UNIVERSITÁ DEGLI STUDI DI MILANO

Computer Science Department

Ph.D. in Computer Science
Cycle XXXII

Analyzing and Modeling Students' Behavioral Dynamics in Confidence-based Assessment

Ph.D. Thesis Of

Rabia Maqsood

Supervisor
Dr. Paolo Ceravolo

Director of the Ph.D. School
Prof. Paolo Boldi

Academic Year 2018–2019

Analyzing and Modeling Students' Behavioral Dynamics in Confidence-based Assessment

Rabia Maqsood
Doctor of Philosophy
Computer Science Department
Universitá Degli Studi Di Milano
2019

Abstract

Confidence-based assessment is a two-dimensional assessment paradigm which considers the confidence or expectancy level a student has about the answer, to ascertain his/her actual knowledge. Several researchers have discussed the usefulness of this model over the traditional one-dimensional assessment approach, which takes the number of correctly answered questions as a *sole* parameter to calculate the test scores of a student. Additionally, some educational psychologists and theorists have found that confidence-based assessment has a positive impact on students' academic performance, knowledge retention, and metacognitive abilities of self-regulation and engagement depicted during a learning process. However, to the best of our knowledge, these findings are not exploited by the educational data mining community, aiming to exploit students (logged) data to investigate their performance and behavioral characteristics in order to enhance their performance outcomes and/or learning experiences.

Engagement reflects a student's *active participation* in an ongoing task or process, that becomes even more important when students are interacting with a computer-based learning or assessment system. There is some evidence that students' online engagement (which is estimated through their behaviors while interacting with a learning/assessment environment) is also

positively correlated with good performance scores. However, no data mining method to date has measured students engagement behaviors during confidence-based assessment.

This Ph.D. research work aimed to identify, analyze, model and predict students' dynamic behaviors triggered by their progression in a computer-based assessment system, offering *confidence-driven* questions. The data was collected from two experimental studies conducted with undergraduate students who solved a number of problems during confidence-based assessment. In this thesis, we first addressed the challenge of identifying different parameters representing students' problem-solving behaviors that are positively correlated with confidence-based assessment. Next, we developed a novel scheme to classify students' problem-solving activities into *engaged* or *disengaged* behaviors using the three previously identified parameters namely: students' *response correctness*, *confidence level*, *feedback seeking/no-seeking behavior*. Our next challenge was to exploit the students' interactions recorded at the micro-level, i.e. event by event, by the computer-based assessment tools, to estimate their intended engagement behaviors during the assessment. We also observed that traditional non-mixture, first-order Markov chain is inadequate to capture students' evolving behaviors revealed from their interactions with a computer-based learning/assessment system. We, therefore, investigated mixture Markov models to map students trails of performed activities. However, the quality of the resultant Markov chains is critically dependent on the initialization of the algorithm, which is usually performed randomly. We proposed a new approach for initializing the Expectation-Maximization algorithm for multivariate categorical data we called K-EM. Our method achieved better prediction accuracy and convergence rate in contrast to two pre-existing algorithms when applied on two real datasets.

This doctoral research work contributes to elevate the existing states of the educational research (i.e. theoretical aspect) and the educational data

mining area (i.e. empirical aspect). The outcomes of this work pave the way to a framework for an adaptive confidence-based assessment system, contributing to one of the central components of Adaptive Learning, that is, personalized student models. The adaptive system can exploit data generated in a confidence-based assessment system, to model students' behavioral profiles and provide personalized feedback to improve students' confidence accuracy and knowledge by considering their behavioral dynamics.

Acknowledgements

I begin by quoting Albert Einstein, who famously said, "If we knew what it was we were doing, it would not be called research".

The journey of my doctoral studies followed the same vein – it started with investigation of different research areas and resulted into a successfully completed research project – which is indeed fulfillment of a dream for me. During this experimental venture, the person with whom I shared all the successful and failed attempts is my supervisor, Dr. Paolo Ceravolo – who offered his excellent guidance and commendable support that helped me to move forward after every little step I took. The trust that I got from Dr. Paolo to explore new ideas that intrigued me during the Ph.D. studies makes me eternally thankful to him.

The second very important phase of this venture continued in Cordoba, Spain, where I spent nine months and worked in Knowledge Discovery and Intelligent Systems (KDIS) Lab at the University of Cordoba, Spain. I feel highly obliged to thank Prof. Sebastián Ventura and Prof. Cristóbal Romero whose guidance and suggestions steered me to complete this research work with high notes. I am also thankful to Dr. Oscar Gabriel Reyes who spared his precious time for discussions on my methodology and results.

I would like to thank the reviewers, Prof. Christian Guetl, Prof. Naif Radi Aljohani and Dr. Alexandra Okada for providing their valuable feedback that helped me in revising the thesis.

This journey of doctoral studies could have become way too difficult, if not impossible, without the utmost support of two very dear friends of mine – my best friend, Sara Saleem and Dr. Fatima Hachem. Sara urged me to accept new challenges that came through my way particularly during the last three years and made me believe that I can succeed every time I doubted myself. Fatima, whom I befriend during the Ph.D. studies and shared plentiful joyful moments, was right there since day one till the very last to help me with academic and living-related issues which occurred frequently during

these three years.

The endless support of my beloved parents, sisters and brothers – who always gave me the strength and confidence to achieve my goals and everything I aspired to do in life – is the driving force that keeps me moving every time I fail – achieve – and dream for something new. All my successes are as much theirs as they are mine!

This section remains incomplete without thanking two more people explicitly. Amna Bilal, another best and very old friend, who has always encouraged me to do my best. I also would like to give due credit to Mrs. Isabel García Sanchez, my landlady in Cordoba. I cannot forget her smile and the welcoming gesture she gave me every time I entered the apartment feeling exhausted. Despite the language barrier between us, her hospitality made my stay in Cordoba an exciting experience which helped me to focus on my research work.

Last but not the least, I would like to thank some fellows and seniors who supported me in any way, including: Dr. Maryam Sepehri, Dr. Bahar Sepehri, Muneeb Kiani, Nicoló Campolongo, José María Moyano, Dr. Shah Nawaz, Tommaso Cesari and Amina Frija. I also acknowledge many other friends and relatives whose encouraging words boosted my enthusiasm to do more.

I pay my profound gratitude to all of you for strengthening me and contributing in this thrilled journey!

Rabia Maqsood

Table of Contents

List of Figures

List of Tables

1 INTRODUCTION

In this chapter, we layout the overall idea of this doctoral research study, which is arranged in the following manner.

- In the first section, we describe an effective yet less explored two-dimensional assessment paradigm, known as, "confidence-based assessment". This assessment model has served as a base in this research work.
- Then, we highlight our motivation for conducting research on this particular topic in the second section.
- Subsequently, in the following two sections we mention the identified research gaps and research questions of this work, respectively.
- Finally we provide an overview of our research methods and main findings, followed by the thesis structure.

Some of the material used in this chapter is taken from our published research proposal (mentioned below), which was modified later for redefining research questions and the proposed methodology for producing even better research outcomes. However, the basic idea remained the same.
"Rabia Maqsood and Paolo Ceravolo. Modeling behavioral dynamics in

confidence-based assessment. In 2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT), pages 452–454. IEEE, 2018".

1.1 Confidence-based Assessment

Confidence-based assessment is a two-dimensional assessment paradigm that takes *confidence* or *expectancy* level a student has about his/her answer, to ascertain his/her actual knowledge. In other words, while answering a question, the student is also asked to specify 'how much confidence' (s)he has in the given answer. In essence, it aims to determine 'do students *know* what they know and what they do not know'. Different ways are used in the literature to obtain this confidence measure, including: a binary value (e.g. high/low), a three-scaled discrete measure (e.g. high/medium/low), a Likert scale response (e.g. 'not sure at all' to 'very sure') or a more complex value in the form of percentage on a scale (e.g. 0% to 100%). Several researchers, for example: Adams and Ewen [2009], Gardner-Medwin and Gahan [2003], Novacek [2013], have discussed the usefulness of confidence-based assessment over traditional one-dimensional assessment approach that takes "number of correctly answered questions" as a *sole* parameter to determine the knowledge level of a student.

This two-dimensional assessment model was introduced primarily to determine students' accurate knowledge in multiple-choice questions, which are more prone to be answered correctly by guessing [Gardner-Medwin and Gahan, 2003, Novacek, 2013]. However, the inclusion of a confidence measure has found to offer more benefits than traditional assessment approach in general. Darwin Hunt studied the relation between knowledge and confidence from cognitive aspects and stressed that *retention* of some learned material is strongly related to "how much confidence" a person has in the attained knowledge [Hunt, 2003]. This claim was supported by experiments performed with students in a real classroom. The results showed that students were able to recall 91% of the correct responses a week later about

which they had high confidence, while only 25% of the least confident correct responses were retained. Adams and Ewen [2009] showed that lack of knowledge retention is observed in students through traditional assessment approach.

Student's response outcomes or knowledge[1] in combination with binary confidence levels[2] provide the following four knowledge regions, which maps to the categories defined by the model in [Hunt and Furustig, 1989]: *uninformed* (wrong answer with low confidence), *doubt* (correct answer with low confidence), *misinformed* (wrong answer with high confidence), and, *mastery* (correct answer with high confidence). These knowledge regions are shown visually in Fig. 1.1. The most critical region is 'misinformed' (top-left region), as a student's belief is high about actually incorrect knowledge. 'Uninformed' region (bottom-left region) reflects a less critical situation because the student acknowledges lack of knowledge or information about the concept presented in the given question(s). Whereas, having low confidence about the knowledge which is in fact correct shows 'doubt' state of a student (bottom-right region). 'Mastery' is the highest level of desired performance which is achieved through having high confidence in the correct knowledge (top-right region). Bruno was the first to exploit these regions to define learners' knowledge profiles [Bruno et al., 2006]. Based on his seminal work [Bruno, 1995], he derived a framework called "Confidence Based Learning" (CBL) which contains three phases: *'diagnose-prescribe-learn'*. CBL constructs learners' knowledge profiles by assessing their confidence in the subject matter and presents personalized learning contents based on the needs of the learner; and, this cycle continues until the learner achieves "mastery" (i.e. gives correct answers with high confidence).

[1] That is, *correct* or *incorrect* answer.

[2] A binary scale of confidence levels is for instance *high* or *low*.

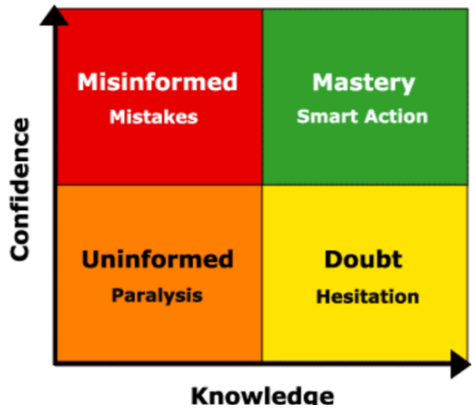


Figure 1.1: Knowledge-Confidence regions[3]

1.1.1 Self-efficacy versus confidence

According to Bandura's model, shown in Fig. 1.2, 'self-efficacy' is one's belief of executing the required behavior to achieve a certain outcome [Bandura, 1977]. And, 'confidence', as discussed in the previous section, is referred to outcome expectations that may occur in response to some behavior. He further explained that individuals can expect that a particular course of action will derive certain outcomes, but doubting in one's capability of doing something changes his/her behavior towards the task (e.g. effort, choice of activities, and persistence). Hence, both measures relate to different cognitive skills and therefore distinction should be made between them.

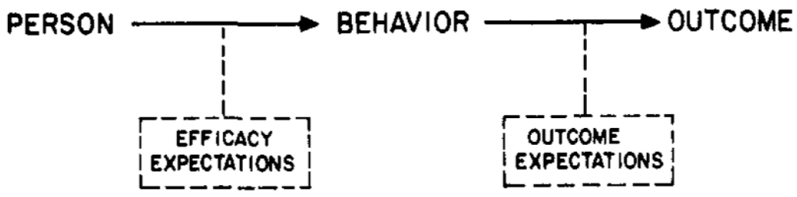


Figure 1.2: Difference between (self) efficacy expectations and outcome expectations (or confidence) – *Bandura [1977]*

[3]https://en.wikipedia.org/wiki/Confidence-based_learning

In other words, self-efficacy is a pre-task measure that influences how a person will behave to achieve some outcome. Accordingly, 'prediction' and 'postdiction' are secondary terms used by some researchers to refer to self-efficacy and confidence skills, respectively [Labuhn et al., 2010]. Some researchers have also used pre- and post-activity self-efficacy measures to examine the change in students' beliefs after performing a set of tasks. For example, in [Kanaparan, 2016], students reported their self-efficacy beliefs about an introductory programming subject at the start and end of the course.

Some researchers, however, did not differentiate between the two measures and used pre-task belief in one's performance, i.e. self-efficacy, to measure his/her confidence about the answer, see for example [Lang et al., 2015, Timmers et al., 2013, Van der Kleij et al., 2012]. On the other hand, empirical studies have shown that these two measures of expectation offer different kinds of information about a learner's attitude and both shall be treated differently [Stankov et al., 2014].

Due to its post-task nature, confidence measure gives a more realistic view of a student's knowledge expectation in relation to a recently completed task [Stone, 2000]. Students are thus expected to increase their confidence accuracy skill over time and make better judgments about their performance. This argument is validated by an empirical study [Nietfeld et al., 2006] conducted on undergraduate students who showed an increase of one standard deviation in their calibration (or confidence) accuracy whereas no significant improvement was found in their self-efficacy.

1.1.2 Measuring and enhancing students' confidence judgment skills

A way of quantifying students' knowledge level in confidence-based assessment is by using a marking scheme that considers both parameters (i.e. confidence level and response outcome). In this respect, various marking

schemes are available in the literature to estimate students' knowledge level, e.g. see [Bruno et al., 2006, Francese et al., 2007, Gardner-Medwin, 2005, Petr, 2000]. These marking schemes share the common idea to highly penalize the wrong answers given with high confidence, and reward less low confident-correct answers; whereas, correct answers given with high confidence receive the maximum credit. The objective is to differentiate between *guesswork* and *actual knowledge*, and in parallel to discourage *under-* and *over- confident* students.

Confidence measure has also gained importance due to its predictive validity for student's academic achievement [Lang et al., 2015, Stankov et al., 2014]. But, very little is known about the change in a student's confidence level and thus, it has been treated as a self-report measure. Initially, change in one's confidence level was perceived to be related with personality traits of an individual, however, recent work in psychology has shown that accuracy in estimating one's performance is rather related to a person's ability [Burns et al., 2016]. The rationale behind this is that a student specifies his/her confidence level about a *recently* completed task and this involves metacognitive judgment of accuracy possessed by each individual.

Additionally, presence of a 'general' confidence factor has been associated with "the habitual way in which people assess the accuracy of their cognitive performance" [Stankov et al., 2015, p.186]. In accordance to this view, it is reasonable to assume that students confidence accuracy would incline towards either of the two extremes, i.e. over- or under- confident; and falling somewhere in the middle between them reflects moderate or good accuracy rate, which is desirable.

Confidence accuracy is also referred as "calibration" in the literature which can be measured as either an *absolute value* or as a *direction* of confidence judgment *direction* [Nietfeld et al., 2006, Rutherford, 2017]. Calibration score (the absolute value), is computed as a difference in student's confidence rating and his/her actual performance. The latter approach is

more effective in differentiating between two or more students who may attain the same calibration score based on simple matching between expected and actual performance. Hence, determining the direction of calibration or bias has received more attention from researchers. There are various measures available which differ in terms of accuracy and respective parameters used to compute bias (see [Rutherford, 2017] for a detailed comparison of different measures).

Several empirical studies have identified that students are poor estimators of their abilities [Labuhn et al., 2010, Lang et al., 2015, Mory, 1994, Petr, 2000, Timmers et al., 2013], that is, they do not specify their confidence level accurately. This is a critical issue associated with confidence-based assessment and in fact is the main reason to limit the adoption of this assessment model in large [Lang et al., 2015]. Additionally, degradation in students academic performance over time is found to be linked with over- and under-confident students. According to [Boekaerts and Rozendaal, 2010], under-confident students may lose motivation for learning due to lack of self-confidence, whereas, overconfidence restrain students from learning something new. He further argued that if attention is not paid to one's poor judgment skill of his/her abilities at task-level, it may become a personality trait.

A natural question that arises here from this discussion is: *"Can this over- and under-confidence judgment behaviors of the students be changed?"*. Because it worth to invest further effort and time only if we can shape students' behaviors from either extreme condition to a (reasonably) better point.

Gardner-Medwin has stressed that students should develop the skills of correct confidence judgment through (self) practice[4] and adopt careful habits of thinking [Gardner-Medwin, 2005] while specifying their confidence rating. Bruno's CBL framework [Bruno et al., 2006] relies on the

[4]Using confidence marking scheme that assigns positive and negative scores based on the confidence level and actual response outcome; as the one developed by himself.

same principle that students should improve their confidence accuracy by self-evaluating their skills in a system which uses confidence-based marking (CBM) scheme to grade their responses. Since, CBM scheme penalizes high confidence-wrong responses, it discourages unthoughtful actions of students who always specify 'high' confidence level for all questions. On the other hand, students who remained 'less' certain about their knowledge receives minimum credit for correct responses. Thus, students are enforced to put conscious efforts to gain performance level by improving their knowledge and confidence estimation skill.

An alternative approach to improve students' rethinking and reflection without using negative marking is proposed in [Hench, 2014], that employs graphical means to provide feedback regarding question difficulty and under or over confidence degree associated with a question. Their technique, named as 'Confidence/Performance Indicators', captures information from prior data of the students about confidence and performance level for each question, which is then used to provide feedback to new students. However, the impact of Confidence/Performance Indicators on students' performance is not evaluated, which could have strengthened its applicability from a theoretical perspective.

Furthermore, some researchers have proposed to steer students' confidence judgment skill from either extreme values (overconfident/underconfident) through useful instructions [Boekaerts and Rozendaal, 2010, Labuhn et al., 2010, Stone, 2000]. For example, underconfident students need to (re)gain trust in their capabilities to do well in the respective subject/domain. Similarly, overconfident students can benefit from the instructions that instill self-realization about improvement in their knowledge and abilities. Nietfeld et al. (2006) in their experimental study with undergraduates, offered feedback on calibration which resulted into an improvement in calibration accuracy and performance of the 'treatment' versus 'control group' students.

These findings are very promising and disclose that there is a good possibility of enhancing learners' confidence estimation skill by adopting a mechanism that highlights main objective of confidence-based assessment and reminds students to specify their confidence more consciously.

1.1.3 Confidence as an attribute of "self-regulation"

Self-regulation is a metacognitive skill which plays central role in fostering students' learning through a three-phase cyclic process containing: *forethought*, *performance* and *self-reflection* activities [Zimmerman, 2000], in the same order. The 'Forethought' phase happens before getting involved into the given task (or a learning process) and includes activities like: task analysis and self-motivation beliefs. The 'Performance' phase relates to the action(s) involved during performing the actual task, for example: self-control, self-observation and task strategies. The 'Self-reflection' occurs after engaging in a task and is related to student's performance. This is determined through self-judgment and self-reaction activities, and may influence the forethought process of a subsequent self-regulatory cycle.

Self-monitoring is identified as another key characteristic that is associated with the self-regulation process and is derived through 'confidence judgment(s)' about expected performance in a task [Nietfeld et al., 2006]. Authors have shown that improving this metacognitive monitoring ability positively impacts (a student's) calibration and performance outcomes.

Similarly, multiple metacognitive theories mentioned in [Boekaerts and Rozendaal, 2010] referred to confidence as 'metacognitive judgment of solution's correctness' which is mandatory to achieve a higher level of self-regulation. Different arguments are presented to express that miscalibration or poor confidence judgments (e.g. over- and under-confidence) threaten students' self-regulation [Boekaerts and Rozendaal, 2010], as students regulate their learning process in accordance to their expectations [Labuhn et al., 2010]. Additionally, self-regulation profiles of novice and expert learners

vary to the extent that it affects their approach and motivation towards learning [Zimmerman, 2002]. However, each self-regulatory process can be learned through guided instructions or feedback and practice.

Further, it is stressed that students' confidence accuracy not only impacts their motivation to 'engage' in a task but also influences the types of strategies they select for doing a task [Boekaerts and Rozendaal, 2010]. This correlation between confidence accuracy and higher level of engagement during assessment is also highlighted by Lang et al. [2015]. Therefore, identifying and increasing student engagement would be a useful step towards fostering an increase in the performance outcomes and realistic assessment of their abilities (or confidence judgment).

1.1.4 Summary

The review of confidence-based assessment has revealed several benefits of this two-dimensional assessment approach on students' academic performance [Nietfeld et al., 2006], knowledge retention [Adams and Ewen, 2009, Hunt, 2003], and metacognitive abilities of self-regulation [Nietfeld et al., 2006] and engagement [Boekaerts and Rozendaal, 2010, Lang et al., 2015] depicted in a learning process.

Confidence when taken as a post-task measure that reflects the ability of an individual in estimating his/her performance. High accuracy in one's ability of confidence judgment is crucial as it relates to good performance and knowledge retention. However, students tend to be inclined towards either extreme values of confidence measure, that is, overconfidence and underconfidence. These extreme measures are problematic as they can negatively impact students' motivation and performance. For example, overconfident students would lose motivation to learn as a result of successive failure attempts despite all their efforts [Boekaerts and Rozendaal, 2010]. Similarly, being underconfident about one's abilities reveals a lack of self-confidence in the student, which in result may decrease his/her interest in the learning

process.

Confidence judgment about solution's correctness is a metacognitive skill that plays an important role in the self-regulation process. The results of some empirical studies have found positive correlation between confidence accuracy and performance (e.g. Stankov et al. [2014]), and self-regulation measures (e.g. Boekaerts and Rozendaal [2010], Labuhn et al. [2010], Nietfeld et al. [2006]) when investigated in real classrooms. Hence, good confidence judgment enhances self-regulation metacognitive skill in students which in return promotes their motivation and engagement in a learning process. One approach could be to use some marking scheme that penalizes the wrong answer(s) and rewards correct answer(s) given with different confidence measure, as explained in Section 1.1.2. However, this approach has some drawbacks including, an increased pressure on students to avoid negative marking, motivation degradation and lack of interest in the ongoing learning or assessment process.

1.2 Motivation for this Research Work

Despite having many benefits over the traditional one-dimensional approach, confidence-based assessment is still less explored by educational researchers. The validity of a strong connection between confidence and increased knowledge offers a paramount opportunity for educators to enhance students' performance by creating more confident and productive students [Adams and Ewen, 2009]. In this context, the utility of computer-based learning/assessment systems should be availed to target a large number of students, whose interactions with the system can be recorded to analyze and gain better insights about 'unproductive' behaviors during confidence-based assessment [Maqsood and Ceravolo, 2018]. Based on (behavioral) data analysis students can be guided with personalized feedback to accelerate their self-reflection and confidence accuracy skills, for example as done in [Hench, 2014]. But, exploiting students' logged data to investigate their behav-

iors during confidence-based assessment is missing in the current literature; which could have led us to construct behavioral profiles of the students and generate predictions about the evolution of their behaviors. This Ph.D. thesis is a step forward in this direction.

