness with the condition that combined the *Combination 2* and the RL teacher. However, the work cannot explain the length of time participants spent in post-test 2. Future work can focus on conditions that combine different teaching strategies. Another observation that the study does not explain is that the long time participants spent in the training exercise 3 in the *RL Teacher* group. It is hard to say it is caused by confusion or by participants is playing with the system. Future work can focus on participants’ behaviors in the training exercises. Existing online learning systems will be an excellent resource for students’ behavior feature studies.

The proposed SHiB ITS contains three basic teaching strategies (*trial-and-error*, *textbook*, and *combination*), the system itself is designed for encouraging users using multiple-solutions to solve problems. The system can be used as an experimental platform that researchers can combine different conditions and test the teaching effectiveness.

In the experiments of ANN advising RL students, the author observed that the state importance the ANN chose is fluctuating around the threshold that was chosen to perform the mistake correcting teaching method (Taylor et al., 2014). As RL students perform even better with a smaller state importance threshold, future work can focus on investigating the teaching effectiveness of ANN trained with different state importance mistake correcting (Taylor et al.,
The RL students for the SHiB ITS training exercise 3 has two features. When an RL teacher is available, the student’s performance improved more than it does under test mode. That indicates there is space to explore new features that can help the RL students perform better.
REFERENCES