1.3 Research Gaps

The state of the art shows that educational psychologists and theorists have contributed vastly in investigating the role and impact of confidence judgment skill on students' performance outcomes, knowledge retention, and self-regulation. However, to the best of our knowledge, these findings are not exploited by the educational data mining community, aiming to exploit students (logged) data to investigate their performance and behavioral characteristics to enhance their performance outcomes and/or learning experiences [Romero et al., 2010]. Existing computer-based assessment systems, which are rather very few, only present confidence-accuracy scores to students. But, student-system logged interactions are not examined to analyze their behaviors during the assessment, which could be useful to identify students having different strengths and weaknesses. This leads us to the following open issue.

- Which behaviors depicted by students during confidence-based assessment can be useful for differentiating them?

Firstly, the literature review showed that one crucial factor which has limited the adoption of confidence-based assessment in large is students' poor judgment of their knowledge (correctness) [Lang et al., 2015]. Hence, the ultimate goal of an adaptive confidence-based assessment system should be to assist students in becoming more certain about 'what they know' and 'what they do not know'. For this reason, it is vital for an adaptive system to monitor and identify students lacking this ability earlier.

Secondly, factor(s) that affect a student's confidence accuracy are not known exactly but it is correlated with engagement [Boekaerts and Rozendaal, 2010, Lang et al., 2015] which determines one's level of involvement in a learning process. Thus, students' higher level of engagement (or involvement) during the assessment is very important for them to make better confidence judgment of their knowledge instead of specifying high or low confidence level in all the questions.

In summary, students' behaviors can be categorized as productive or unproductive based on these two factors, that is, engagement and confidence accuracy. But, investigation of students behaviors during confidence-based assessment is ignored by researchers to date and this study aims to fill this gap. At the same time, it is of critical importance that we develop a suitable mechanism to model and predict students' varied behaviors using their interactions with a computer-based assessment system – where predicting students' future behavior is rarely addressed in the existing works, see Section 2.1.3.

1.4 Research Questions

This research study aimed to identify, analyze, model (or represent) and predict students' dynamic behaviors triggered from their progression in a computer-based assessment system, offering *confidence-driven* questions. In particular, we defined the following two research questions (RQs)[Maqsood and Ceravolo, 2018].

- RQ-1: What behaviors can be used to determine student engagement/disengagement in confidence-based assessment?
- RQ-2: How can we model and predict these behaviors to construct students' behavioral profiles?

First research question deals with identifying suitable parameters reflecting students' problem-solving behaviors during confidence-based assessment;

this defines the 'theoretical aspect' of this work. And, the second question is related to the data processing technique(s) to analyze and represent the pre-discovered behaviors; this defines the 'empirical aspect' of this work as well as the implementation work we performed. Hence, this research work contributes to add value to the fields of educational research (i.e. theoretical aspect) and (educational) data mining (i.e. empirical aspect).

The outcomes of this work will layout a framework for an adaptive system, contributing to one of the central components of Adaptive Learning, that is, personalized student models – used to offer personalized feedback to the students; which itself is a complete research topic and is left as a future work (will be discussed in the last chapter).

1.5 Research Methods and Findings

To answer the two research questions (RQs) given in the previous section, we defined three different research objectives which are mentioned in Chapter 3, Section 3.1. For each of the research objective, we conducted a research study. Sections 3.3, 3.4 and 3.5 describe them using two real datasets (see Section 3.2 for details of the experimental studies and the collected data).

To be precise, our first objective (Objective I) was to analyze the correlation between students' different problem-solving parameters in relation to confidence-based assessment. We used different statistical methods in the first research study to analyze correlations between different parameters. The results which are described in detail in Section 3.3.3.2, show the usefulness of three interesting problem-solving parameters namely: student's *response correctness*, *confidence level* and *feedback seeking behavior*. These three parameters were then used in the second research study to achieve our second objective (Objective II), which aimed at defining a classification scheme to categorize students' problem-solving behaviors as engaged or disengaged. The proposed engagement/disengagement scheme (given in Section 3.4.1) was qualitatively evaluated on a real dataset using the K-means

clustering algorithm, details of the results can be found in Section 3.4.2.3. Finally, our third objective (Objective III) was to develop a mechanism to model and predict students' varying engagement/disengagement behaviors using probabilistic models. In the third research study, we proposed a new method called "K-EM" for mixture Markov models which estimates the model parameters for multivariate categorical data (see Section 3.5.2.2). To evaluate the performance of K-EM, we carried out different experiments which along with the obtained results are presented in Sections 3.5.2.5 and 3.5.2.6, respectively. Our findings show that the resultant Markov chains for the two datasets achieved better accuracy in predicting students future behavior. To conclude, we developed a new method to model and predict students' dynamic behaviors and the resultant Markov models can be used to construct students' personalized behavioral profiles, which are used to visualize and interpret students' intended behaviors (see Section 3.5.2.6, paragraph A).

1.6 Thesis Structure

The remaining of this thesis is organized in the following chapters.

- **Chapter 2 (Background):** It provides background information about the related concepts used in this work to answer our research questions, as mentioned in the previous section. For RQ-1, we reviewed various theoretical and non-theoretical approaches used to define and estimate student engagement. For RQ-2, we studied different data mining and machine learning methods and chose Markov chains and mixture Markov model to represent students' varying behaviors. In this chapter, these two probabilistic models are explained using basic notions and concepts.

- **Chapter 3 (Objectives, Methodology, and Results):** In this chapter, we first state three research objectives which were formulated

by keeping in view our research questions and the background knowledge of the related concepts, as mentioned in Chapter 2. Next, we provide details of the experimental studies and computer-based assessment systems used for data collection. Subsequently, we discuss our methodology and the obtained results for all research objectives; which were also presented in the following published research works.

1. "Rabia Maqsood and Paolo Ceravolo. Modeling behavioral dynamics in confidence-based assessment. In 2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT), pages 452—454. IEEE, 2018".
2. "Rabia Maqsood and Paolo Ceravolo. Corrective feedback and its implications on students' confidence-based assessment. In International Conference on Technology Enhanced Assessment. Springer, 2018".
3. "Rabia Maqsood, Paolo Ceravolo, and Sebastián Ventura. Discovering students' engagement behaviors in confidence-based assessment. In 2019 IEEE Global Engineering Education Conference (EDUCON), pages 841—846. IEEE, 2019".
4. "Rabia Maqsood, Paolo Ceravolo, Cristobal Romero, and Sebastián Ventura. Modeling and predicting students' engagement behaviors: A new approach for mixture Markov models. (Under review)".

- **Chapter 4 (Conclusive Remarks):** This final chapter contains our conclusions and a summary of the contributions made. Then, we present a detailed discussion on our methodology and obtained results, including their limitations and recommendations for potential improvements. Finally, we discuss the future work directions of this research work.

- **Appendices:** The appendices are divided into two sections which present the following contents. Appendix A contains some details and screenshots of the two computer-based assessment tools used in this thesis for experimentation and data collection. While, Appendix B contains detailed results of students' future (or next activity) prediction accuracy computer for the resultant clusters obtained using three model-based clustering methods namely, EM, emEM and our proposed K-EM method.

2

Background

In this chapter, we present a background overview of the key topics exploited in this doctoral research work. In particular, the background is partitioned into the following three sections.

- First section reviews the notion of student engagement as it is defined and measured in the current literature. We then discuss different approaches adopted to estimate students' engagement/disengagement using their logged data.

- In the second section, we introduce, through basic notations, definitions and related sub-topics, "Markov chains", a commonly used probabilistic model to analyze sequential data.

- Section three provides details about the mixture Markov models yielded through model-based clustering and related issues, that include: different methods for determining the number of mixtures and description of a well-known algorithm to estimate mixture parameters given the input data.

2.1 Engagement

Although there are a lot of definitions available for engagement in the literature, a universally accepted one is still missing. In general, engagement is referred as *active participation* in an ongoing *task* or *process*. It has become a crucial notion for technology-enhanced learning due to its correlation with students' academic performance [Cocea and Weibelzahl, 2007, Joseph, 2005, Kanaparan, 2016, Pardos et al., 2014]. In fact, engagement is a distinctive characteristic that strongly indicates a person's motivation to perform an activity [Cocea and Weibelzahl, 2011]. The concept of 'school engagement' started getting attraction in late 90's through realization of the existence of some factors that might have played a role in students' poor academic performance and high rate of dropouts [Fredricks et al., 2004]. Likewise, earlier works are primarily based on theoretical reasoning with a focus on developing theoretical models and frameworks that may be useful to build a connection between students' actions and their thought (or cognitive) process. And, the identified relation(s) can be helpful to understand reasoning behind different actions performed by a student. In the followings, we discuss some different perspectives adopted by researchers to understand the term 'engagement' and particularly 'student engagement' with a focus on quantitative approaches.

2.1.1 Theoretical and non-theoretical engagement models

Fredricks et al. [2004] described engagement as a *multifaceted* construct that comprises the following three dimensions: cognitive, emotional, and, behavioral. *Cognitive* engagement refers to the investment of effort and thoughtfulness to comprehend complex learning ideas and concepts. *Emotional* engagement focuses on the student's positive and negative reactions to the environment. And, *behavioral* engagement draws on the idea of students' participation in learning activities.

Bouvier et al. [2014] proposed a quantitative approach to analyze and monitor engagement behaviors using a trace-based method that exploits users' logged interactions with interactive systems. The idea is to transform low-level raw traces into useful high-level abstractions of different engagement behaviors. Their approach constitutes of three theoretical frameworks that include: *Self-Determination Theory*, *Activity Theory* and *Trace Theory*. The proposed approach works in three steps: starting with the identification of abstract engaged behaviors in relation to users different needs (*Self-Determination Theory*), these relationships are given meaningful identity through mapping onto different activities that a user may perform (*Activity Theory*), and, are reified at the operation level by translating into corresponding do-able actions (*Trace Theory*) to be performed in an interactive environment.

To assess the benefits and usefulness of computer systems used in the classroom for educational purposes, one attempt Their direct observations of the participants resulted into a taxonomy of student engagement with seven levels, arranged in order of complexity: *disengagement*, *unsystematic engagement*, *frustrated engagement*, *structure dependent engagement*, *self-regulated interest*, *critical engagement*, and *literate thinking*. Higher levels of the taxonomy reflect students' competence to navigate and operate the computer system in a more strategic way that are aligned with their learning goals. We believe that such hierarchical or structured arrangement of engagement levels could be useful to characterize the change in a student's behavior from one level to another and identify possible contributor(s) to that change, to help students in reaching to a higher engagement level.

Similarly, to identify different levels of engagement, a scheme is proposed by Tan et al. [2014] in which students observable behavioral factors (i.e. students' actions with e-learning environments, e.g. intelligent tutoring system) are mapped onto five levels of engagement, see Table 2.1. According to them, student engagement is a reflection of active involvement in a

Table 2.1: Mapping of engagement levels to engagement indicators – *Tan et al. [2014]*

Level	**Behavioral Indicators**
Level 5: Enthusiasm in learning	Work on additional tasks. Respond to others' questions in an online forum. Multiple solutions on tasks.
Level 4: Persistency	Revisiting and spent more time on more difficult tasks. Appropriate use of hints. Completion of all tasks. Completion on time.
Level 3: Participation	Work on moderately challenging tasks. Completion of a minimum number of tasks.
Level 2: Passive participation	Guessing on the majority of tasks. An incompletion on all or the majority of tasks. Frequent but inappropriate use of hints.
Level 1: Withdrawal	No response to assignments.

learning process, and this 'involvement' can be identified through student-system logged interactions. As expected, reaching to a higher engagement level requires more effort.

The three dimensions of engagement defined by Fredricks et al. [2004] (i.e. cognitive, emotional and behavioral) are well-accepted and studied widely in the literature; some researchers have utilized all three dimensions in their works while others have focused on a specific dimension for determining student engagement. Kanaparan [2016] in her thesis has studied all three dimensions of engagement as a determinant of higher student performance in an introductory programming course. Indicators used in her

work for behavioral engagement include: help-seeking, persistence, and effort invested in solving problems and overall learning process; indicators for cognitive engagement include: deep learning, surface learning, and trial-and-error learning strategies; and indicators for emotional engagement include: interest, enjoyment, and gratification a student feels towards the learning environment. Survey questionnaires were used to collect students responses to these different dimensions of engagement.

Some researchers, however, preferred to construct their non-theoretical schemes for determining student engagement through data analyzes. For example, Beal et al. [2006] adopted the notion of students' active participation in a current task for defining engagement. And, developed their scheme for its estimation through the classification of student-system interactions, in fact as a collection of students' actions [Beal et al., 2007], recorded by an ITS. Another self-defined classification scheme for categorizing students logged activities into engaged and disengaged behaviors is proposed by Brown and Howard [2014]. They relied on data analyzes to define two engagement classes (referred to as, on- and off- tasks). Joseph [2005] on the other hand, used a more sophisticated method known as, Item Response Theory (IRT),

Hershkovitz and Nachmias [2009] referred to engagement as an 'intensity' measurement of motivation, meaning that a student's approximate level of engagement can be determined through activities performed during a learning process. Cocea and Weibelzahl [2006] considered engagement as a fundamental component of motivation which impacts students' quality of learning, especially in e-learning environments. Their quantitative approach identified two levels of motivation namely, 'engagement' and 'disengagement'; using a set of human-expert pre-defined rules that classify students' logged interactions with a learning environment. A third level called 'neutral' was introduced in a latter work [Cocea and Weibelzahl, 2007] to classify cases which do not relate to either engaged or disengaged levels.

Pardos et al. [2014] studied students' behavioral engagement along with affective states using students logged interactions with a mathematics tutoring system (called ASSISTments). The automated behavioral detector model aims to identify two specific behavioral events depicting students' active (or in-active) participation during assessment, namely: 'off-task' and 'gaming' behaviors.

Besides engagement behavior detection, there are several other works on analyzing and manipulating students' logged data to gain better understating about their usage of the learning environments. However, we have specifically reviewed the ones which targeted student engagement and/or behavior detection.

2.1.1.1 Conclusion

We conclude that a student's engagement reflects his/her active involvement in a learning process that becomes even more important when students are interacting with a computer-based learning or assessment system. The prime objective of these systems is to facilitate students to learn and improve their learning outcomes, however, if a student does not show interest or engage appropriately during the learning process (s)he may seek failure or degradation in performance [Cocea and Weibelzahl, 2007] and consequently abandon the learning process. There are evidences to show that students' online engagement (which is estimated through their behaviors while interacting with a learning/assessment environment) is also positively correlated with good performance scores in standardized exams [Pardos et al., 2014] and students' academic outcomes [Kanaparan, 2016, Vogt, 2016]. Recently, researchers have been showing great interest in measuring student online engagement after realizing that the student's knowledge gap cannot be addressed easily if he/she does not show interest while interacting with a learning environment [Desmarais and Baker, 2012].

Ideally, student engagement should be estimated through a combination

of these three factors. Since the three engagement measures differ in nature, it is usually impractical to collect all the data automatically in real-time, providing the evidence of their presence or absence. In this work, we focused on *behavioral engagement* of the students which are found to be correlated with students academic achievements [Fredricks et al., 2004]. Our primary reason for choosing this particular engagement dimension is its observable nature [Kanaparan, 2016, Tan et al., 2014]; that is, activities or actions performed by a student Another reason for focusing on behavioral engagement is its potential of changing students unwanted response towards the usage of e-learning environments, which is crucial for acquiring real benefits of educational software [Bangert-Drowns and Pyke, 2001].

2.1.2 Different approaches for data collection

Student engagement has been studied from different perspectives as discussed in Section 2.1.1. Similarly, several different approaches exist for data collection to measure or estimate student engagement. Some of them mentioned by Chapman [2003] includes: self-report (through a questionnaire); checklists and rating scales - done by teachers; direct observations of students in a class; (student) work sample analyzes (e.g. project, portfolio, etc.); and, case studies. Kanaparan [2016] used survey questionnaires to collect students responses about their cognitive, behavioral and emotional engagement in an introductory programming course. Their answers were analyzed and coded by researchers to identify possible indicators to be used for each respective engagement dimension. One critical problem that weakens the validation of survey/questionnaire results is the wording used in questions' statements that many students may find ambiguous while answering. However, it is still a widely used approach for data collection in several domains.

One of the earliest data collection approaches also includes direct (or field) observations of the participants by human recorders who take notes

during the experiment. Bangert-Drowns and Pyke [2001] used this approach in their work, where human recorders monitored and took notes of the students while they were interacting with educational software in the classroom. Their field notes include student-system interactions, body posture, off-task behavior, and verbalization; which were later used to make inferences about students behaviors and to estimate their approximate engagement in the learning process.

With the advancement of technological devices used for educational purposes, the popularity of video cameras for data collection is also increasing. In a recent work of Hamid et al. [2018], students' facial image data is used to predict their engagement behavior using machine learning technique. Students facial expressions were captured through a camera during a problem-solving session, and classified as engaged or disengaged through face detection and eye positioning features. This is a good application of powerful image classification algorithms in educational domain. However, the work is still limited in terms of the number of features used for engagement behavior classification. Further, students can easily *game* the 'system' (which is engagement behavior classifier, in this case); by getting involved in some off-task activities on their computers. Tracking and integrating student activities (in the learning environment) with their facial expressions in the existing model will strengthen the usefulness of such approaches.

Computer-based learning or assessment systems have the capability to record all the actions performed by the students in an uninterrupted manner, which makes it one of the most popular approach for data collection in the educational domain to perform quantitative data analyzes. For example, Beal et al. [2007, 2006], Brown and Howard [2014], Cocea and Weibelzahl [2006, 2007], Hershkovitz and Nachmias [2009], Joseph [2005], Pardos et al. [2014], Tan et al. [2014]; all exploited students logged data to demonstrate and assess their approach for estimating student engagement. Bouvier et al. [2014] tested their engagement framework using players' logged data in a

social game environment. Furthermore, users logged traces are also utilized to understand their engaged behaviors within team discussions using a collaborative environment in [Seeber et al., 2014]. Log files usually contain the following basic information: activity type, timestamp, and unique user id; which are mostly sufficient to create each person's activity profile; activities can be temporally ordered, if required. However, other attributes can also be recorded from students' interactions based on the requirements of the problem to be investigated (for example session-id; student's scores, lesson number, and other performance attributes; mouse movements; etc.).

A different approach could be a combination of two or more data collection techniques. For example, Beal et al. [2006] proposed to integrate multiple data sources to better estimate student engagement. In particular, they collected data from the following three sources: a) students' self-report data about their motivation (in mathematics), b) teachers report on students' motivation and achievement, and, c) activity classification of students' interactions with an intelligent tutoring system. The two datasets used by Pardos et al. [2014] for engagement detection were also collected from multiple sources. More precisely, they used field observations and students logged interactions captured by an ITS. Human experts' field observations were synchronized with student logged data to define a mapping between recorded interactions and various affective and behavioral states.

2.1.2.1 Conclusion

To increase the viability of our approach, we decided to exploit students logged data which is captured in real-time uninterruptedly as students interact with a computer-based system. Moreover, the generality of the data (attributes) recorded by these systems makes it possible to reuse the developed model or approach with data collected from other learning environments, as demonstrated by the works of [Cocea and Weibelzahl, 2011] and [Tan et al., 2014]. Exploratory work of Tan et al. [2014] showed that comparing

behavioral engagement of two groups of students who have worked on different ITSs did not reveal any significant difference. Cocea and Weibelzahl [2011] on the other hand showed the validity of their previously developed engagement detection model using data from a less structured learning management system.

Although integrating multiple sources of data collection can provide additional information about students behaviors with interactive learning systems. However, it requires more time and effort to gather the data and devise a mechanism to synchronize data collected in multiple forms (e.g. survey results, students real-time interactions, facial expressions, etc.). Another restriction of such an approach is that the whole process of data collection and analysis cannot be automatized, which is essential if the developed or proposed method is to be implemented in an adaptive learning system.

2.1.3 Determining student engagement through logged data

In relevance to this research work, here we review the parameters (or data attributes) and techniques used in research studies which utilized students' logged data as a source for determining student engagement as highlighted earlier in Section 2.1.2.

Cocea and Weibelzahl [2006, 2007] defined engagement as an attribute of motivation called 'interest' - that a student has in a particular domain or subject, and so it determines the 'effort' (or amount of time) a student spends in an activity. Variables used in this study include frequency and effort spent on both reading pages and quizzes activities performed by students while interacting with a learning environment. Students logged sessions were labeled as 'engaged', 'disengaged' or 'neutral' by human experts based on a set of rules defined earlier from manual analysis of the data. Eight data mining techniques were then used to construct a more accurate prediction model for (dis)engagement, for example, Bayesian nets, Logistic regression, Decision tree, etc.

In [Hershkovitz and Nachmias, 2009], students' logged data was collected from an online vocabulary, which was analyzed visually by human experts to identify the important variables relating to their theoretical framework of motivation. Then, different variables were grouped by similarity using Hierarchical clustering algorithm. Cluster group containing 'time on task' (percentage) and 'average session duration' variables were mapped to students engagement behaviors.

Engagement tracing approach proposed by Joseph [2005] is based on Item Response Theory, which computes the probability of a response's correctness given the amount of time spent on it. In their model, engaged students are assumed to give the correct answer(s) with a certain probability. Whereas, disengaged students have an associated probability of guessing the answer correctly.

Beal et al. [2006] developed their notion of student engagement which assumes active participation of the students in a current task. To estimate student engagement, three problem-solving related variables were used, namely: 'response correctness', 'time spent per problem' and 'help usage'. Students' problem-solving activities were classified into five different engagement levels including: *Independent-a, Independent-b, Guessing, Help abuse* and *Learning*. A brief description of these engagement levels is given below.

- Independent-a: student provides a correct response
- Independent-b: student provides an incorrect response, followed by a correct response to the same problem
- Guessing: immediate selection of one or more answers (i.e. within first 10 seconds)
- Help abuse: multiple help requests for the same problem without reading the previous one (within 10 seconds interval)
- Learning: help requested and read (i.e. displayed for at least 10 seconds), before providing an answer or another help request

As we can see, they defined a time limit of 10 seconds from pre-analysis of the data and used it as a boundary condition for differentiating between different behaviors, for example: help abuse and learning.

Another work using a self-defined notion of student engagement is done by Brown and Howard [2014], they adopted on- and off- tasks terminology to refer to the engaged and disengaged behaviors, respectively. Keyboard and mouse events were recorded as students interacted with a computer-based assessment system. The collected data includes three event processes: 'total time', 'response accuracy' and 'proper function execution' [1]. Events were labeled as disengaged if 'off-task' behavior is identified and engaged otherwise. And, the entire student trace was classified as off-task if it contains at least 25% off-task events. Engaged behavior was further divided into three levels based on: 'response correctness' and 'time spent', as described below; disclosing students with different skills and needs.

- Student on-task and has a series of fast responses with a series of correct answers (OCF) – may needs questions of higher difficulty.
- Student on-task and has a series of slow responses with a series of correct answers (OCS) – may understand the material and require more time to think.
- Student on-task and has a series of slow responses with a series of incorrect answers (OIS) - may lack understanding and need questions of lesser difficulty.

The two engagement behaviors investigated in [Pardos et al., 2014] include: 'off-task' (or on-task) behavior and 'gaming' (or not-gaming) behavior. Feature selection methods were applied to identify the most suitable students'

[1] Functions defined as a combination of keyboard stroke and/or mouse position, for example: 'begin test', 'next page', 'previous page' functions are defined based on mouse left click. A student's behavior is classified as 'on-task' if the mouse is clicked in identified locations and 'off-task' otherwise.

activity parameters for different behavioral states. And, using these features as input, an automated detector model was constructed for each behavior separately through machine learning classification techniques. The list of features used for detecting both behaviors is as follows.

- Off-task behavior detection features: *total number of attempts, time taken, total number of incorrect actions, average number of scaffold requests* and *correct actions taken by the student.*
- Gaming behavior detection features: *use of bottom-out hints, total number of hints used, average hints count, total number of incorrect actions* and *scaffolding requests, if any requested by the student.*

To provide a real implementation of their student engagement blueprint (given in Table 2.1), Tan et al. [2014] defined a behavioral classification scheme containing the following 11 categories: 'off-task', 'gaming', 'guessing', 'on-task', 'on-task using hints', 'completion minimum work', 'completion on time', 'revisit moderate-difficult tasks', 'revisit hard tasks', 'extra-task' and 'extra-time'. The first five behavioral indicators were defined at problem-level, whereas the remaining six at session-level containing n temporal order problems. In this work, students' recorded observations were arranged in temporal order before computing student engagement, which was not considered in most of the reviewed research studies.

Engagement detection has gained popularity in other domains as well, e.g. *gaming*; due to its potential of providing usable information about targeted users. Bouvier et al. [2014] has tested their theoretical engagement framework (described in Section 2.1.1) on a real game-based environment, to determine if a player is engaged or disengaged. The players' actions during game-playing were logged by the application to record their behaviors, as a series of action traces. These traces were examined and annotated as 'engaged' or 'disengaged' behavior by human experts, generating a set of transformation (or classification) rules.

Table 2.2 provides the summary of student activity attributes and approaches used by several researchers to estimate student[2] engagement using their logged interactions with computer-based learning or assessment environments.

Table 2.2: Summary of different parameters and approaches used for determining engagement using logged data

Research work	**Engagement measurement variables**	**Approach used**
Joseph [2005]	Question response time; answer correctness	Item Response Theory based model, given the response's correctness and time spent, is used to find the probability of a student being engaged or disengaged
Beal et al. [2006]	Response correctness; time spent per problem; help usage	Problem solving activities classified into different engagement levels based on correctness, time and help usage behaviors
Cocea and Weibelzahl [2007]	No. of pages read; time spent reading pages; no. of quizzes; time spent on quizzes	Sessions labeled as engaged/disengaged by human experts using some pre-defined rules; eight data mining techniques used this labeled data to construct better prediction model of (dis)engagement

[2]The work of Bouvier et al. [2014] is based on determining player engagement or disengagement in a game using their recorded actions.

Table 2.2 – *Continued from previous page*

Research reference	**Engagement measurement variables**	**Approach used**
Hershkovitz and Nachmias [2009]	Time on task percentage; average session duration	Used hierarchical clustering to find closest similar variables (that match to theoretical model of motivation) amongst seven pre-identified variables (through students' data analysis)
Bouvier et al. [2014]	Time stamp of a player's action in the game; action type	Human experts annotated users logged traces as engaged or disengaged
Brown and Howard [2014]	Time on task; response correctness; function (events defined based on keyboard stroke and/or mouse click position)	Traces labeled as on-/off- task (through data analysis) based on proper function execution; on-task (or engaged) traces use response correctness and time on task features to further distinguish students
Pardos et al. [2014]	Total no. of attempts; time taken; no. of correct & incorrect actions; average scaffolding requests; total hints used; average hints count	Machine learning feature selection and classification methods were used on experts' annotated student actions for constructing automated engagement detectors

Table 2.2 – *Continued from previous page*

Research reference	**Engagement measurement variables**	**Approach used**
Tan et al. [2014]	No. of hints used; problem-reading time; problem difficulty level; response correctness; no. of completed tasks; and others	Observations were arranged in temporal order, which were then used to define various behavioral indicators at problem and session levels

2.1.3.1 Conclusion

We have reviewed many existing research studies conducted to determine (student) engagement/disengagement in an ongoing task or process using their logged data with computer-based learning environments. Different approaches used in these works can be distinguished as *supervised*, *unsupervised* or *hybrid*. Supervised approach rely thoroughly on human experts to annotate students' activities or sequences of activities (also referred as 'traces') into engaged or disengaged behaviors, for example as done in [Bouvier et al., 2014, Brown and Howard, 2014, Tan et al., 2014]. Unsupervised approach rather use different data mining or machine learning methods to categorize students problem-solving activities into engagement/disengagement behaviors based on some parameters or rules identified earlier through data analysis, see for example [Beal et al., 2007, 2006, Cocea and Weibelzahl, 2006, Joseph, 2005, Pechenizkiy et al., 2009]. Whereas, hybrid approach combines both human experts' annotations and data mining or machine learning methods to construct student engagement detectors, for example [Cocea and Weibelzahl, 2007, Pardos et al., 2014].

Each approach has its own benefits and limitations and the selection of an appropriate approach is based on many factors which include but are not limited to: type of input data, objectives of the experimental work and the

available resources (e.g., amount of time, human experts, etc.).

Many of the existing works have used counts of performed actions (or activities) to estimate student engagement. Their results are quite interesting and offer valuable insights into the usage of the system. However, in our opinion, engagement is a behavioral construct that 'evolves' as a student (or user) progresses in a computer-based environment by performing a series of actions/activities. Therefore, it can be more interesting if the temporal order of students' actions is maintained and considered for making more accurate estimations about student engagement (for example, as done in [Beal et al., 2007, 2006, Bouvier et al., 2014, Tan et al., 2014]).

Furthermore, it is fascinating to construct automated engagement/disengagement detectors which may be used to classify behaviors of new students. For that, predicting students' future engagement behaviors is a pre-requisite to construct approximate detectors, but very few attempts are made in the existing literature; among them include the works of Cocea and Weibelzahl [2006, 2007].

2.2 Methods for Data Analytics

There are various data mining and machine learning methods to analyze sequential data, for example: sequential pattern mining [Guerra et al., 2014, Shanabrook et al., 2010] and (association) rule mining [Fournier-Viger et al., 2017, Romero et al., 2010], sequential clustering methods [Boroujeni and Dillenbourg, 2018, Köck and Paramythis, 2011], various deep learning techniques [Qiu et al., 2016], process mining techniques [Bogarín et al., 2018], etc. We did a quick critical analysis of these different methods for selecting the right set of techniques for our problem. Data mining techniques, for example: sequential pattern mining, association rule mining; are basically well suited for extracting useful patterns from the data that also require ample human intervention during data analysis and pattern/rule interpretation. Deep learning methods likes Artificial Neural Networks (ANN), Recurrent

Neural Networks (RNN), etc., which are quite trendy nowadays to learn complex relations from input data using multiple layers – require large data sets to train a model. These conditions do not apply to our subject problem, that is, our datasets were not large enough to train model using some deep learning techniques; neither we needed to learn some very complex relations from the data, as discussed in Section 3.5.2.4.

We needed a more sophisticated mechanism to model, analyze, predict and more importantly visualize student engagement/disengagement behaviors. Thus, we choose Markov chain, which is one of the popular methods in the family of probabilistic techniques. One big advantage of using Markov chain is that it allows modeling sequential data; preserves the temporal information between different observations as well as it reveals disguised relationships using probabilities. Another advantage is the visualization of a Markov chain which is represented by a directed graph. The visualization allows easy interpretation of intended behaviors of the students, which can be useful for the teachers, researchers, students (themselves, for feedback), practitioners, theorists, and other stakeholders.

However, a simple Markov chain is not always sufficient to capture the complex relationships (as it was the case in our work, see the next chapter for details). Therefore, we picked an advanced variant of it, called "Mixture Markov Model", which is supported by well-established efficient algorithm(s) for model training. In the following sections, we provide background information about Markov chain and mixture Markov model which are used in our work for modeling and predicting students' future behaviors.

2.3 Markov Chain

A Markov chain is a mathematical model to represent a stochastic or random process. In particular, we are talking about *discrete-time* stochastic process described by a (finite) sequence of *discrete* random events (or variables). Each event belongs to the set of possible outcomes of an experiment bounded

together with an occurrence of probability. That is, the outcome of a given experiment can affect the subsequent outcome [Grinstead and Snell, 2012].

Let's assume a set of states also called the *state space* of a chain, $S = \{s_0, s_1, ..., s_t\}$, representing the all possible discrete outcomes of an experiment. The process can be in one of these states at any time t where $t = \{0, 1, 2, ...\}$; starting from one of these states and moving successively from one state to another. If the chain is currently in state s_i and moves to another state s_j in the next time step, this can be represented by conditional probability p_{ij}, called as a *transition probability*. Alternatively, p_{ij} can be written as Eq. (2.1), which reads as the probability of moving from state i to state j equals the probability of state j given state i has already occurred.

$$p_{ij} = Pr(s_j \mid s_i) \tag{2.1}$$

The collection of transition probabilities of moving from one state to another including the *self* state (i.e. p_{ii}) are represented by a square matrix T of size *states* x *states*, and is referred as the *matrix of transition probabilities* or simply the *transition matrix*. Each state also has an associated initial probability, s_{init}, specifying the probability of a particular state as a starting state.

Thus, a Markov chain, M, can be defined as a mathematical model containing:

1. A state space represented by a finite set of states, $S = \{s_0, s_1, ..., s_t\}$.

2. A set of initial state probabilities, $S_{init} = \{s_{0_{init}}, s_{1_{init}}, ..., s_{t_{init}}\}$ specifying the probability for each state as a starting state. Note that the sum of all states initial probabilities (row vector) is equal to 1.

3. A transition matrix, T (as given below), wherein each cell containing a probability p_{ij} of transitioning from state i to state j. The initial state i corresponds to the i-th row in the matrix and the final state j to the j-th column. By the *law of total probability*, each row $i \in S$

must sums to 1, Eq. (2.2).

$$\mathbf{T} = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0t} \\ p_{10} & p_{11} & \cdots & p_{1t} \\ \vdots & \vdots & \ldots & \vdots \\ p_{t0} & p_{t1} & \cdots & p_{tt} \end{bmatrix}$$

$$\sum_{j \in S} p_{ij} = \sum_{j \in S} Pr(s_t = j \mid s_{t-1} = i) = 1 \tag{2.2}$$

Furthermore, a Markov chain must hold the *Markov property*, which states that the probability of a future state given the entire past only depends on the immediate past [Nicolas, 2013]. That is,

$$Pr(s_{t+1} \mid s_1, s_2, ..., s_t) = Pr(s_{t+1} \mid s_t) \tag{2.3}$$

for all $t \geq 0$.

This does not mean that all sequenced events are totally independent. For example, given a sequence of five states: $s_1 s_2 s_3 s_4 s_5$, it is not true that states s_5 and s_1 are independent. However, let's say, given s_4, s_5 is conditionally independent of s_1.

Another property that we are considering in Markov chains used in this work, is *time homogeneity*. In time-homogeneous Markov chain, probability distribution of each state remains the same for each time step t [Tolver, 2016], that is, $p_{ij}(t) = p_{ij}$ for all $t \geq 0$. From now on, we refer to *discrete time-homogeneous Markov chain* simply as a Markov chain. It is an extensive topic with many other properties and variations, which are out of the scope of this thesis. Therefore, in the following subsections, we only discuss the relevant subtopics which are important for understanding the construction of probabilistic models used in this work later. Now, let's consider a simple example to put these concepts into a practical situation.

Example 2.3.1 *A teacher gave a set of four exercises* $E = \{E_1, E_2, E_3, E_4\}$*, to the class which can be attempted in any order.*

That is, starting from any particular exercises, a student can attempt to solve any other exercise. All exercises have equally likely starting probability. We also assume that most of the students will follow a natural ordering of the exercises while solving them, that means the transition probabilities between subsequent states in increasing order is higher than that in the backward direction and jumps between different states. If we map this situation through a Markov chain, we get the following representative elements.

- *The set E with four exercises represents the state space S of the chain. Thus, each exercise is represented by a corresponding state $S = \{E_1, E_2, E_3, E_4\}$.*
- *We also know that all exercises have an equally likely starting probability, which are represented as:*

$$S_{init} = \{0.25, 0.25, 0.25, 0.25\}$$

- *An example transition matrix T for the given situation could be:*

$$\mathbf{T} = \begin{array}{c} \\ E_1 \\ E_2 \\ E_3 \\ E_4 \end{array} \begin{array}{c} \begin{array}{cccc} E_1 & E_2 & E_3 & E_4 \end{array} \\ \begin{pmatrix} .1 & .7 & .1 & .1 \\ .2 & .1 & .6 & .1 \\ .1 & .1 & .2 & .6 \\ .1 & .1 & .1 & .7 \end{pmatrix} \end{array}$$

Note that, a self-loop (or same state) transition probability reflects a situation when a student(s) may attempts the same exercise more than once.

2.3.1 Graphical representation

Another way of representing a Markov chain is through a directed graph, also known as "transition diagram". The states of a Markov chain are represented by nodes in the graph and transition probabilities between each

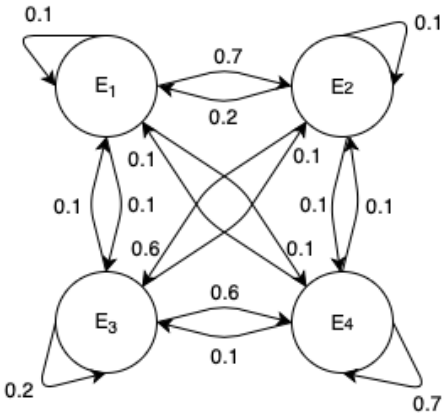


Figure 2.1: Transition diagram of Example 2.3.1 Markov chain

possible pair of states is shown by a directed edge. The label on each edge shows the transition probability between respective states. Transition diagram of Example 2.3.1 Markov chain is given in Fig. 2.1.

2.3.2 Order of a Markov chain

The order of a Markov chain refers to the number of previously observed states taken into account to determine the next or future state. A first-order chain relies just on the current state to determine the next state. The order of a chain can be increased to second-, third-, fourth-, ..., order; to increase its 'memory' for making better predictions. In general, a k-order Markov chain depends on "k" previous states to predict the next state, see Eq. (2.4).

$$Pr(s_{t+1} \mid s_1, s_2, ..., s_t) = Pr(s_{t+1} \mid s_{t-k}, s_{t-k+1}, ..., s_{t-1}) \qquad (2.4)$$

Note that, for a first-order Markov chain having S states, the transition matrix T is of size S^2 (i.e., $|S| \times |S|$). And, the size of the transition matrix increases exponentially for a higher-order chain. In general, a chain is considered to be a "first-order" Markov chain unless specified explicitly, which is one of the simplest yet effective mechanisms to represent sequential events.

2.3.3 Computing the probability of an input sequence

Let's consider an input sequence X of length L, comprises of random variables $X_1, X_2, ..., X_L$, representing some outcome of a discrete process. And, we want to compute its probability (or likelihood). From the chain rule of probability, we have:

$$Pr(X) = Pr(X_L|X_{L-1}, ..., X_1) \times Pr(X_{L-1}|X_{L-2}, ..., X_1) \times ... \times Pr(X_1) \tag{2.5}$$

By assuming that the probability of a variable is dependent only on the previous variable (i.e. Markov property), Eq. (2.5) becomes:

$$\begin{aligned} Pr(X) &= Pr(X_L|X_{L-1}) \times Pr(X_{L-1}|X_{L-2}) \times ... \times Pr(X_2|X_1) \times Pr(X_1) \\ &= Pr(X_1) \prod_{i=2}^{L} Pr(X_i|X_{i-1}) \end{aligned} \tag{2.6}$$

Eq. (2.6) denotes the probability of an input sequence X, where $Pr(X_1)$ represents the starting probability of element X_1.

2.3.4 Maximum likelihood estimation (MLE)

So far, we have described some basic definitions to understand Markov chains and the probability computation of an input sequence However, in a real-life scenario, we are not given a model representing some data distribution, but, instead, all we have is the input data. So, the real question is, how we can estimate the (Markov) model parameters using the input data?

The most commonly used approach for this purpose is "maximum likelihood estimation" (MLE), which makes the data D *look as likely as possible* under the model. Suppose, given some input sequences, we want to compute the transition probabilities between different states (each representing a unique symbol in the input sequences). We can compute the estimated

transition probabilities $\hat{p}_{ij}$ as:

$$\hat{p}_{ij} = \frac{n_{ij}}{\sum_{k=1}^{S} n_{ik}} \tag{2.7}$$

Where, n_{ij} is the number of transitions from state i to state j, divided by the total number of transitions from state i to all other states. Since the input data that we get is just a sub-sample of the original data, so we have to be careful while determining model parameters. For example, the absence of some of the symbol in the given input sample will lead to zero transition probability. But, knowing the problem domain, we usually do not want this particular behavior.

A Bayesian approach to overcome this problem is known as 'Laplace smoothing' in which probabilities are not strictly computed from the input data. But, instead, we start from some *prior* belief, for example, by adding a pseudo count of 1 to each symbol in the state space. With this, the smoothed Eq. (2.7) would become:

$$\hat{p}_{ij\ (smoothed)} = \frac{n_{ij} + 1}{\sum_{k=1}^{S} n_{ik} + n_{ij}} \tag{2.8}$$

However, we can replace the value 1 with any other constant value, Eq. (2.9) shows the generalized form.

$$\hat{p}_{ij\ (smoothed)} = \frac{n_{ij} + \alpha}{\sum_{k=1}^{S} n_{ik} + n_{ij}\alpha} \quad \text{for } \alpha > 0 \tag{2.9}$$

Example 2.3.2 *Let's consider the same real life scenario discussed in Example 2.3.1. However, this time we are not given a Markov chain distribution. Instead, all we get is some samples of input data of three students, as given below.*

Student 1: E_1, E_2, E_4
Student 2: E_1, E_1, E_2, E_4, E_4
Student 3: E_2, E_1, E_4, E_2

Note that, none of the students have attempted exercise E_3. Given these inputs, we want to construct a Markov chain representing students' approximate problem-solving behaviors. For that, we have to estimate transition probabilities between different states. But, instead of computing all the transitions, let's assume for now that we are interested in the followings.
$p(E_1 \mid E_1), \quad p(E_2 \mid E_1), \quad p(E_3 \mid E_1),$ *and* $\quad p(E_4 \mid E_1) = ?$

Using Eq. (2.7)*, we made the following calculations.*

$$p(E_1 \mid E_1) = \frac{1}{4}, \quad p(E_2 \mid E_1) = \frac{2}{4}, \quad p(E_3 \mid E_1) = \frac{0}{4}, \quad p(E_1 \mid E_1) = \frac{1}{4}$$

As we can see, $p(E_3 \mid E_1)$ is zero because of 'zero frequency' of exercise E_3 in the input data. Likewise, the same situation can occur for two states (s_1, s_2) where one state is never followed by the other, for example, if we try to compute $p(E_1 \mid E_2)$. This is an undesirable outcome since it will affect our future computation and analysis that we may perform for a new student using this obtained Markov chain. Hence, strictly relying on the input data is not useful if the estimated transition matrix contains many zero entries.

As mentioned earlier, Laplace smoothing is the commonly used technique to replace zero valued transition probabilities. We add a constant 1 and get the smoothed transition probabilities as follows, see Eq. (2.8)*.*

$$p(E_1 \mid E_1) = \frac{1+1}{4+4}, \quad p(E_2 \mid E_1) = \frac{2+1}{4+4},$$
$$p(E_3 \mid E_1) = \frac{0+1}{4+4}, \quad p(E_1 \mid E_1) = \frac{1+1}{4+4}$$

2.3.5 Prediction using Markov chains

Markov chains are not only useful for modeling sequential data, but we can also predict future state(s). More precisely, a future state at time $t+1$ can be calculated using the probabilities distribution computed at time t [Levin and Peres, 2017]. That is, given an initial state row vector μ_t which is a distribution of X_t and the transition matrix T, the next state transition probabilities are calculated using Eq. (2.10).

$$\mu_{t+1} = \mu_t \times T \quad \text{for } t \geq 0 \tag{2.10}$$

Each i-th entry in μ_{t+1} shows the probability of occurrence of i-th state in next time step. Thus, the state with the highest probability can be predicted as the next or future state, i.e., $argmax(\mu_{t+1})$.

2.4 Mixture Markov Model

As suggested by its name, mixture Markov model is a (finite) *mixture* of Markov chains yielded through a clustering method called "model-based clustering". This clustering method postulates a *generative statistical model* for the data which is optimized based on a likelihood (or posterior probability) [Meilă and Heckerman, 2001].

In problems related to *time series* data, distance-based clustering methods are not a good choice due to difficulty in defining appropriate distance functions between data elements [Pamminger et al., 2010]. Model-based clustering has gained popularity from over the last two decades for both continuous and discrete data as it identifies clusters based on their shape and structure of the data rather than proximity between data points [Meilă and Heckerman, 2001].

Given some input data, the clustering method finds model parameters that best fits the data according to some criterion. Consider a dataset $X = \{X_1, X_2, ..., X_N\}$ containing N independent, identically distributed observations. Each X_i is a sequence of L observations drawn from the discrete set of M symbols, i.e., $X_{iL} \in \{1, 2, ..., M\}$. The clustering method aims to identify K disjoint subsets of X, called mixture components (or clusters), containing subsets of sequences sharing similar properties. Mixture modeling framework assumes that data have been generated from K mixtures represented by the probability distribution function as shown in Eq. (2.11).

$$p(X_i \mid \Theta_K) = \sum_{k=1}^{K} \pi_k \; p_k(X_i \mid \theta_k) \tag{2.11}$$

Here, K denotes the total number of mixture components (or clusters)

and $\Theta_K = \{\pi_k, \theta_k\}$ represents the set of mixture parameters. Each π_k shows the mixing proportion (or prior probability) of the k-th component such that $0 \leq \pi_k \leq 1$ for all $k = 1, 2, ..., K$ and $\sum_{k=1}^{K} \pi_k = 1$. Every component has its own probability mass function represented by $p_k(X_i \mid \theta_k)$, whose parameters θ_k are to be estimated. We assume that all sequences grouped into different clusters are represented by a first-order Markov chain showing respective data distribution. As mentioned in Section 2.3, a Markov chain is defined by a set of initial state probabilities S_{init} and a transition matrix T, containing transition probabilities p_{ij} from state i to state j; $i, j \in \{1, 2, ..., M\}$ with M unique states. Thus, each model parameter θ_k is a stochastic Markov chain represented by an S_{init} vector of initial state probabilities and T matrix of transition probabilities. The probability mass function for k-th mixture is then written as in Eq. (2.12):

$$p_k(X_i \mid \theta_k) = Pr(X_{i1_{init}} = x_{i1}) \prod_{l=2}^{Li} Pr(X_{il} = x_{il} \mid X_{i(l-1)} = x_{i(l-1)}) \quad (2.12)$$

To simplify the notations, we denote the initial state probability as $\beta_n = Pr(X_{i1_{init}} = n)$ and transition probability $\gamma_{nm} = Pr(X_{il} = m \mid X_{i(l-1)} = n)$ with the following restrictions: $\sum_{n=1}^{M} \beta_n = 1$ and $\sum_{m=1}^{M} \gamma_{nm} = 1$ for $n = \{1, 2, ..., M\}$. Hence,

$$p_k(X_i \mid \theta_k) = \prod_{n=1}^{M} \beta_n^I \prod_{n=1}^{M} \prod_{m=1}^{M} \gamma_{nm}^{p_{inm}} \quad (2.13)$$

where $I = \begin{cases} 1 & \text{if } X_{i1} = n \\ 0 & \text{otherwise} \end{cases}$ and p_{inm} is the number of transitions from state n to state m in the sequence X_i. Now, assuming that each of the sequence is generated from one of the K components, Eq. (2.11) changes to

$$p(X_i, T_i \mid \Theta_K) = \sum_{k=1}^{K} \pi_k \prod_{n=1}^{M} \beta_n^I \prod_{n=1}^{M} \prod_{m=1}^{M} \gamma_{knm}^{p_{inm}} \quad (2.14)$$

for a $M \times M$ matrix T_i with elements p_{inm}.

Statisticians refer to model-based clustering as a mixture model of K components [Cadez et al., 2003] and, in the literature, both terms are often used interchangeably. However, model-based clustering requires an additional step than just finding a finite mixture model, that is, to assign each sequence to its appropriate cluster from K mixtures based on a pre-specified rule [Melnykov et al., 2010]. *Bayes decision rule* is the most commonly used method for this purpose, e.g. Melnykov [2016], Pamminger et al. [2010]; all used the same method. Bayes rule assigns each sequence X_i to the cluster k that has the maximum *posterior probability* value, given in Eq. (2.15).

$$Pr(k \mid X_i) = \frac{\pi_k \; p_k(X_i \mid \theta_k)}{p(X_i \mid \Theta_K)} \tag{2.15}$$

Usually, the model-based clustering involves estimating both the number of mixture components and model parameters for each mixture that best defines the given data, unless the input data is synthetic wherein the number of distributions generating the artificial data is known at prior. This problem of estimating the cluster parameters or structure in the absence of any other information except the given data is known as "*cluster analysis*" [Fraley and Raftery, 1998]. In the followings, we first discuss different methods available for determining the model structure or number of mixture components. Next, we provide details of the Expectation-Maximization algorithm which is an efficient framework for estimating mixture model parameters.

2.4.1 Choosing the number of clusters

Like K-means, model-based clustering also requires the user to specify a prior number of mixtures which is one of the challenging problems for researchers. However, model-based clustering has the advantage of being supported by formal statistical methods to determine the number of clusters and model parameters [Magidson and Vermunt, 2002]. The two most commonly used methods which are based on *information criterion* to select the optimal value of K are Bayesian Information Criterion (BIC) [Schwarz et al., 1978]

and Akaike Information Criterion (AIC) [Akaike, 1998]. Since both methods are based on information criterion, so a lower BIC or AIC score means less information lost and hence a better-fitted model. Eq. (2.16) and (2.17) respectively show the formulas for calculating BIC and AIC scores.

$$BIC = -2\ln(\mathcal{L}) + p\ln(n) \tag{2.16}$$

$$AIC = -2\ln(\mathcal{L}) + 2p \tag{2.17}$$

In the above equations, p is the number of free parameters to be estimated, n is the sample size and $\mathcal{L}$ is the maximized likelihood value of the estimated model.

Both measures are used for model selection amongst a finite set of models. However, while fitting models to the input data, the likelihood can be increased by adding parameters, but doing so may result in overfitting. The BIC resolves this issue by introducing a penalty term for the number of parameters in the model, see the right-most term in Eq. (2.16). Whereas, AIC is independent of the sample size. Hence, the primary difference between both measures is that BIC penalizes heavily in contrast to AIC. Dziak et al. [2019] has discussed in detail some other variants of these two measures (e.g., CAIC, adjusted BIC) and several related issues; that provides additional information for choosing a suitable measure for the given problem.

Another approach to determine the optimal value of K is based on the Bayesian method, for example as used by Meilă and Heckerman [2001]. The method uses (log) posterior probability of model structure given the training data, $\log P(K \mid D_{train})$. Using Bayes' formula, the value is computed as:

$$P(K \mid D_{train}) = \frac{P(K)P(D_{train} \mid K)}{P(D_{train})} \tag{2.18}$$

By assuming equal prior probability for each model structure, the formula reduces to the (log) *marginal likelihood* of the model structure $\log P(D_{train} \mid K)$.

Grim [2006] proposed a heuristic approach to identify the unique number of clusters sequentially for model-based clustering performed on categorical

data. In particular, initially starting with a single component, a new component is added to the estimated model sequentially and initialized as a product of uni variate uniform distribution with equal initial weight. The process is repeated until the algorithm converges.

2.4.2 Parameters estimation: Expectation-Maximization algorithm

Expectation-Maximization (EM) algorithm introduced by Dempster et al. [1977], is a well-known iterative procedure to estimate finite mixture model parameters by maximizing the likelihood of observing a *complete data*. More precisely, mixture modeling framework assumes that each sequence X_i is generated by one of the K component distributions, however, its true membership label is unknown [Melnykov, 2016]. The EM algorithm aims to incorporate these missing labels.

As before, let's assume that the input data $X = \{X_1, X_2, ..., X_N\}$ contains sequences of observations which are generated by some distribution(s). We call X the "incomplete data" and assume that a "complete data" set exists, $Y = (Y_1, Y_2, ..., Y_N) = ((X_1, Z_1), (X_2, Z_2), ..., (X_N, Z_N))$ where $Z_i = \{z_{i1}, z_{i2}, ..., z_{iK}\}$ represents the "missing data" [Fraley and Raftery, 1998], with

$$z_{ik} = \begin{cases} 1 & \text{if } X_i \text{ belongs to component } k \\ 0 & \text{otherwise} \end{cases} \tag{2.19}$$

The incomplete (or observed) data log-likelihood is:

$$\begin{aligned} log(\mathcal{L}(\Theta \mid X)) &= \log \prod_{i=1}^{N} p(X_i \mid \Theta) \\ &= \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \; p_k(X_i \mid \theta_k) \right) \end{aligned} \tag{2.20}$$

which is difficult to optimize as it contains the log of the sum [Bilmes et al., 1998]. Now, considering the existence of unobserved data, the mass

function of an observation X_i given Z_i is computed as $\prod_{k=1}^{K} p_k(X_i \mid \theta_k)^{z_{ik}}$ assuming that each missing variable Z_i is independent and identically distributed having initial probabilities $\pi_1, \pi_2, ..., \pi_K$. Hence, the complete-data log-likelihood becomes:

$$\begin{aligned} \log(\mathcal{L}(\Theta \mid X, Z)) &= \log\ (p(X, Z \mid \Theta)) \\ &= \sum_{i=1}^{N} \sum_{k=1}^{K} z_{ik}\ [\log\ \ \pi_k\ p_k(X_i \mid \theta_k)] \end{aligned} \tag{2.21}$$

To maximize this complete-data log-likelihood, expectation-maximization algorithm iterates over the two steps namely, *expectation step* and *maximization step*; until it reaches convergence (or some stopping criterion). As mentioned by Gupta et al. [2011], the general idea is that we make an initial guess about complete data Y and solve for the θ that maximizes its log-likelihood. And, this estimated model θ can be maximized by making a better guess about the complete data Y in an iterative manner. The two steps are given below.

1. Expectation (or E) step: estimates the conditional expectation values z_{ik} of complete-data log-likelihood function given the observed data (i.e. posterior probabilities of the hidden variables).

2. Maximization (or M) step: finds the parameter estimates to maximize the complete-data log-likelihood from the E-step.

Mathematically, these are defined below in Eq. (2.22) and (2.23), respectively.

E-step:

$$z_{ik} = \frac{\pi_k\ p_k(X_i \mid \theta_k)}{\sum_{k'=1}^{K} \pi_{k'}\ p_{k'}(X_i \mid \theta_{k'})} \tag{2.22}$$

The probability distribution function is already defined in Eq. (2.13).

M-step:

$$\pi_k = \frac{1}{N}\sum_{i=1}^{N} z_{ik} \ , \quad \beta_{kn} = \frac{\sum_{i=1}^{N} z_{ik}\ I(X_{i1} = n)}{\sum_{i=1}^{N} z_{ik}} \ , \quad \gamma_{knm} = \frac{\sum_{i=1}^{N} z_{ik}\ p_{inm}}{\sum_{i=1}^{N} z_{ik}\ \ \sum_{m'=1}^{p} p_{inm'}} \tag{2.23}$$

The EM algorithm guarantees to converge the (log) likelihood function to one of the local maximas and never gets worse [Bilmes et al., 1998, Gupta et al., 2011] than the previous iteration. In this respect, the algorithm is considered to converge when the difference between the (log) likelihood value of two subsequent iterations goes down a *user-defined* threshold. This is the most commonly used approach for stopping the EM algorithm [Melnykov et al., 2010]. Another method to set the stopping criteria for the EM algorithm is to define a maximum number of iterations as a threshold value.

Our data is a collection of discrete events ordered temporally into variable length sequences, each representing the problem-solving activities performed by the students while interacting with a computer-based assessment system. Hence, we focused on the version of the EM algorithm specific to categorical data. In this work, we also assume that each mixture component is represented by a first-order time-homogeneous Markov chain. This is also sometimes referred as Markov chain clustering [Pamminger et al., 2010], in which the probability distribution of each mixture component is represented by a first-order transition matrix. To cluster multivariate categorical data, EM algorithm requires the following three parameters to get started.

- Number of mixtures (K)
- Initial transition matrices for K mixtures
- Initial weights of K mixtures

One critical issue associated with the EM algorithm is that it relies on the initial values of the mixture parameters to identify homogeneous mixture

components for multivariate data. And, initialization of these input parameters play a key role in the performance of the EM algorithm [Hu, 2015, Melnykov et al., 2010, Michael and Melnykov, 2016]. In the following, we highlight different approaches used for this purpose.

2.4.2.1 EM initialization

Standard EM algorithm initializes the 'initial transition matrices' randomly for the K given by the user. And, each mixture component is usually assigned an equal initial weight (i.e. $w_1 = \cdots = w_K = 1/K$), e.g., as done in the EM algorithm implementation by Melnykov [2016] [3]. To get even better starting parameters, it is common to start the EM algorithm with multiple random initial guesses and choose the model θ with highest likelihood [Gupta et al., 2011].

The idea of running a preliminary clustering algorithm on the input data has also gained popularity for the EM initialization. For example, Hu [2015] has reviewed many works utilizing different clustering methods to initialize the EM algorithm like K-means, K-means++, complete linkage hierarchical clustering method, and other variants. Their own proposed scheme called "Combined K-means Data Segments" (CKDS) is an improvement over simple K-means algorithm used previously for parameters initialization of the EM algorithm.

Gupta et al. [2011] also used K-means clustering algorithm for EM initialization to find the Gaussian mixtures which contain continuous data and each mixture component is represented by some Gaussian distribution. Fraley and Raftery [1998] relied on an agglomerative hierarchical clustering to approximate the initial parameters before running the actual EM algorithm.

In a similar vein, many variants of the EM algorithm are proposed to further improve the quality of resultant mixtures and most of these methods

[3] They implemented a variant of the EM algorithm called emEM in an R package named as ClickClust. We used the same package as a baseline to perform experiments as it will be explained in the next chapter.

focused on improving the EM algorithm initialization. "emEM" proposed by Biernacki et al. [2003] is one of these popular methods. The small or first *em* represents the execution of expectation-maximization algorithm in the *initialization* phase. The result of small em is then used to start the actual EM algorithm. To further improve the quality of the emEM algorithm, Michael and Melnykov [2016] proposed an effective method called "emaEM". The new approach is based on a model averaging technique to incorporate the output of different models generated at each iteration instead of picking one *best* model.

Although, many approaches have been developed to produce better initial parameters for the EM algorithm, which are critical to estimate optimal parameters of the resultant model. But, most of these methods are targeted for Gaussian Mixture Model (GMM) and not applicable to the problem domain of this research work. In fact, this issue is not investigated thoroughly for multivariate categorical data and yet an open problem for the researchers.

3

Objectives, Methodology, and Results

In this chapter, details of the practical steps that we took to find answers for the research questions (RQs, mentioned in Chapter 1), are presented through the following sections.

- In the first section, we describe our research objectives defined in relation to the two research questions.
- The second section is reserved for the details of the two experimental studies conducted using computer-based assessment tools, which recorded students' interactions and their performance parameters. We also describe the form of the raw data retrieved to perform different analyses at later stages.
- The subsequent three sections contain details of the research studies carried out for each research objective and the obtained empirical results and findings.

The materials used in this chapter are primarily taken from the following published papers, presented at different venues.

1. "Rabia Maqsood and Paolo Ceravolo. Modeling behavioral dynamics in confidence-based assessment. In 2018 IEEE 18th International

Conference on Advanced Learning Technologies (ICALT), pages 452—454. IEEE, 2018".

2. "Rabia Maqsood and Paolo Ceravolo. Corrective feedback and its implications on students' confidence-based assessment. In International Conference on Technology Enhanced Assessment. Springer, 2018".

3. "Rabia Maqsood, Paolo Ceravolo, and Sebastián Ventura. Discovering students' engagement behaviors in confidence-based assessment. In 2019 IEEE Global Engineering Education Conference (EDUCON), pages 841—846. IEEE, 2019".

4. "Rabia Maqsood, Paolo Ceravolo, Cristobal Romero, and Sebastián Ventura. Modeling and predicting students' engagement behaviors: A new approach for mixture Markov models. (Under review)".

3.1 Research Objectives

By keeping in view our research questions (RQs, given in Section 1.4) and the background knowledge of the related concepts (given in Chapter 2), we identified some research objectives that are presented below with our reasoning.

The first research question (RQ-1) was: "*What behaviors can be used to determine student engagement/disengagement in confidence-based assessment?*".

The study of the existing works has brought forward various theoretical and non-theoretical approaches for determining student engagement; including the possibility of examining students' activities to categorize their behaviors, introduced by Fredricks et al. [2004] as 'behavioral engagement'. It is sometimes also referred as 'online engagement' when the input data is coming from students' interactions with computer-based learning environments. We found it to be most relevant for our intended goals. To measure online or behavioral engagement, prior empirical studies have commonly used 'time-on-task' and 'response correctness' parameters, see Table 2.2. But, we needed to identify some additional parameter(s) for determining student engagement in confidence-based assessment, which also takes into account the 'confidence level' a student has in the submitted answers. Inclusion of this particular parameter in engagement detection model will supplement it with new information about a student's confidence accuracy, which is crucial for differentiating between students (see research gaps in Section 1.3). Thus, we defined the following two research objectives.

1. Investigate the correlation of different student performance parameters with the confidence-based assessment.

2. Define a scheme to classify students' activities into engagement or disengagement behaviors.

The second research question (RQ-2) was: "*How can we model and predict*

these behaviors to construct students' behavioral profiles?".

Modeling or representing the identified behaviors using a *suitable* technique was our next challenge. By 'suitable', we mean a technique that allows us to represent and analyze the identified behaviors of the students, as well as make predictions about their intended future behaviors; which are the key requirements for constructing students behavioral profiles. Furthermore, the selected technique should allow easy interpretation of various engagement and disengagement behaviors; so that students with diverse strengths and weaknesses can be identified. These behavioral profiles can be utilized in future to provide appropriate support and guidance by class teachers or an adaptive system (if our proposed solution is implemented in a real system; which is one of the future work directions of this thesis, see next chapter). After studying and analyzing various data mining and machine learning methods that consider temporal ordering between data items, for example: sequential pattern mining [Fournier-Viger et al., 2017, Guerra et al., 2014, Shanabrook et al., 2010] and sequential rule mining [Cohen and Beal, 2009, Fournier-Viger et al., 2017], sequential clustering methods [Boroujeni and Dillenbourg, 2018, Köck and Paramythis, 2011]; and process mining techniques [Bogarín et al., 2018]; we selected probabilistic modeling methods due to their successful application in sequential data modeling and prediction tasks. Furthermore, the support of well established statistical methods increases the validity of results obtained through probabilistic modeling approaches.Thus, our third research objective was to:

3. Model, analyze and predict students varying engagement and disengagement behaviors using probabilistic models.

For each research objective, we performed an empirical research study by taking students' logged data as input and used various methods to analyze and perform operations on the data. Before describing our methodology for the three carried research studies and the obtained outcomes, we first present details of the data collection process.

3.2 Experimental Studies

In order to collect data, we conducted two experimental studies with undergraduate students taking introductory programming courses; who participated in confidence-based assessment using the computer-based tools. The first experimental study involved 94 freshmen of the National University of Computer and Emerging Sciences, Pakistan, while the second one was held with 210 undergraduate students of the Universitá degli Studi di Milano, Italy.

In the following subsections, we provide details of the two experimental studies, including their design, the tools used and parameters of the collected datasets.

3.2.1 Design

3.2.1.1 The first experimental study

In this study, an existing code-tracing tool called 'CodeMem'[1] was used to deliver confidence driven questions to the students. The tool was developed for evaluating code tracing skills of the students learning C/C++. Some snapshots of the tool and sample questions are given in Appendix A.1.

Three sessions of 40-45 minutes each were conducted in different weeks and students were given six (code tracing) problems per session in a self-assessment setting, that is, no time limit was specified for any question and there was no impact on students' course records based on their participation and/or performance in this study. Each session consisted of questions related to one topic, more specifically, questions were designed from the following three topics: basic operators *(variable initialization, arithmetic operators)*, selection statements *(if-else)* and repetition *(while loop)*, respec-

[1]Developed by a team of three students from NUCES-CFD (Pakistan), under the supervision of the principal investigator of this research study. The tool was modified to incorporate objectives of the experimental study.

tively. The questions were designed carefully (by the principal investigator of this research study) to maintain difficulty levels from easy to medium.

Students were asked to specify their confidence level (as high or low)[2] before submitting a solution. In fact, two submit buttons ('High confidence submit' and 'Low confidence submit') were available *(on student portal)* so that students can make a conscious choice of their confidence level for each answer. Moreover, students were allowed to freely navigate the system and attempt a question multiple times before making a final submission. The assessment model used for designing the tool and data collection is given in Section 3.2.2.1.

3.2.1.2 The second experimental study

In this study, another tool called 'QuizConf'[3] was used to deliver confidence driven questions to the students. The tool was designed to facilitate students for assessing their problem-solving skills in an introductory programming course. Some snapshots of the tool and sample questions are given in Appendix A.1.

This study was also conducted for students' self-assessment purposes, however, with relatively different settings. The class teacher uploaded 39 multiple choice questions related to basic concepts of an introductory programming course. More specifically, 13 different exercises were uploaded with code flow diagrams. Each exercise contained 3 multiple choice questions, each on a separate page. The students were asked to use the tool for their self-assessment and preparation of the final examination. As before, they were required to specify confidence level (as high or low) with each submitted response using a dedicated submit button (i.e. 'High confidence

[2]We used binary scale for confidence measurement instead of a more complex rating (e.g. as mentioned in Section 1.1), which may confuse students in estimating their confidence about solution's correctness [Petr, 2000, Vasilyeva et al., 2008].

[3]Developed by the principal investigator of this research study solely to collect data for this research work.

submit' and 'Low confidence submit'). This tool was also designed by following the same assessment model as the one used for data collection in the first experimental study, see Section 3.2.2.1 for details.

3.2.2 Computer-based assessment

The utility to evaluate numerous students at a time has increased the usage of computer-based assessment (CBA) systems largely in blended learning for both summative and formative assessment of the students [Thelwall, 2000]. The type of questions usually offered in CBA systems include multiple choice, true/false and fill-in-the-blank questions. The answers submitted by hundreds of student can be evaluated in just a few seconds through comparison with pre-defined answers for each question type, which increases its popularity. Another prominent feature of CBA systems is their capability of logging students' performance data and their interactions with the tool during assessment. This logged data can be exploited to analyze and determine students' performance outcomes and approximate behavior using various machine learning and data mining techniques.

In this work, data generated through student-system interactions with the two CBA tools (details of which are given in Appendix A.1), offering confidence-based questions, was exploited to examine students behaviors during the assessment. However, we first needed to define an assessment model to show the possible set of activities that students may perform by interacting with the tools and to record their data accordingly.

3.2.2.1 The assessment model

We constructed an assessment model, see Fig. 3.1, by considering the general activities offered in a traditional CBA system and linked binary confidence measures (i.e. high and low) with answer submission activity. The model helps us to visualize assessment as a process; containing all the activities (shown as nodes) which a student may perform before or after submitting

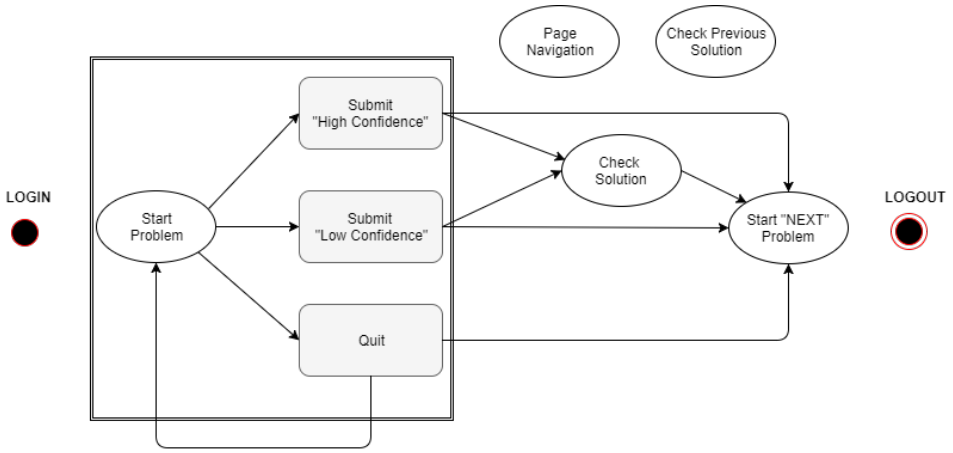


Figure 3.1: Model followed in a general CBA system with confidence measure

an answer. And, a student repeats the same cycle multiple times in a session (while solving a set of exercises by *Start Next Problem*). Directional arrows represent possible navigational flow between subsequent activities.

In this model '*Login*' and '*Logout*' are acting as start and end activities, respectively; to represent one session of a student. Once a problem has started (by '*Start Problem*'), it may be submitted with high or low confidence (i.e. '*Submit High Confidence*' or '*Submit Low Confidence*' activities, respectively). A problem may be attempted any number of times in case of a '*Quit*' before making a final submission (see reverse directional arrow from *Quit* to *Start Problem* activity, this shows reattempting the same problem).

Each student is informed immediately of his/her response's correctness (i.e. either correct or incorrect) upon submission of an answer. But, the correct solution with elaborated feedback is offered only upon a student's explicit request through '*Check Solution*' activity for a recently submitted problem. Whereas, a solution for some previously submitted problems can be requested using '*Check Previous Solution*' activity to revise pre-learned material from same or other topics that may be useful in answering a current question. Furthermore, there are other activities usually available in a CBA

system, but they do not contribute to the learning process, which is grouped as '*Page Navigation*'; for example: change the password and/or personal information, visit the home page, etc.

Note that, '*Page Navigation*' and '*Check Previous Solution*' activities are not connected with any other activity node in the model. This shows that these activities may be performed in any order and any number of times before and/or after starting and submitting a problem. Also, *Start Problem*, *Submit..*, and *Quit* activities are enclosed in a double-lined square box to show strict atomicity of the problem-solving process, that is, no other activity can be performed while a student is solving a problem.

As it can be seen, the assessment model is kept as general as possible to increase the viability of our empirical results, achieved through different experiments. It will allow comparing our results with a larger pool of studies that may have been developed using other CBA tools.

3.2.3 Collected data description and pre-processing

As mentioned earlier, the assessment model shown in Fig. 3.1 was implemented in the two CBA systems (used in the experimental studies) to collect students' data. Each activity was recorded with a timestamp along with some other parameters that are useful to identify the students and different problem-solving sessions. These include: a unique 'student id', 'session id', 'activity name', 'time stamp' (data and time), 'activity time spent', 'question id', 'response correctness', 'confidence level', 'web page url' and 'email id'. A sample of collected raw data (containing only the important variables)[4] is shown in Fig. 3.2.

We refer to the two datasets obtained from the first and second experimental studies respectively as "Dataset1" and "Dataset2". The total number of records contained in both datasets are 18,172 and 79,517.

[4]Student's email ids, web page URLs and session ids are not shown here. Also, student ids are anonymized for privacy reasons.

id	Activity	Activity label	Time stamp	Time spend	Question id	Response correctness	Confidence level
St3	Login	Login	24/11/2017 05:10	00:00:00	NULL	NULL	NULL
St3	Home Page	Page Navigation	24/11/2017 05:10	00:00:06	NULL	NULL	NULL
St3	View Assignments	Page Navigation	24/11/2017 05:10	00:00:03	NULL	NULL	NULL
St3	Start Problem	Start Problem	24/11/2017 05:10	00:08:45	160	NULL	NULL
St3	submit high confidence	submit high confidence	24/11/2017 05:19	00:08:45	160	correct	high
St3	View Assignments	Page Navigation	24/11/2017 05:19	00:00:19	NULL	NULL	NULL
St3	Start Next Problem	Start Next Problem	24/11/2017 05:19	00:01:49	161	NULL	NULL
St1	Login	Login	24/11/2017 05:21	00:00:00	NULL	NULL	NULL
St1	Home Page	Page Navigation	24/11/2017 05:21	00:00:02	NULL	NULL	NULL
St1	View Assignments	Page Navigation	24/11/2017 05:21	00:00:23	NULL	NULL	NULL
St1	Start Problem	Start Problem	24/11/2017 05:21	00:15:51	160	NULL	NULL
St3	submit high confidence	submit high confidence	24/11/2017 05:21	00:01:49	161	incorrect	high
St3	View Assignments	Page Navigation	24/11/2017 05:21	00:00:09	NULL	NULL	NULL
St2	Login	Login	24/11/2017 05:21	00:00:00	NULL	NULL	NULL
St2	Home Page	Page Navigation	24/11/2017 05:21	00:00:10	NULL	NULL	NULL
St3	Check Solution	Check Solution	24/11/2017 05:21	00:00:14	161	NULL	NULL
St2	View Assignments	Page Navigation	24/11/2017 05:21	00:00:11	NULL	NULL	NULL
St3	View Assignments	Page Navigation	24/11/2017 05:22	00:00:03	NULL	NULL	NULL
St3	Start Next Problem	Start Next Problem	24/11/2017 05:22	00:09:15	162	NULL	NULL
St2	Start Problem	Start Problem	24/11/2017 05:22	00:05:29	160	NULL	NULL
St2	submit low confidence	submit low confidence	24/11/2017 05:27	00:05:29	160	correct	low
St2	View Assignments	Page Navigation	24/11/2017 05:27	00:00:03	NULL	NULL	NULL
St2	Start Next Problem	Start Next Problem	24/11/2017 05:27	00:06:33	161	NULL	NULL
St3	submit low confidence	submit low confidence	24/11/2017 05:31	00:09:15	162	correct	low
St3	View Assignments	Page Navigation	24/11/2017 05:31	00:00:07	NULL	NULL	NULL
St3	Start Next Problem	Start Next Problem	24/11/2017 05:31	00:00:05	163	NULL	NULL
St3	View Assignments	Page Navigation	24/11/2017 05:31	00:00:02	NULL	NULL	NULL
St3	Check Previous Solution	Check Previous Solution	24/11/2017 05:31	00:00:19	161	NULL	NULL
St3	View Assignments	Page Navigation	24/11/2017 05:31	00:00:02	NULL	NULL	NULL
St2	submit high confidence	submit high confidence	24/11/2017 05:34	00:06:33	161	correct	high
St2	View Assignments	Page Navigation	24/11/2017 05:34	00:00:03	NULL	NULL	NULL
St2	Start Next Problem	Start Next Problem	24/11/2017 05:34	00:06:10	162	NULL	NULL
St1	submit high confidence	submit high confidence	24/11/2017 05:37	00:15:51	160	correct	high
St1	View Assignments	Page Navigation	24/11/2017 05:37	00:00:33	NULL	NULL	NULL

Figure 3.2: Sample raw data collected from one of the CBA tools

Next, to create students problem-solving profiles, all records were arranged by a tuple <session id; student id; time stamp>. This gives us lists of unique sessions of the students, containing all the activities performed in a 'Login-Logout' session in temporal order. The total number of unique 'Login-Logout' sessions contained in the Dataset1 and Dataset2 are 296 and 771, respectively. This data was further processed and analyzed to achieve our research objectives, as described in the following sections.

3.3 Objective I: Investigate the Correlation of Different Student Performance Parameters with Confidence-based Assessment

The description of the assessment model given in Section 3.2.2.1 highlights a number of student performance parameters recorded by the two assessment tools, including: student's confidence level, response correctness, time spent on a problem, check solution (or feedback, for the recently submitted problem) and check previous solution. Among these, 'response correctness' and 'time spent on a problem' are the most commonly used parameters that reflect a students' active or inactive involvement in a learning process (refer to Table 2.2). As mentioned in Section 3.1, our concern was to identify additional parameter(s) (if exists) that may be useful to reflect students behaviors and study the correlation of all these potential parameters (of student engagement detection model) with confidence-based assessment.

Our research revealed that the role of automated feedback provided (to students) in confidence-based assessment has been studied for over 30 years, for different purposes [Kulhavy and Stock, 1989, Mory, 1994, Timmers et al., 2013, Van der Kleij et al., 2012, 2015, Vasilyeva et al., 2008]. The earliest study by Kulhavy and Stock [1989] reported students' different usage of feedback based on their confidence level and actual answer. In particular, students with high confidence and wrong answers spent more time on read-

ing feedback, whereas, feedback gained less importance in case of correct answers given with high confidence. Similar results were found by Mory [1994] and Vasilyeva et al. [2008], when feedback reading time is compared in relation to different confidence levels; and, this information is used to provide adaptive feedback to the students based on their different needs (evident from their respective confidence level and response correctness). These studies show that automated feedback, usually provided in CBA systems, has been perceived differently by students having distinct confidence level and response correctness.

Student's response outcomes (correct/incorrect) in combination with confidence levels (high/low) provide four knowledge regions namely: doubt, mastery, uninformed and misinformed; as mentioned in Section 1.1. We referred to these regions as "confidence-outcome categories", and borrowed their names' abbreviations from Vasilyeva et al. [2008]: high confidence - correct response (HCCR), high confidence - wrong response (HCWR), low confidence - correct response (LCCR) and low confidence - wrong response (LCWR)[5].

These distinct confidence-outcome categories capture a discrepancy between students' confidence (that reflects his/her expected performance) and the actual performance they achieved. This discrepancy or knowledge gap can be filled through correct information that is usually offered to the students through *task-level* feedback in a CBA system. According to the lifelong learning perspectives, one of the goals of a learning environment is to foster students' perseverance and determination. In this respect, 'feedback' (given to the students) offers a paramount opportunity to induce or inspire a positive continuation of the learning process. Appropriate utilization of this feedback is indispensable for performing self-reflection which is an im-

[5]Alternative terminologies are available in the literature. For example, Hunt [2003] distinguished these knowledge regions as: uninformed (wrong answer with low confidence); doubt (correct answer with low confidence); misinformed (wrong answer with high confidence); and, mastery (correct answer with high confidence).

portant ingredient for leveraging students self-assessment process [McMillan and Hearn, 2008]. In this respect, being able to identify students' varying behaviors towards the available "corrective feedback"[6] could be useful to determine student *engagement/disengagement* during assessment and thus, support adaptation in a confidence-based assessment system [Maqsood and Ceravolo, 2018]. However, a preliminary step was to establish that feedback-seeking is correlated with distinct confidence-outcome categories and it has a positive impact on students' learning.

Before moving to our first research study, we provide a brief overview of automated feedback types commonly offered in CBA systems in general and the ones offered in the tool used in our research work for data collection.

3.3.1 Different feedback types

Computer-based assessment systems allow for automating multiple types of feedback. In case of formative assessment, various types of *task-level* feedback are discussed in Shute [2008]. However, we consider the following three most commonly used feedbacks which are offered to enhance students' understanding of their knowledge level and misconceptions they may have in the subject matter.

- Knowledge of Result (KR): notifying if the student's answer is correct or incorrect.
- Knowledge of Correct Response (KCR): providing a correct solution to the student.
- Elaborated Feedback (EF): a detailed explanation about the correct response that may additionally discuss the merits of the wrong answer given by the student.

[6] *Task-level* feedback allowing students to fill knowledge gap(s) in one's understanding of subject material; described in more detail in Section 3.3.2.

Several research studies conducted in the past have compared the usefulness of these feedback types from different perspectives. For example, findings of Van der Kleij et al. [2012] showed that KCR and EF are more favorably perceived by the students when offered in an immediate context (i.e implicitly given after each response submission) as compared to delayed settings (i.e. provided upon student's request). In addition to that, EF feedback has proved to have a higher impact on students' learning outcomes as compared to KR and KCR feedback types [Van der Kleij et al., 2015]. The experimental study conducted in [Timmers et al., 2013] investigated the link between students' motivational beliefs, effort invested during the assessment and students' behavior towards feedback provided by a CBA system. Their results indicate that feedback-seeking is predicted by success expectancy, task-value beliefs and the student effort invested in the formative assessment. Readers are redirected to the work of Hattie and Timperley [2007], Mory [2004] and Shute [2008], for a comprehensive discussion on designing appropriate feedback types in different assessment approaches.

3.3.2 Feedback types offered in the CBA tools used for data collection

The assessment model (given in Fig. 3.1), that we used for collecting data using the two CBA tools, shows a 'check solution' activity which informs a student about the correct solution along with detailed feedback for a recently submitted problem. In relevance to the different feedback types discussed in the previous section, we can say that the feedback that we provided to the students through 'check solution' activity is a combination of knowledge of correct response (KCR) and elaborated feedback (EF). We believe that a correct response along with brief explanation or comparison of the correct solution with a student's original response will serve the essential purpose of feedback, that is, to fill knowledge gap(s). We referred to this combination of feedback as "corrective feedback" (CF), which is originally defined by

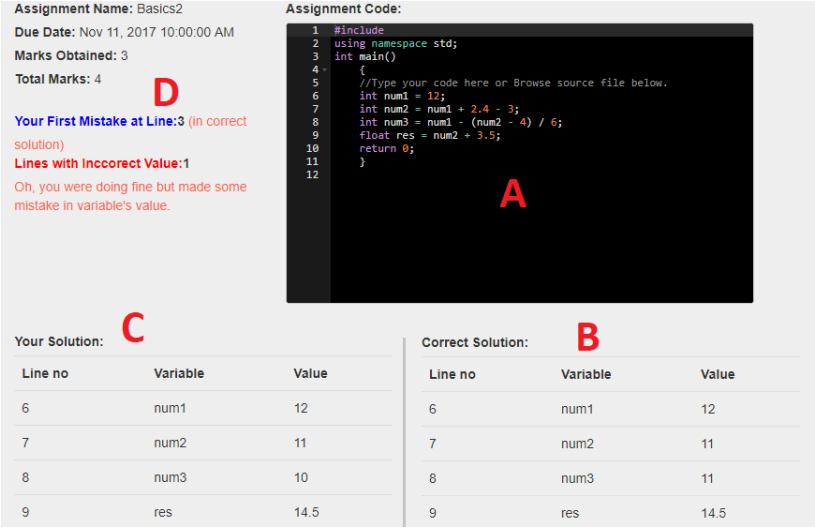


Figure 3.3: An example (screenshot) of corrective feedback provided to a student in the first experimental study, using the CodeMem tool. The feedback page contains four labeled sections (A: shows code snippet; B: shows correct solution auto-generated by the tool; C: shows student's submitted solution; D: explanation/error(s) highlighting area)

knowledge of result (KR) feedback in [Hattie and Timperley, 2007]. The corrective feedback (CF) was available upon student's explicit request by clicking on a dedicated button *(to display correct solution along with the student's submitted solution for mistakes identification and filling knowledge gap)*. An example of CF[7] provided to a student in the first experimental study (using the CodeMem tool) is shown in Fig. 3.3. An example of CF offered to a student participated in the second experimental study (using the QuizConf tool) is shown in Fig. 3.4.

As mentioned in Section 3.2.2.1, the tools also provided knowledge of result (KR) feedback implicitly to all the students after the submission of each answer (example screenshots of KR feedback are given in Appendix A.1).

[7]We avoided textual explanation of the correct solution and instead highlighted student's error(s) for easy comparison with the correct solution.

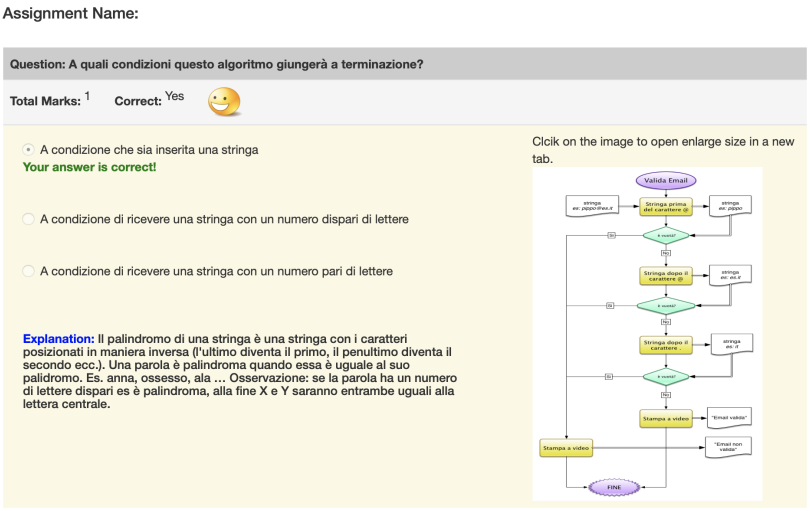


Figure 3.4: An example (screen-shot) of corrective feedback provided to a student in the second experimental study, using the QuizConf tool. The student's selected answer is labeled as 'correct' in this case; and a detailed textual feedback is provided at the bottom.

The feedback notify to the student if the (last) submitted solution was correct or incorrect. Now, the student can either request for the corresponding corrective feedback or ignore that at all. Hence, analyzing students' response towards the available corrective feedback can offer useful insights to understand their behaviors during confidence-based assessment. This defines the purpose of our first research study; details of the methodology and obtained results are presented in the subsequent sections. From here onward, we will refer to students' response towards corrective feedback simply as feedback-seeking/no-seeking.

3.3.3 Research study I

To perform empirical analyses of students' behaviors towards corrective feedback concerning distinct confidence-outcome categories, we exploited student-system interactions obtained from the first experimental study (de-

scribed in Section 3.2.1.1). To be sure that we are producing solid conclusions, we first wanted to determine confidence judgment accuracy of the students participated in our study. The current literature offers conflicting results about the accuracy of higher education students in specifying their confidence level. For example, Lang et al. [2015] and Timmers et al. [2013] observed that students are poor estimators of their abilities; while, Vasilyeva et al. [2008] found that students confidence accuracy was fairly well. We hence, state our first research question (RQ-1.1) for assessing students' ability in estimating their confidence in response's correctness.

Moreover, to determine how distinct confidence-outcome category response(s) may affect a student's behavior towards the available feedback in terms of seeking/no-seeking and its related time (i.e. time spent on reading feedback), we constructed two research questions RQ-1.2 and RQ-1.3, respectively. Finally, if feedback-seeking has any positive impact on students' confidence and/or response outcome in the subsequent attempt, RQ-1.4. As mentioned in section 3.2.1.1, the tasks given to the students in this study were "code tracing" problems which require a multiple-step solution and are not so easy for novice learners[8]. It was expected that seeking (corrective) feedback will help students in filling their knowledge gap(s) and answer later questions correctly from the same topic and consequently improve their confidence accuracy. In particular, this study was conducted to answer the following research questions.

- RQ-1.1: To what extent are higher education students able to estimate their confidence judgment in response's correctness?
- RQ-1.2: Does feedback-seeking/no-seeking behavior varies with distinct confidence-outcome categories?
- RQ-1.3: Do students spend different amounts of effort on reading feedback with respect to distinct confidence-outcome categories?

[8]This should not be confused with questions' difficulty levels.

Table 3.1: Number of problems solved with different confidence levels (in rows) and response outcome levels (in columns)

	Correct	**Wrong**	***Total***
High	452	564	*1016*
Low	38	103	*141*
Total	*490*	*667*	*1157*

- RQ-1.4: Does seeking feedback positively affect students' confidence and/or response outcome in the next attempt?

3.3.3.1 Data description

The Dataset1 contains logged traces of student-system interactions along with a timestamp recorded for each activity, during the first experimental study. We treated each "Login-Logout" session as a new case to analyze students' multiple problem-solving traces. Sessions with zero problem submission were ignored as they reflect *exploratory behavior* of the students with the system (e.g. page navigation, check previous solutions and/or scores, etc.). The remaining dataset includes 231 logged sessions of 94 students, who submitted 1,157 solutions in total[9]. Table 3.1 shows the distribution of the number of problems solved with different confidence levels (in rows) and response outcome levels (in columns).

3.3.3.2 Data analyses and results

A Higher Education Student Confidence Judgment in Response's Correctness

In line with the existing observations of Lang et al. [2015], Mory [1994] and Timmers et al. [2013], data in the Table 3.1 show a relative majority of students giving wrong answers with high confidence (HCWR = 49%), in

[9]Note that some students did not submit solutions of all 18 problems.

contrast to other confidence-outcome categories (HCCR = 39%, LCCR = 3%, LCWR = 9%). Also, a big difference in the ratio of responses (both correct and incorrect) given with high and low confidence (i.e. 88% and 12%, respectively) shows that students rated their confidence level as high more often, out of which their judgments were inaccurate in 56% times (see data in the row labeled as 'High'). Without the need for a formal test we, therefore, conclude that higher education students are mostly wrong in their confidence judgments or tend to overestimate their abilities, and this answers our first research question (RQ-1.1).

B Sessions of Variable Lengths

As students were free to solve as many questions as they could in the given time, the number of submitted problems in each session may not be equal. Also, we consider each "Login-Logout" as a new case, some students have multiple "Login-Logout" sessions. Therefore, we have sessions of different lengths based on the count of submitted problems, i.e. 1,2,3,4,5 and 6 for Dataset1 (six being the maximum number of problems that can be submitted in any session).

Table 3.2 shows the percentages of problems solved with different confidence-outcome category for each session length. We highlight a few interesting observations from this data in the following.

First, the percentage of problems solved with HCWR is much higher in all sessions as compared to other category responses; this supports our earlier observation that students overstate their confidence level. Second, the maximum number of correct responses given with high confidence (HCCR) appears to be in sessions with length 6; which shows that students having "mastery" or better knowledge tend to involve in longer problem-solving sessions. Furthermore, responses of LCCR category are visible in sessions of length 4 and above. This observation may be interpreted as students involved in longer sessions *do not* hesitate to admit their lower level of knowledge in some questions. Lastly, sessions of length 1 & 2 contain the highest

Table 3.2: Problems solved per distinct confidence-outcome category in variable lengths sessions

Login-Logout session length	**Confidence-Outcome category**				**Total problems solved**
	LCCR count,(%)	LCWR count,(%)	HCCR count,(%)	HCWR count,(%)	
1	1,(7.1%)	1,(7.1%)	1,(7.1%)	11,(78.6%)	*14*
2	0,(0%)	6,(14.3%)	6,(14.3%)	30,(71.4%)	*42*
3	1,(2%)	9,(17.6%)	13,(25.5%)	28,(54.9%)	*51*
4	4,(14.3%)	2,(7.1%)	4,(14.3%)	18,(64.3%)	*28*
5	4,(8%)	13,(26%)	15,(30%)	18,(36%)	*50*
6	28,(2.9%)	72,(7.4%)	413,(42.5%)	459,(47.2%)	*972*

percentages of HCWRs; which reveals poor behavior of low performing students who may have quit earlier due to less motivational level. The support of these conclusions in terms of observed sessions is however quite low, hence, we believe that findings with a large dataset are required for confirmation.

C Comparison of Feedback Seeking in Variable Lengths Sessions

Before moving towards the next research question, it was necessary to show that feedback-seeking behavior of the students is not affected by sessions of different lengths. Therefore, we decided to compare feedback-seeking frequencies and sessions of different lengths (as determined by the count of problems solved in each Login-Logout session). A moderate positive correlation between the two will validate that it is appropriate to compare sessions of different lengths and sessions conducted during different weeks as the students' behaviors towards feedback remained persistent. A positive correlation is expected because naturally more problems solved will increase feedback-seeking activity; a moderate positive correlation, however, indicates that session length is not a determinant for this increased value.

We applied Spearman's rank correlation coefficient (non-parametric) test with N=205 sessions[10] and the results show a significant positive relation ($r[205] = 0.40,\ p < 0.01$). It is thus appropriate to compare sessions of different lengths and feedback-seeking behaviors for further analyses.

D Feedback Seeking Behavior in Distinct Confidence-Outcome Categories

Comparison of feedback seek vs. no-seek per confidence-outcome category is shown in Fig. 3.5. We can see that feedback-seeking behavior is prominent in case of wrong answers given with high and low confidence (HCWR and LCWR), and feedback no-seeking in case of correct answers (HCCR and LCCR). Percentages of submitted solutions followed by a feedback-seeking activity for each distinct category are as follows: HCCR=14%, HCWR=74.8%, LCCR=18.4%, and LCWR=82.5%. These observations reveal that students sought feedback for some intended purpose and not just arbitrarily.

Next, we used Chi-square independence test and found a significantly positive correlation between confidence-outcome categories and feedback-seeking, $X^2\ (3, N = 1157) = 432.87, p < 0.01$. Based on the results of Chi-square, we reject the null hypothesis; and conclude that feedback-seeking/no-seeking behavior is correlated with confidence-outcome categories, answering RQ-1.2.

However, the Chi-square test did not provide us with a function for predicting feedback-seeking behavior from confidence-outcome. We, therefore, ran a logistic regression using confidence-outcome categories and time taken to solve problems (in seconds), as our independent variables to predict feedback seeking. Our dataset contains 577 feedback-seeking and 580 feedback no-seek observations; so there's no class bias in the data. Both HCWR and LCWR were found to be positively related with feedback-seeking at a significance probability of 0.001 ($p < 0.001$), see results in Fig 3.6. With 75%

[10]Total sessions - sessions with zero feedback seek (231 - 26 = 205)

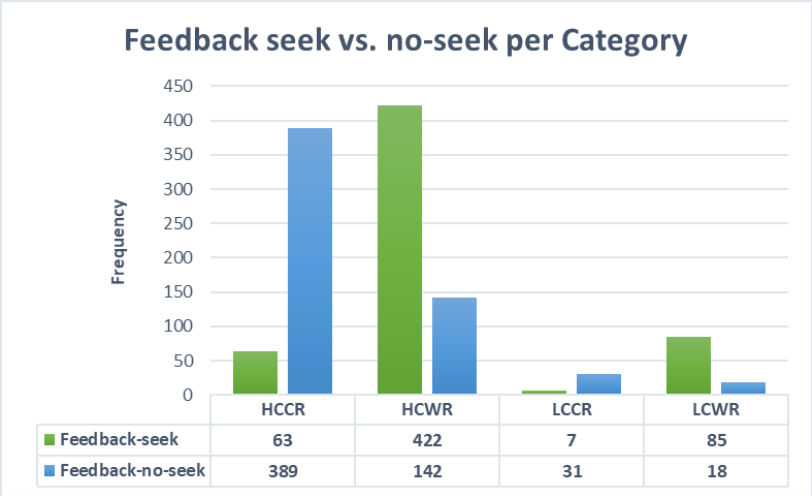


Figure 3.5: Comparison of feedback seek vs. no-seek per confidence-outcome category

train data, we achieved 80.28% prediction accuracy of the derived logistic model and area under the ROC curve is 0.7974.

Based on these outcomes, we agree with Vasilyeva et al. [2008] that students' feedback (seeking) behavior is attributed to the response outcome irrespective of their confidence level. However, unlike Timmers et al. [2013], running a logistic regression we did not find "time taken to solve problems" (or effort) as a significant predictor variable for feedback-seeking (see Fig 3.6).

E Feedback Reading Time in Distinct Confidence-Outcome Categories

Now we will present methods we used to study the impact of different confidence-outcome categories on feedback reading time (i.e. RQ-1.3). Table 3.3 shows descriptive statistics of feedback reading time associated with distinct categories (560 records in total, 17 records were eliminated with no time recorded due to abnormal termination of students' sessions).

Table 3.3: Statistics of feedback reading time per confidence-outcome category

Category	**Count Feedback Seek (Cfs)**	**Total Time Spent (in Sec.)**	**Min.**	**Max.**	**Mean**	**Standard Deviation**
HCCR	61	5074	3	968	83.2	167
HCWR	408	33228	2	1716	81.4	131
LCCR	7	1003	6	532	143.3	230
LCWR	84	5487	5	1012	65.3	126

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8840  -0.5327  -0.5218   0.7698   2.0284

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)     -1.925e+00  1.987e-01  -9.692   <2e-16 ***
CategoryHCWR     2.959e+00  1.965e-01  15.059   <2e-16 ***
CategoryLCCR     6.898e-01  4.928e-01   1.400    0.162
CategoryLCWR     3.417e+00  3.276e-01  10.430   <2e-16 ***
Time taken       9.765e-05  3.421e-04   0.285    0.775
(solve problem)
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3.6: Logistic regression to predict feedback seeking behavior

To visualize data normality, we drew a box plot chart which shows that feedback reading time is not normally distributed within different confidence-outcome categories. Hence, we took feedback reading time in logarithmic scale on the x-axis for better visualization, see Fig. 3.7; as there were huge differences in time spent per category.

The chart shows that median, upper and lowers quartiles of HCWR is greater[11] than that of HCCR and LCWR (we ignored LCCR in our analysis due to insufficient number of instances: Cfs=7). This observation confirms our intuition that students will take more time in filling knowledge gaps when the discrepancy is high between their expected and actual performance. Some prior works [Kulhavy and Stock, 1989, Mory, 1994] also revealed similar results, however, in our dataset, the count of feedback seek with HCWR (Cfs=408, from Table 3.3) is enormous than that of HCCR and LCWR (61 and 84, respectively); thus more evidence is required to support our results.

Next, feedback reading time was regressed on four confidence-outcome categories and time taken to solve problems; no variable showed a significant

[11]As the data is shown in logarithmic scale for better visualization, thus, the slight increase in median of HCWR should not be ignored.

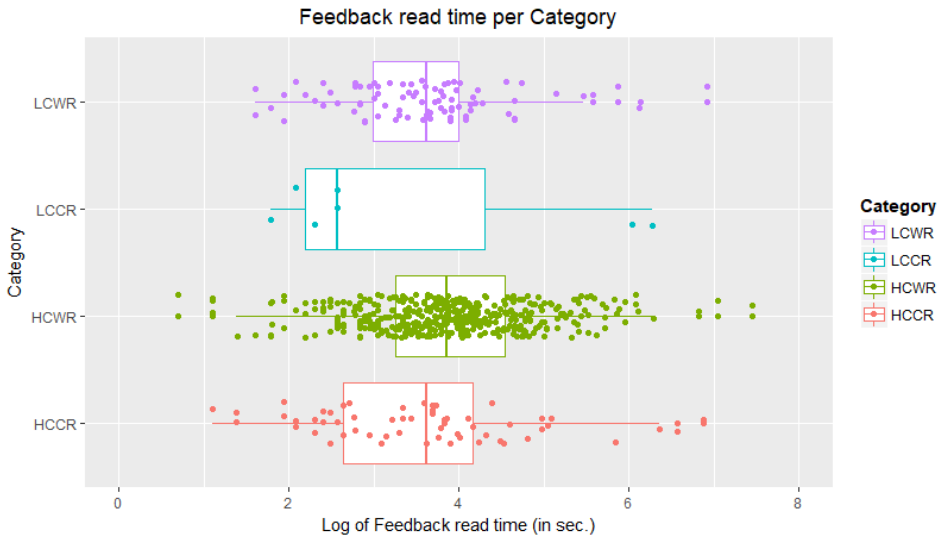


Figure 3.7: Box plot chart: feedback reading time per confidence-outcome category

relation with feedback reading time except for "time taken to solve problems" ($p < 0.05$). Thus, we answer to our third research question (RQ-1.3) as 'no', because we did not find sufficient evidence to claim that students spend different time on reading feedback after distinct confidence-outcome category responses. We, in fact, agree in large with the views presented by Van der Kleij et al. [2012] that feedback reading time is difficult to predict because of its dependence on multiple factors, for example, student's motivation to learn, his/her reading speed, the information presented in the feedback, etc.

F Impact of Feedback Seeking Behavior on Confidence-Outcome Category in Next Attempt

Here, we enlighten on our findings on how feedback-seeking behavior may impact a student's confidence level and response outcome in the subsequent attempt (RQ-1.4). To do this, we called the confidence-outcome category of the last submitted solution as "Original Category", and determined the

impact of feedback-seeking vs. no-seeking on: 1) confidence, 2) response outcome, and, 3) category (a combination of confidence and response outcome); in the next attempt[12]. Tables 3.4, 3.5 and 3.6 contain charts showing comparison of feedback-seeking vs. no-seeking on students' confidence, response outcome and category, respectively; based on the original category. In the followings, we provide a precise description of our observations of the charts shown in all three tables, followed by concluding remarks.

Impact on Confidence Level in Next Attempt. Although a lesser number of students sought feedback with HCCR original category, a slight increase in the confidence levels of students is observed as compared to those with no feedback-seeking, see the charts shown in Table 3.4. Students with HCWR initial category showed a slight decrease in their confidence levels with feedback seek. This decrease in confidence level after seeking feedback may be interpreted as a realization of one's high estimation of his/her abilities (i.e. high confidence in a wrong response). A similar observation is reported by Vasilyeva et al. [2008], although they presented different reasoning. Further, we find an increase in high confidence in case of LCCR initial category after feedback-seeking; while no change in the confidence is observed for students having LCWR initial category. In general, there is a positive impact of feedback-seeking on students' confidence levels in the next attempt; confidence level increases in the case of correct responses and decreases minimally in case of wrong responses.

Impact on Response's Outcomes in Next Attempt. All charts in Table 3.5 show an increase in correct responses in the subsequent attempt for students who sought feedback irrespective of their original category. As

[12]To analyze the impact of feedback on performance attributes in the next attempt, we removed first problem solved per 'Login-Logout' session from the original dataset (N=1,157, total sessions=231), as it has no ancestor variable to observe; this leaves us with 926 records.

mentioned earlier, questions were designed from the same topic for each experimental session and it was expected that seeking feedback will help students in answering later questions correctly. However, this might not be true for all students as only seeking feedback is not enough; it requires a positive attitude and willingness of a student to process the information presented [Timmers et al., 2013].

Therefore, observations of Table 3.4 and Table 3.5 answer our last research question RQ-1.4, that is, seeking feedback positively affected student confidence and response outcome in the next attempt.

Impact on Category in Next Attempt. To visualize the combined effect of a change in students' confidence and response outcome in the next attempt, Table 3.6 contains charts for confidence-outcome categories.

Students with HCCR initial category showed an increase in correct responses given with high confidence (HCCR) and a decrease in HCWR and LCCR responses after seeking feedback. A similar increasing trend is found in HCWR initial category cases, except for a slight increase in responses (correct and incorrect, both) with low confidence. Again, we will interpret this behavior as a positive reflection of one's overestimation about his/her abilities in the previous attempt. Seeking feedback also helped students with LCCR initial category in giving more correct answers with high confidence and lesser wrong answers with either confidence level. While students with LCWR initial category showed increase in correct responses given with high confidence (HCCR) in the next attempt; a ratio of LCCR and LCWR remained constant for answers followed by feedback-seeking and no-seeking activity.

To conclude, feedback-seeking has a positive impact on students' confidence level, response outcome and consequently on the confidence-outcome category in the subsequent attempt. Finally, to test the statistical significance of the relationship between feedback (seek/no-seek) and category in the next attempt, we used Chi-square independence test. The result

shows sufficient evidence to reject the null hypothesis, $X^2\ (3, N = 926) = 27.44,\ p < 0.01$; therefore, we conclude that feedback (seek/no-seek) behavior and the confidence-outcome category in the next attempt are not independent.

3.3.4 Discussions

Confidence-based assessment reveals a discrepancy between a student's confidence about an answer in contrast to the actual outcome. Feedback can play a central role in filling this *knowledge gap* provided that it contains the correct solution and allows students to make a comparison with their submitted solution [Vasilyeva et al., 2008] (i.e., through textual explanation or by highlighting errors; we adopted the latter approach). We called this combination of 'knowledge of correct response' and 'elaborated feedback' as "corrective feedback". Students' positive attitudes towards different feedback types are reported by several researchers from varying perspectives, including their confidence (or certitude) level [Kulhavy and Stock, 1989, Mory, 1994, Timmers et al., 2013, Van der Kleij et al., 2012, 2015, Vasilyeva et al., 2008]. However, the confidence level considered in some of these studies is not related to each individual answer submitted by a student (e.g., [Timmers et al., 2013, Van der Kleij et al., 2012, 2015]). Also, a detailed analysis of students' behaviors towards feedback in confidence-based assessment and its potential effect(s) were missing in the literature.

We conducted three experimental sessions with higher education students using a computer-based assessment system. This exploratory work analyzed logged data from different aspects which provide useful insights for our future work (that is, research objective II) and supporting adaptation in a confidence-based assessment system, as discussed below.

First, our result using the Dataset1 (given in Table 3.1) shows that higher education students do not specify their confidence level accurately as also identified by Lang et al. [2015], Mory [1994] and Timmers et al. [2013]. One

Table 3.4: Impact of feedback seek vs. no-seek on confidence level in next attempt

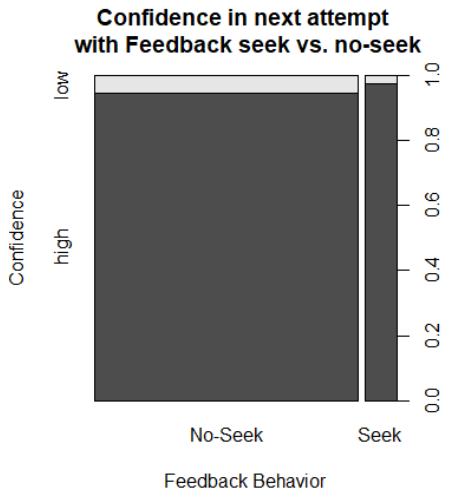
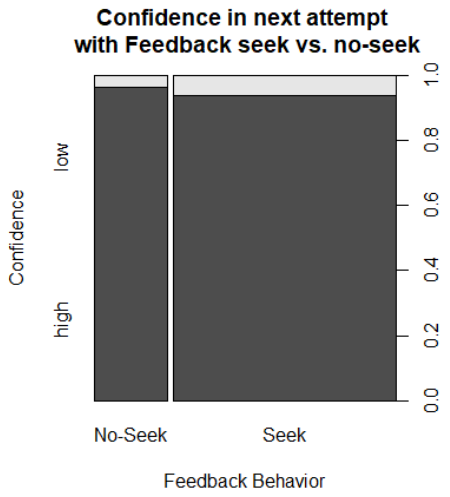
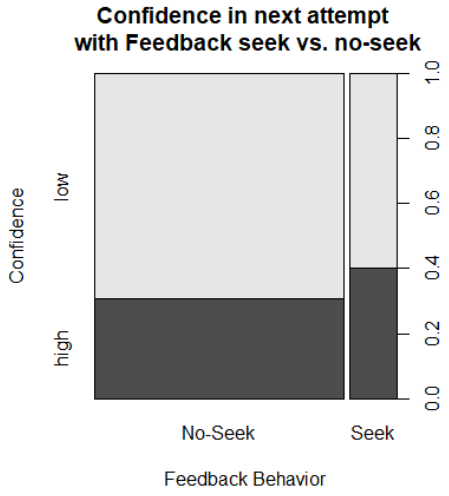
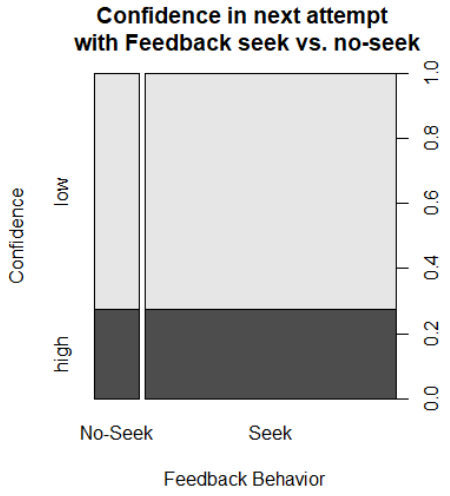
Original Category: HCCR (No-Seek:89%; Seek:11%)	**Original Category: HCWR** (No-Seek:25%; Seek:75%)
	
Original Category: LCCR (No-Seek:84%; Seek:16%)	**Original Category: LCWR** (No-Seek:15%; Seek:85%)
	

Table 3.5: Impact of feedback seek vs. no-seek on response outcome in next attempt

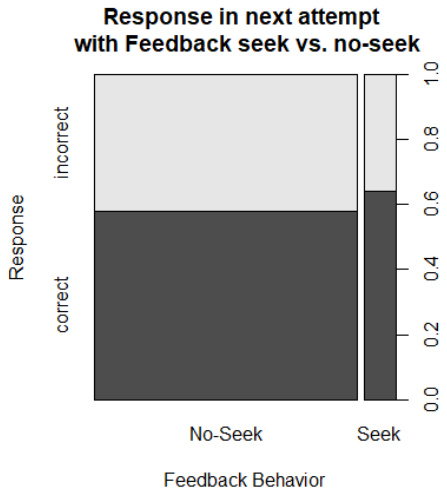
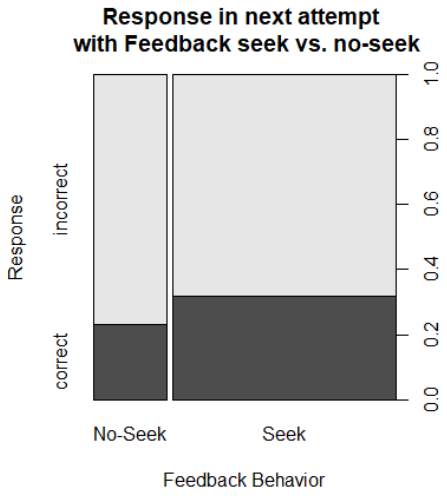
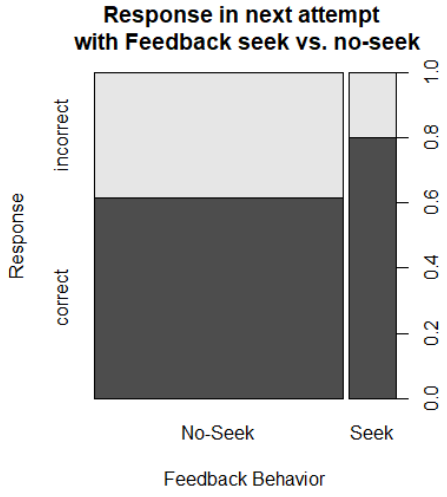
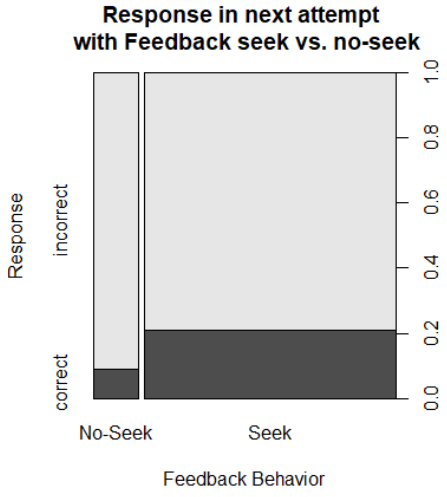
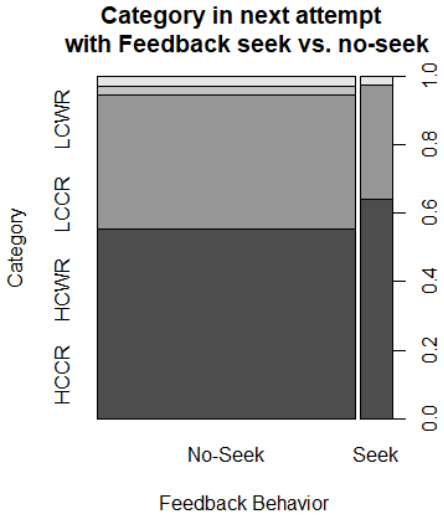
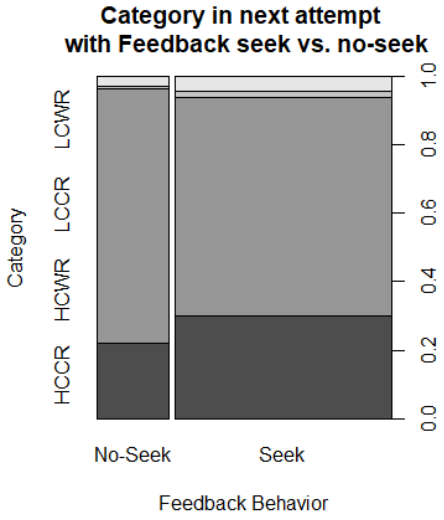
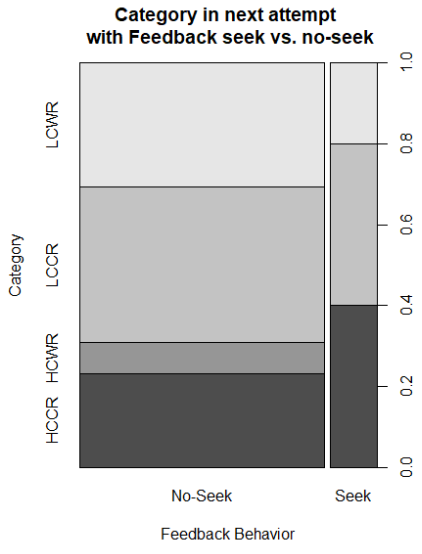
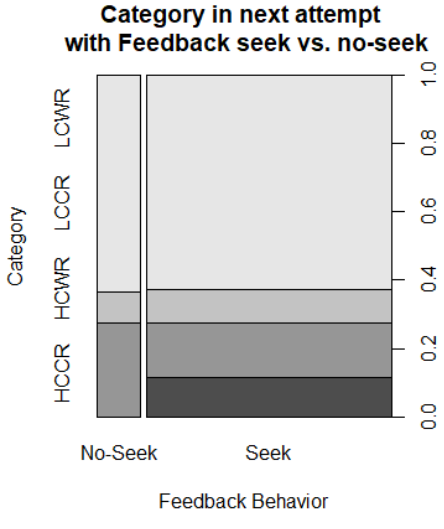
Original Category: HCCR (No-Seek:89%; Seek:11%)	**Original Category: HCWR** (No-Seek:25%; Seek:75%)
	
Original Category: LCCR (No-Seek:84%; Seek:16%)	**Original Category: LCWR** (No-Seek:15%; Seek:85%)
	

Table 3.6: Impact of feedback seek vs. no-seek on confidence-outcome category in next attempt

Original Category: HCCR (No-Seek:89%; Seek:11%)	**Original Category: HCWR** (No-Seek:25%; Seek:75%)
	
Original Category: LCCR (No-Seek:84%; Seek:16%)	**Original Category: LCWR** (No-Seek:15%; Seek:85%)
	

approach to minimize this inaccuracy is by using some marking scheme, as done in [Vasilyeva et al., 2008], that assigns positive and negative scores based on distinct confidence-outcome category responses. However, this can impose internal pressure on the students to avoid penalization which may affect their performance as well. Hence, we propose to construct a prediction model which estimates the confidence accuracy of each student. And, utilize this information to generate personalized "feedback about self-regulation" (FR) that guides a learner on how to direct and regulate their actions towards learning goals [Hattie and Timperley, 2007]. In this case, FR feedback can help over- and under-confident students to improve their confidence accuracy and knowledge to achieve *mastery* in the subject domain.

Second, in Section 3.3.3.2 – paragraph C, we compared feedback-seeking behavior of the students in sessions of variable lengths to analyze if this behavior is affected by the number of problems solved in a 'Login-Logout' session. This was also important as we conducted three experimental sessions in different weeks and we wanted to see if it is appropriate to compare them. Our results show a moderate positive relation between session lengths and feedback-seeking behavior which confirms that students' behaviors towards feedback remained persistent.

Third, we find strong evidence that students' behaviors towards feedback vary with distinct confidence-outcome categories (see Section 3.3.3.2 – paragraph D). More specifically, as the intuition suggests, we find that students' feedback-seeking/no-seeking behavior is associated with their response's outcome (i.e. students read corrective feedback in case of wrong responses). Another important factor which has been associated with feedback-seeking behavior is 'effort' (or time spent) in solving a problem [Timmers et al., 2013]. However, we did not find any significant relationship between the two in our analysis (see the results of logistic regression in Fig. 3.6). Although, if proved, it could have been used to argument why all high discrepancy instances (i.e. wrong response given with high confidence) were not followed

by a feedback-seeking activity[13]. We can assume that students who spent less time answering a question, would rarely be interested in knowing correct response and/or their mistakes.

Next, we compared the students feedback reading time with respect to distinct confidence-outcome categories as described in Section 3.3.3.2 – paragraph E. We observed that students took more time in reading feedback in case of wrong responses given with high confidence which is in line with prior results of Kulhavy and Stock [1989] and Mory [1994]. However, we failed to find a significant difference in feedback reading times; hence, further investigation is required to study these time-specific behaviors in confidence-based assessment.

Another distinctive contribution of the research study I is that we determined the impact of seeking feedback on student's confidence level and response outcome in a subsequent attempt which is detailed in Section 3.3.3.2 – paragraph F. We find a significantly positive effect of feedback-seeking versus no-seeking on the confidence-outcome category. We remind that questions were designed from the same topic for each experimental session in this study. Therefore, it was expected that seeking corrective feedback will help students in answering a later question(s) correctly. However, our results show that feedback-seeking also affected students' confidence level positively. For example, in case of "*low confidence-correct response*" approximately 10% of the students who sought feedback changed their confidence level as "high" in next question (see Table 3.4). We can assume hypothetically that seeking corrective feedback in case of a correct response helped a student in doubt (or low confident) to gain confidence about his/her knowledge. Positive change in students confidence level as an effect of seeking feedback was also observed by Vasilyeva et al. [2008], but we provide detailed results which are proven statistically. To conclude, investigating students' behaviors towards

[13]Leave aside students' personal characteristics for a moment; which may affect their feedback reading time: motivation, reading speed, etc., as discussed in [Timmers et al., 2013].

feedback offer valuable information in case of confidence-based assessment as compared to the traditional one-dimensional assessment approach.

Overall, we achieved very promising results which support our initial thoughts that capturing students' behaviors towards feedback in the confidence-based assessment will serve as a useful parameter in determining their engagement/disengagement behaviors [Maqsood and Ceravolo, 2018]. In this regard, our next objective aims at defining various engagement/disengagement behaviors using: student's response outcome, confidence level and followed feedback-seeking activity. Seeking and utilization of available feedback is much dependent on students' engagement level [Mory, 2004, Timmers et al., 2013], which may vary within and across different sessions.

3.4 Objective II: Define a Scheme to Classify students' activities into engagement/disengagement behaviors

The results and findings of research study I forms the baseline for the second objective of this work. More specifically, we have identified the following student performance parameters for constructing a student engagement model: 'response correctness', 'confidence level', 'feedback-seeking/no-seeking behavior'. We did not find 'time spent' (or effort invested) on a problem as a predictive factor of problems solved with distinct confidence-outcome categories (see Section 3.3.3.2 – paragraph D); thus, we ignored this parameter. In the followings, we provide details of our proposed method to reify students' engagement/disengagement behaviors during confidence-based assessment.

3.4.1 Activities classification into positive and negative student engagement behaviors

Distinct confidence-outcome categories are defined in terms of varied knowledge regions by Hunt [2003], namely: HCCR shows mastery of a student in the subject domain; LCCR depicts doubt or hesitation about one's knowledge; HCWR means that the student has misconceptions, and LCWR shows unknowing knowledge state of a student. In this respect, seeking or no-seeking corrective feedback followed by a specific category response can direct us to different engaged and disengaged behaviors of the students during assessment.

As intuition suggests, our results (of research study I) indicate that students' feedback-seeking behavior is predicted by the wrong response given with either confidence level [Maqsood and Ceravolo, 2019], therefore, we do not differentiate in feedback-seeking or no-seeking behavior in case of a correct response. However, we define different engagement and disengagement behavior classes in case of a wrong response based on student's associated confidence level and feedback-seeking behavior. To make these categories more logical, we gave them meaningful labels, see Table 3.7, which are precisely explained in the following text.

Seeking corrective feedback in case of correct solution is infrequent as the student already knows questioned content, however, it could be useful for responses given with low confidence (i.e. LCCR) wherein the respective student has doubts about his/her knowledge which is in fact correct. But, our data shows that students rarely paid attention to corrective feedback after giving a correct response; precisely, students with HCCR and LCCR requested corrective feedback for only 11% and 16% times, respectively. Therefore, we only differentiated between correct responses given with high and low confidence as high knowledge (HK) and less knowledge (LK), respectively.

On the other hand, different reactions to corrective feedback in case

Table 3.7: Mapping of student problem-solving activities into (dis)engagement behaviors

Confidence-Outcome Category	**Student Response to Corrective Feedback**	**New label for (Dis)Engagement Behavior**
HCCR (*mastery*)	Feedback Seek (FS) or Feedback No-Seek[a]	High Knowledge (HK)
LCCR (*doubt*)		Less Knowledge (LK)
HCWR (*misinformation*)	Feedback Seek (FS)	Fill-knowledge Gap (FG)
	Feedback No-Seek	Knowledge Gap (KG)
LCWR (*unknowing*)	Feedback Seek (FS)	Learn (LE)
	Feedback No-Seek	Not Interested (NI)

[a]No label is stored for this activity in the traced log, so it is considered by absence of FS activity after each submitted problem.

of wrong responses lead to distinct engagement/disengagement states. For example, seeking corrective feedback in case of highly confident wrong response (HCWR) means that a student is trying to fill the knowledge gap that occurred as a misconception or discrepancy between his/her expected and actual knowledge; thus, we name it as Fill-knowledge Gap (FG). While not seeking feedback in case of HCWR means that the student did not attempt to repair the gap(s), so we label it as Knowledge Gap (KG). Low confidence wrong response (LCWR) reflects the *unknowing* knowledge state of a student, and therefore, seeking feedback, in this case, means that a student is trying to learn (LE). Also, in the assessment model we followed for data collection (as given in [Maqsood and Ceravolo, 2018]), a student can only view the correct solution (and elaborated feedback) after he/she submits a problem [14]. This could also be a reason for students to submit a low confident response(s); still, it reflects the unknowing state of the respective student. However, we assume that a student is not interested (NI) in the assessment process if he/she ignores corrective feedback following a low confident wrong response.

3.4.2 Research study II

In the second research study, we aimed to introduce a novel approach to determine students' engagement/disengagement by analyzing their navigation traces generated during interaction with a computer-based assessment tool, which we just explained in the previous section. Our approach is more generic as it is not based on human expert's defined time limits which are computed from students' collected data who participate in that specific study, as it is done in Beal et al. [2006], Joseph [2005], and, Brown and Howard [2014]. For example, after analyzing students data, Beal et al. Beal et al. [2006] identified 10 seconds *activity-time* limit as a boundary condi-

[14]This allows a student to re-attempt each problem any number of times before making a final submission.

tion to classify students problem-solving activities into various engagement behaviors.

Then, our next objective was to evaluate the usefulness of the proposed method. For that, we applied the scheme on a real dataset, which we previously called 'Dataset1' and performed different data analyses.

3.4.2.1 Data description

Since, in this study, we needed to validate our proposed scheme; we retained the traces of the students only whose real identities (i.e. unique name and/or class registration id) could be retrieved from their accounts information in the computer-based assessment tool. A many-to-one mapping was defined between profiles/accounts obtained from the logged data and the real students who participated in the first experimental study. The data of 21 Login-Logout sessions (out of 231 total logged sessions) were discarded since it was not possible to identify real identities of the respective students. Hence, the original data (or Dataset1) of 94 students was reduced to 91 students, with 210 Login-Logout sessions containing 1,046 submitted answers. From each profile, we extracted students' Login-Logout traces containing relevant problem-solving activities as discussed below (activities labeled with 'page navigation' and 'check previous solution' were removed from the traces, see 'activity label' in Fig 3.2).

A Traces transformation

Each Login-Logout session contains the confidence-outcome category[15] for each submitted problem followed by a (corrective) feedback-seeking (FS) activity if it was requested for that specific problem[16]. Here is a sample

[15]The activities labeled (in raw data, Fig 3.2) with: 'submit high confidence' and 'submit low confidence' in combination with the confidence levels (i.e. high or low) were changed to respective categories, as mentioned in Section 3.3.

[16]The activities labeled (in raw data, Fig 3.2) with 'check solution' were changed to label FS.

Table 3.8: Students' sample traces, activities are separated by a hyphen '-'

Trace1:	HK-HK-LK-HK-LK
Trace2:	HK-HK-FG-HK-FG-HK
Trace3:	KG-KG-LE-LE-NI-KG
Trace4:	HK-HK-HK-FG-FG-FG-HK
Trace5:	FG-HK-HK-LK-LK

trace with activities separated by a hyphen '-':

HCCR-FS-HCWR-FS-HCWR-HCCR

In this trace, a student has submitted two HCCR and two HCWR problems and performed feedback-seeking (FS) activity for the first two submitted problems only (which are HCCR and HCWR, respectively). We transformed each trace into its equivalent (dis)engagement behaviors as defined by activities classification in Table 3.7. So, the above sample trace changes to:

HK-FG-KG-HK

Likewise, all traces were transformed into respective engagement/disengagement behavioral patterns. Table 3.8 contains some sample traces, each trace containing a sequence of different engagement/disengagement behavioral patterns representing students' problem-solving behaviors. And, Table 3.9 shows the frequency distribution of all the behavioral patterns in Dataset1.

As students were free to solve any number of problems in the given time, the collected data contained sessions of variable lengths. We assume that variable length sessions can be considered equivalent (or matching) if a similar group of activity patterns is found in two or more traces. Therefore, to identify similar problem-solving behavioral groups, we computed proportion value for each behavioral pattern p_i per Login-Logout session using Eq. (3.1):

Table 3.9: Frequency distribution of (dis)engagement behavioral patterns in Dataset1

(Dis)Engagement behavioral pattern	HK	LK	FG	KG	LE	NI
Frequency	422	36	370	120	82	16

$$\frac{\sum_i p_i}{\sum_{N \in P} \sum_{j=N} p_j} \tag{3.1}$$

Where $i, j \in P$ and $P = \{\text{HK, LK, FG, KG, LE, NI}\}$. Using this expression, proportion count for each pattern in the sample trace <HK–FG–KG–HK> is:

HK=0.5; LK=0; FG=0.25; KG=0.25; LE=0; NI=0

All traces in the whole dataset were converted to corresponding patterns' proportion count in a similar manner.

3.4.2.2 Problem-solving sessions clustering

Next, to determine groups of (Login-Logout) sessions showing similar (dis)engagement behaviors of students, we decided to perform K-means clustering. To obtain the optimal number of 'k' we used NbClust method of R which uses 30 different indices (for example Cindex, CH index, Beale index, DB index, Silhouette index, Dunn index, etc.) and returns the best value by maximal voting [Charrad et al., 2012]. We get k=4 from NbClust method, and to verify it, we plot this value on the elbow method, using a (red) dashed-line for k=4, as shown in Fig. 3.8. As we can see, both methods suggest that k=4 is a suitable value for the number of clusters for Dataset1. Then, we run K-means algorithm using k=4 for 15 iterations with 25 initial random points, to obtain stable clusters. Table 3.10 contains variables means of four resultant clusters.

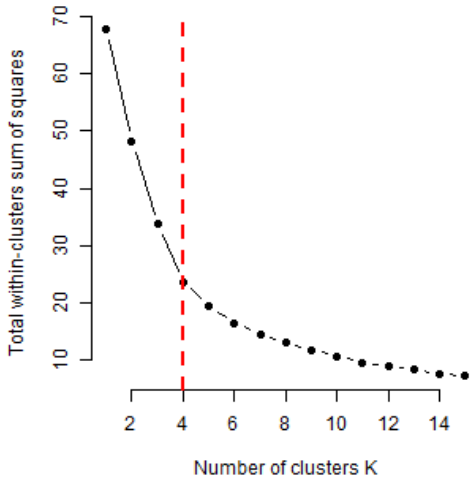


Figure 3.8: Optimal k=4 (computed using NbClust method of R), plotted on elbow method

Table 3.10: Variables means within each cluster (important variables for each cluster are highlighted in gray color)

Variables	**Cluster 1 (N=27)**	**Cluster 2 (N=71)**	**Cluster 3 (N=83)**	**Cluster 4 (N=29)**
HK	0.07	0.17	0.7	0.14
LK	0.17	0.01	0.02	0
FG	0.1	0.75	0.23	0.16
KG	0.03	0.05	0.02	0.67
LE	0.55	0.01	0.02	0.01
NI	0.08	0.01	0.01	0.02

Cluster 1 (N=27) largely contains sessions of low confident students with learning as a dominant activity being performed (LE=55%), followed by 17% correct responses given with low confidence (i.e. less knowledge – LK). Sessions in this cluster represent positively engaged behaviors as students having little or no knowledge seek corrective feedback to learn the material. Cluster 2 (N=71) also represents positively engaged students having high confidence with 75% fill-knowledge gap (FG) and 17% high knowledge (HK) activities. Students in this group were overconfident of their knowledge and they showed concern about mistakes made or inaccurate knowledge.

Cluster 3, which is the largest one (N=83), contains sessions with 70% high knowledge (HK) activities. Further, students in this cluster showed great interest in fill-knowledge gap (FG) activity when their responses were incorrect (i.e. 23%). Therefore, Login-Logout sessions in this group reveal highly positive engagement of students during the assessment. Cluster 4 contains sessions with higher ratio of knowledge gap (KG) activities (i.e. 67%) which show students' disengagement during the assessment. In other words, students were overconfident of their knowledge and they did not show a keen interest in filling their knowledge gap(s) either, FG=16%; and gave very few correct responses: 14% only.

To summarize, we obtained three groups of positively engaged problem-solving behaviors while one group showing student disengagement during the assessment (i.e., Cluster 4). Even the two most related positive engagement groups, that is Cluster 1 and 2, in which students mostly gave wrong responses and seek feedback for learning; reveal different needs of the students due to low and high confidence level, respectively. Thus, monitoring students' feedback-seeking behavior to capture student (dis)engagement during confidence-based assessment discloses useful insights about their problem-solving behaviors and needs. Moreover, clustering Login-Logout sessions of students in confidence-based assessment resulted in varying groups having dominance[17] of a specific behavior in each cluster which shows that students

[17]That is, each cluster is dominated by a specific engagement behavior with more than

largely depicted recurrent engagement behaviors in respective Login-Logout sessions.

3.4.2.3 Data analyses and results

Now that we have developed a scheme for classifying students' problem-solving activities into six different engagement and disengagement behaviors. With the application of the proposed scheme on a real dataset that resulted in four distinct behavioral groups (using K-mean clustering algorithm). One natural question that arises here is that whether these behavioral groups also differ in quantitative student performance or just represent engagement/disengagement behaviors as defined by us – (RQ-2.1)? Clearly, there is a huge performance difference in problem-solving sessions of Cluster 3 and others in the remaining clusters. But, it is difficult to say precisely for Cluster 1 and Cluster 2 sessions. Another question is, how these distinct engagement groups relate to the students' actual performance in the course – (RQ-2.2)? In the following text, qualitative analyses that we performed to answer these questions are presented.

A (RQ-2.1): Does these engagement groups also differ quantitatively in student performance scores?

To compute the accumulative performance score for each session, we assigned positive and negative scores to student responses based on their respective confidence-outcome category. For this, we chose one of the simplest confidence-based marking scheme (CBM) used in the literature [Vasilyeva et al., 2008], that assigns different points to distinct confidence-outcome category responses in the following manner: HCCR = +2; LCCR = +1; HCWR = -1; and LCWR = 0. This scoring scheme is based on the core idea of usual CBM schemes, that is, to reward more to correct responses given with high confidence and give less credit to low confident correct responses;

50% of its presence in the entire traces.

while highly confident wrong responses are penalized, and zero points are given to wrong but low confidence responses. Next, to compare quantitative performance scores of sessions relating to different clusters, we draw boxplot chart for each cluster, see Fig. 3.9; number shown inside the box represents 'median' performance score of that particular cluster.

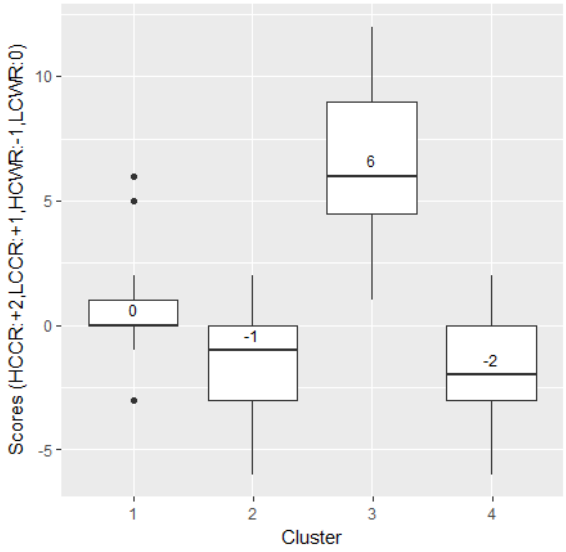


Figure 3.9: Student performance scores per cluster (with median score of the cluster shown in each box)

We can see that as expected Cluster 3 sessions achieved the highest performance scores as compared to other clusters. Interestingly, Cluster 1 which represents low confident responses, show a positive score higher than that of Cluster 2 and Cluster 4 sessions which incurred high penalization for categories relating to highly confident wrong responses (e.g. FG and KG). Problem-solving activities of Cluster 1 support our initial thought that students with less knowledge or who have doubts about it would prefer to submit a low confident response(s) to get the correct solution for their learning purposes and thus yielding engagement behavior during the assessment process.

Clusters 2 and Cluster 4 both contain a majority of high confident wrong responses but with opposing engagement behaviors, that is, fill-knowledge gap (FG) and knowledge gap (KG), respectively. The boxplot charts of both show that Cluster 2 sessions achieved slightly higher performance scores than those of Cluster 4 sessions. We can, thus, conclude that Cluster 2 sessions not only represent positively engaged problem-solving behaviors, but students achieved better scores as compared to negatively engaged or disengaged behavioral group (i.e. Cluster 4). However, in contrast to our expectations, the difference in performance scores of both clusters is not significant (i.e. only 1 absolute number) and one possible reason could be that many students might not have paid attention to the available corrective feedback and rather just opened it for curiosity. A detailed analysis of students' feedback reading time with respect to distinct confidence-outcome category responses is presented in [Maqsood and Ceravolo, 2019] and we found that students with high confidence wrong responses spent the minimal time of 2 seconds on reading feedback.

B (RQ-2.2): How these engagement groups relate to the students' actual performance in the course?

To analyze the relation between these student engagement groups and their performance in real class, we selected 20% high and low performance students (i.e. 18 out of 91) based on their standardized final scores[18] computed at the end of the course. Then, we extracted respective engagement groups of these students from the clustering we performed earlier, see Table 3.11.

As we can see, high-performance students' sessions majorly lie in Cluster 2 (47%) and Cluster 3 (45%) which were dominated by 'fill-knowledge gap' (FG) and 'high knowledge' (HK) problem-solving activities, respectively. We can conclude that high-performance students largely have high confidence in their knowledge and showed positive attitudes during the as-

[18]Standardized or z-score is computed as: (student's final score — class mean) divided by class standard deviation (https://en.wikipedia.org/wiki/Standard_score).

Table 3.11: Engagement groups of high and low performance students

Class Performance Groups	**Student Engagement Groups**				***Total Sessions***
	Cluster 1 count(%)	Cluster 2 count(%)	Cluster 3 count(%)	Cluster 4 count(%)	
High performance students	2(4%)	22(47%)	21(45%)	2(4%)	*47*
Low performance students	9(28%)	10(31%)	9(28%)	4(13%)	*32*

sessment process. Whereas, low-performance students depict varying behaviors of confidence levels and engagement during the assessment. Specifically, their problem-solving behaviors prominently relate to 'learning' (LE-Cluster 1, 28%), 'fill-knowledge gap' (FG-Cluster 2, 31%) and 'high knowledge' (HK-Cluster 3, 28%) groups. Also, the disengagement behavioral group representing 'knowledge gap' (KG) activities is relatively more visible in low-performance students (Cluster 4, 13%) wherein students did not show any interest in knowing the actual answer and correct their mistakes.

To conclude, the comparison between engagement behaviors of high and low-performance students show a difference in their problem-solving activities during the assessment. We can thus claim that high and low-performance students not only differ in their confidence-outcome category responses but also depict different feedback-seeking behaviors which relate to varied engagement behaviors in confidence-based assessment.

3.4.3 Discussions

Given that feedback-seeking/no-seeking behavior of students in confidence-based assessment offers more valuable information in contrast to traditional one-dimensional assessment [Maqsood and Ceravolo, 2019], we proposed to utilize this information to classify students' problem-solving and feedback-seeking/no-seeking activities into positive and negative engagement behaviors. Specifically, we defined six distinct (dis)engagement behaviors based on the following three attributes: student response's correctness, the associated confidence level for each submitted solution, and a followed corrective feedback-seeking activity (if it was requested for that specific problem during assessment), see Table 3.7. In fact, it is the very first attempt to investigate students varying behaviors during confidence-based assessment.

Then, as described in Section 3.4.2.2 clustering students' traces based on these problem-solving behaviors resulted into three groups of student engagement and one group of disengagement. These distinct groups show

some dominant problem-solving behaviors depicted by the students during assessment (see highlighted cells in Table 3.10), that reveal their active involvement in the ongoing assessment process.

As pointed out by Baker and Rossi [2013], validating models of student engagement is more challenging than validating their knowledge models. We determine the usefulness of our proposed scheme by comparing the resultant behavioral groups with students' actual performance. More precisely, our analysis shows that these groups of engaged/disengaged behaviors also differ in quantitative student performance scores in confidence-based assessment (details are given in Section 3.4.2.3, paragraph A). Although, a significant difference in performance scores is not observed between Cluster 2 and Cluster 4 revealing opposite student engagement behaviors (i.e. engaged vs. disengaged, respectively). Our previous results indicate that feedback-seeking has a positive impact on students' confidence and performance [Maqsood and Ceravolo, 2019], hence, students depicting engaged behaviors like 'learning' (or LE) and 'fill-knowledge gap' (or FG) are expected to show better performance outcomes than respective disengaged behaviors, 'not interested' (or NI) and 'knowledge gap' (or KG).

Additionally, our results presented in Section 3.4.2.3, paragraph B also showed that high and low-performance students relate differently to these engagement groups. These results show the usefulness of our approach for capturing students' engagement using new parameters; which was previously determined usually by response correctness and activity-time (taken to solve a problem) [Beal et al., 2006, Joseph, 2005]. Our approach of determining student engagement is not based on human expert's defined rules to classify different activities as 'engaged' or 'disengaged', which are usually domain-specific and dependent on the dataset in investigation; but it is rather based on theoretical reasoning (given in Section 3.4.1).

But, student engagement is not a stable factor and is subject to change over time [Joseph, 2005], even during a single session. And, this requires

the construction of a suitable mechanism to represent students varying behaviors. We addressed this challenge in the next research objective of our work.

3.5 Objective III: Model, analyze and predict students varying engagement and disengagement behaviors using probabilistic models

Computer-based assessment systems enable tracking students' activities at micro-level, i.e. event by event; but this information can be exploited only if events are encoded with a suitable representation model. As our next steps, we intended to model and analyze sequential traces reflecting students engagement/disengagement behaviors using probabilistic models to understand the evolution of their behaviors from one state to another, within and across different Login-Logout sessions. And, to predict students future behavior for twofold purposes: a) to test the validity of our approach, and, b) to support identification of *unproductive* behaviors beforehand so that respective student(s) can be offered personalized assistance (which is one of the future works of this thesis, see next chapter).

Different probabilistic models (e.g. Markov chain, hidden Markov model, mixture Markov model, and other variants) are found to be considerably useful in studies conducted for analyzing human behaviors, for exmaple, [Beal et al., 2007, Cadez et al., 2003, Fok et al., 2005, Khalil et al., 2007, Park et al., 2018, Taraghi et al., 2015]. The underlying idea is to exploit trails of sequential activities performed by the users while interacting with a computer-based system. These sequential activities can be easily modeled and visualized by a suitable probabilistic model to interpret users' intended behaviors in a realistic manner. Each unique action or activity performed by the users is represented as a state of the model. And, transition probabilities between different states show a change in a person's current activity to another ac-

tivity. In Chapter 2, we have explained basic notations and definitions of Markov chain which is one of the simplest methods and forms the basis for other specialized models.

Earlier in this chapter (Section 3.4.1), we present a scheme to categorize students problem-solving activities into different engagement and disengagement behaviors. In particular, six behavioral patterns are defined based on theoretical reasoning, namely: high knowledge (HK), less knowledge (LK), fill-knowledge gap (FG), knowledge gap (KG), learn (LE), and, not interested (NI). We refer to these categories as behavioral patterns as they do not represent sole actions a student performs, but instead, each discrete label is a composite of three attributes of a student's problem-solving behavior and thus reflecting his/her (dis)engagement behavior.

Now, our objective was to construct a mechanism to model these engagement/disengagement behaviors which can be used to analyze students' sequential problem-solving traces, wherein each activity is represented by a behavioral pattern belonging to the set P, where $P = \{HK,\ LK,\ FG,\ KG,\ LE,\ NI\}$. The results of research study II revealed the existence of different behavioral groups in students' problem-solving traces which were found to be associated with the students' real performance, see Section 3.4.2.3. However, the K-means algorithm is restricted to be applicable to categorical data [Huang, 1998] and neither we considered temporal ordering between different activities. Hence, in research study III we decided to employ a more sophisticated clustering method for multivariate categorical data known as "model-based clustering".

Before presenting our methodology for research study III, in the following sub-section, we show the outcomes of a preliminary experiment that we performed to compare the obtained behavioral groups at student versus trace level.

3.5.1 Student versus trace level behavioral groups

In research study II (Section 3.4.2.2), we grouped students' problem-solving behaviors at trace level which is the lowest representation wherein each trace contains all the activities performed in single Login-Logout session. Moreover, the many-to-one mapping defined between traces and the students' real identities in Section 3.4.2.1 show that multiple traces belong to individual students.

The identification of behavioral groups through clustering performed at trace level did not take into account students' identities. This means that multiple traces belonging to a student were might be represented by different (obtained) clusters. Our intuition behind this approach was to capture the potential *drift* in students' problem-solving behaviors that may occur from one Login-Logout session to another, similar to as it was done in a recent work by Hansen et al. [2017].

However, a more common approach in educational data mining is to group students sharing similar characteristics (i.e. problem-solving behaviors to be more precise in the current context) or performing data analyses at the student level. To get convincing proof of the advantage of using the prior approach, we performed an experiment to compare student versus trace level behavioral profiles using Dataset1, containing 197 Login-Logout sessions. Each session/trace contains minimum 2 and maximum 6 solved problems; traces with only 1 solved problem were removed as nothing can be inferred or predicted from a single activity.

Table 3.12 shows sample data at student level where each row represents a student's record containing the counts of problems solved with each behavioral pattern from set P. And, Table 3.13 contains sample data at trace level; each row represents a single trace (or a Login-Logout session) with the proportion of problems solved (since traces were of different lengths) with each behavioral pattern from set P.

By following the same methodology of clustering problem-solving ses-

Table 3.12: Sample data at student level (showing frequency of activities solved with different behavioral patterns by each student)

Student	**HK**	**LK**	**FG**	**KG**	**LE**	**NI**
St1:	10	0	1	1	0	0
St2:	4	0	5	0	0	0
St3:	2	0	0	2	0	0
St4:	3	1	2	0	0	0
St5:	5	0	1	1	0	0
St6:	1	0	3	4	0	0
St7:	3	0	2	0	9	0
St8:	0	2	1	0	7	4
St9:	1	7	0	1	0	2
St10:	0	0	3	1	2	6

Table 3.13: Sample data at trace level (showing proportion of problems solved with different behavioral patterns in individual Login-Logout sessions)

Traces	**HK**	**LK**	**FG**	**KG**	**LE**	**NI**
T1:	0.00	0.00	1.00	0.00	0.00	0.00
T2:	0.50	0.00	0.00	0.50	0.00	0.00
T3:	0.50	0.17	0.33	0.00	0.00	0.00
T4:	0.75	0.00	0.25	0.00	0.00	0.00
T5:	0.00	0.20	0.60	0.00	0.20	0.00
T6:	0.10	0.40	0.10	0.00	0.40	0.00
T7:	0.00	0.25	0.00	0.00	0.25	0.50
T8:	0.50	0.00	0.00	0.50	0.00	0.00
T9:	0.60	0.10	0.25	0.15	0.00	0.00
T10:	0.00	0.33	0.00	0.00	0.33	0.33

Table 3.14: Comparison of the students' next activity prediction accuracy of clusters obtained at student and trace level (accuracy computed with 5 folds cross-validation, 5 iterations)

Cluster	Student level	Trace level
1	67.04% (20 students; 50 traces)	58.78% (25 traces)
2	53.41% (13 students; 28 traces)	66.54% (64 traces)
3	49.95% (35 students; 49 traces)	67.74% (82 traces)
4	59.30% (18 students; 52 traces)	66.46% (26 traces)
5	51.53% (6 students; 18 traces)	–

sions (as described in Sec 3.4.2.2), the optimal number of clusters (K) returned by the NbClust method were 5 and 4 respectively for student and trace level data. Then, K-means clustering was run on both data for 15 iterations with 25 initial random points to get stable clusters. After obtaining the clusters, a first-order Markov chain was constructed for all the 9 clusters. Table 3.14 shows the comparison of student's next activity prediction accuracy computed by 5-folds cross-validation, averaged over 5 iterations; for student and trace level clusters.

Form a quick comparison of the prediction accuracy values given in Table 3.14, we can see that the prediction accuracy of clusters at trace level is better than what we got for student-level clusters. This supports our initial idea that it is more appropriate to analyze individual's problem-solving behaviors at trace level, which may reveal a drift in a student's behavior over time.

3.5.2 Research study III

Markov chain is primarily an efficient method to model sequential data and make predictions. However, student engagement is not a stable factor and

is subject to change over time [Joseph, 2005]; therefore, striving for a single 'best' model to represent students' behaviors is not adequate. In the previous section, we obtained distinct behavioral groups from students' problem-solving activities using the simplest K-means clustering algorithm which did not consider temporal ordering between activities performed by the students. Given these limitations, the objective of our next research study was to construct an even better method for modeling and predicting students' engagement/disengagement behaviors represented using six different discrete labels (i.e. HK, LK, FG, KG, LE and NI).

For multivariate categorical time series, it is difficult to define suitable distance measure between observation sequences and '*model-based clustering*' appears to be a promising alternative [Pamminger et al., 2010]. In the followings, we briefly review the basic idea of model-based clustering and the issue of finding good initialization parameters; which we have already discussed in detail in Section 2.4.

3.5.2.1 Model-based clustering

Model-based clustering is a probabilistic method that results in a set of K mixture models (or clusters). All input observations belong to multiple clusters with different probabilities and each mixture component represents a different data distribution through a Markov chain. Hansen et al. [2017] also insisted on the use of mixture Markov chains to model sequential traces of students as they have the capability to capture *drift* in students' behaviors through different mixture components. Furthermore, an experiment performed by Cadez et al. [2003] revealed that predictions made with a simple (or non-mixture) first-order Markov chain is less accurate than the ones made using a mixture of Markov chains. Keeping in view the finding of [Cohen and Beal, 2009] which shows that the next action pattern of a student depends more likely on the previous pattern and not much on earlier patterns, we selected first-order Markov chains to represent the mixture

components.

Expectation-Maximization (EM) algorithm is a well-known iterative procedure to estimate finite mixture model parameters by maximizing the likelihood of observing a *complete data*. More precisely, mixture modeling framework assumes that each observation sequence s is generated by one of the K component distributions, however, its true membership label is unknown [Melnykov, 2016]. The EM algorithm aims to incorporate these missing labels. That is, given some observed data Y, the EM algorithm tries to find a model $\theta \in \Theta$ with maximum (log) likelihood estimation (MLE) [Gupta et al., 2011], where Θ is the symbol of parameter values. Formally:

$$\hat{\boldsymbol{\theta}}_{MLE} = arg\, max_{\theta \in \Theta}\ \log\ p(Y|\theta) \tag{3.2}$$

In order to find such a model, EM algorithm iterates over the following two steps until it reaches convergence (or some stopping criterion).

1. Expectation (or E) step: estimates the conditional expectation of complete-data log-likelihood function given the observed data.

2. Maximization (or M) step: finds the parameter estimates to maximize the complete-data log-likelihood from the E-step.

Finding an optimal 'global' maxima is challenging for EM and it usually ends up with one of the best 'local' maxima. However, initialization of the algorithm parameters plays a critical role in finding an optimal solution [Hu, 2015, Michael and Melnykov, 2016]. According to Gupta et al. [2011], performing a preliminary cheaper clustering (like K-means or Hierarchical) for initializing the EM algorithm is expected to give better results than random assignment. In their work, this approach is used for Gaussian Mixture Model (GMM) that describe probability distribution of continuous data, and clusters' means and covariance matrices are taken from the K-means results. Hu [2015] used hierarchical clustering for initialization of the EM algorithm for finding model parameters for GMM. However, this is

not straightforward in case of a mixture model for multivariate categorical data. The initial parameters required for the EM algorithm in this case are given in the next section. In this exploratory study, we propose a new approach for performing model-based clustering on categorical data which takes the results of K-means clustering algorithm as input to initialize the EM algorithm; hence, we named this method as "K-EM".

3.5.2.2 K-EM: Initializing EM with K-means Results

To cluster multivariate categorical data, EM algorithm requires the following three parameters to get started:

1. Number of mixtures (K).
2. Initial transition matrices for K mixtures.
3. Initial weights of K mixtures.

Like K-means, EM algorithm also requires a prior number of mixtures to be defined by the user which is one of the challenging problems for researchers. However, model-based clustering has the advantage of being supported by formal statistical methods to determine the number of clusters and model parameters [Magidson and Vermunt, 2002]. The two most commonly used methods which are based on 'information criterion' to select the optimal value of K are Bayesian Information Criterion (BIC) [Schwarz et al., 1978] and Akaike Information Criterion (AIC) [Akaike, 1998]. Both methods penalize complex models, thus, the models with the lowest BIC and AIC scores are better. The primary difference between both measures is that BIC penalizes heavily in contrast to AIC. Standard EM algorithm initializes the 'initial transition matrices for K mixtures' randomly where K is given by the user. And, each mixture component is usually assigned an equal initial weight (i.e. $W(C_1) = \cdots = W(C_K) = 1/K$),

As mentioned earlier, initializing the EM algorithm using partitioning obtained through K-means or Hierarchical clustering method is referred as

a practical solution [Gupta et al., 2011, Michael and Melnykov, 2016]. Our scenario required to work with categorical data, that is, the set of behavioral patterns P introduced in Section 3.5. One may argue here that another variant of K-means clustering called K-modes [Huang, 1998] is more suitable for categorical data that defines the similarity between two sequences based on matching elements. But, our data contains traces of different lengths and we considered the patterns' frequencies to compute distances between different traces; thus, we performed K-means clustering (see Maqsood et al. [2019], for details). This way traces having similar problem-solving pattern distributions were grouped together, highlighting the most frequent behaviors depicted by the students during assessment as discussed in Section 3.4.2.2. Considering the usefulness of the previously obtained clusters through K-means algorithm, in this exploratory work, we performed experiments by setting the three initial parameter values for the EM algorithm based on the results of K-means clustering, and, to the best of our knowledge, no work to date has reported results of this approach. K-EM is performed in three steps as given below.

1. Run K-means clustering algorithm on input data with multiple initial points for multiple iterations (to obtain stable results).

2. Use results from Step 1 to initialize the EM algorithm in the following manner.

   (a) Set the number of mixtures (K) equal to the number of clusters (K') obtained using K-means algorithm.

   (b) Construct a first-order Markov chain for each resultant cluster ($C_{k'}$) containing $S_{k'}$ sequences. Use these transition matrices as initial transition matrix for respective K mixture components.

   (c) Weights of K mixtures are set to the ratio of the number of sequences in each respective obtained cluster, that is, $W(C_k) = S_{k'} / \sum_{i=1}^{K'} S_i$.

Table 3.15: Summary of solved problems in both datasets

Data	Number of solved problems			
	Minimum	Maximum	Average	Total
Dataset1 (197 traces)	2	6	5	1033
Dataset2 (348 traces)	2	39	17	5792

3. Run the usual EM algorithm.

3.5.2.3 Data description

For this work, both datasets were available to us: Dataset1 and Dataset2. The experiment described in Section 3.5.1 shows the advantage of performing analyses at trace level over student level data. Thus, both datasets were transformed to trace level by defining temporal ordering between all the activities using 'timestamp' recorded with each activity. During data pre-processing, we removed sessions of length 1 as we needed to compute transition matrices of traces and make predictions, which is impossible for a single activity trace. The final processed data contain sequentially ordered activities for each Login-Logout session. Table 3.15 contains a summary of the remaining datasets.

All activities (in both datasets) are transformed into respective discrete engagement and disengagement behavioral patterns (as mentioned in Section 3.5). Table 3.16 contains 10 sample traces of students data with session lengths of sizes between 2 and 6 (showing respectively minimum and maximum number of solved problems each with a specific behavioral pattern from set P). Table 3.17 shows the frequency distribution of each behavioral pattern in both datasets.

Table 3.16: Students' sample traces (with lengths between minimum 2 and maximum 6, activities are separated by a hyphen '-')

Trace1:	HK-HK-LK-HK-LK
Trace2:	HK-HK-FG
Trace3:	KG-KG-LE-KG
Trace4:	HK-HK-HK-FG-FG-FG
Trace5:	HK-FG-KG-FG
Trace6:	LE-LE-LK-LK-LK
Trace7:	NI-NI
Trace8:	HK-HK
Trace9:	LK-HK-FG-FG-LK-LK
Trace10:	FG-HK-HK-LK

Table 3.17: Frequency distribution of behavioral patterns

Behavioral Pattern	**HK**	**LK**	**FG**	**KG**	**LE**	**NI**
Dataset1 *(N=1033)*	421	35	363	117	81	16
Dataset2 *(N=5792)*	2052	1771	677	53	1101	138

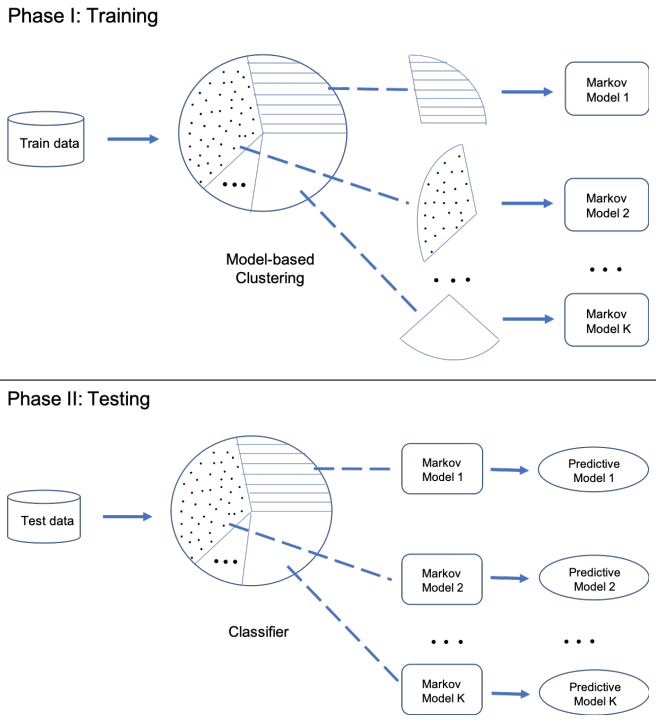


Figure 3.10: Methodology for constructing mixture Markov models and evaluating their prediction accuracy

3.5.2.4 Methodology

The methodology we adopted to construct mixture Markov models and evaluate their prediction accuracy is shown in Fig. 3.10.

- The input data is split randomly into 'train' and 'test' data using three different ratios (represented as Train-Test): 90-10, 85-15 and 80-20; to compare the performance of the algorithm on different data distributions.
- Model-based clustering is performed on train data which produces K mixture Markov models, each represented by a first-order Markov chain.

- Next for the test data, all sequences are first classified to the best mixture component (generated in the previous step using train data), by the *classifier*.

- Finally, for each mixture component having T_n test sequences, we predict next activity using respective Markov predictive model.

In the following subsections, we provide details of pre-existed algorithms selected for performance comparison, the classifier and predictive models with their accuracy computation.

A Comparison with existing algorithms

To compare the performance of the proposed K-EM method, we selected the following two existing algorithms which were also applied on the input data using the same methodology as given in Fig. 3.10.

1. EM [Dempster et al., 1977] — the original EM algorithm in which initialization is performed randomly.

2. emEM [Biernacki et al., 2003] — a variant of the EM algorithm in which Expectation-Maximization algorithm is also run in the initialization phase, as reflected by the prefix 'em'. The best model is then picked as the starting point (or initial model) followed by the actual EM algorithm.

B Classifier

The classifier estimates posterior probability of all test data sequences given K mixture Markov models (generated earlier through model-based clustering performed on train data, see upper-half of Fig. 3.10). Each sequence is then assigned to the best mixture component based on the highest posterior probability using Bayes decision rule. This procedure is explained in more

detail in Section C. Our code implementation of the classifier is available at GitHub [19].

C Predicting students' future behavior and computing prediction accuracy

The second objective of this research study was to predict students' future behavioral patterns so that their varying behaviors can be identified and referred for further actions (if needed). Also, prediction is a mechanism for validating developed learner models [Desmarais and Baker, 2012], which in our case represent students engagement/disengagement behaviors using mixture Markov chains. As mentioned earlier, Markov chains serve dual purposes of modeling and predicting sequentially ordered activities. With first-order Markov chain, we make the *Markovian* assumption that a student's future behavior is dependant on his/her current behavior only and not on the previous history. That is:

$$P(q_{i+1}|q_1, q_2, ..., q_i) = P(q_{i+1}|q_i) \tag{3.3}$$

Having a K mixture of Markov models and a corresponding first-order Markov chain for each obtained cluster, next activity predictions for each mixture component were made separately (as shown in Fig. 3.10). Cadez et al. [2003] showed that a mixture of first-order Markov chains is different than a simple (or non-mixture) first-order Markov chain and that making predictions with the prior approach resulted into better accuracy.

Prediction accuracy for each cluster is computed as the number of correct predictions divided by the total number of predictions made, see Eq. (3.4). The answer is multiplied by 100 to convert it into a percentage.

$$\begin{matrix}\text{Prediction accuracy} \\ \text{of Cluster}_i\end{matrix} = 100 \text{ x } \frac{\text{No. of correct predictions}}{\text{Total predictions}} \tag{3.4}$$

[19] https://github.com/r-maqsood/Mixture-Markov-Models-R.

Table 3.18: Base model prediction accuracy for both datasets

Dataset1	Pr(HK) = 421/1033 = 40.76%
Dataset2	Pr(HK) = 2052/5792 = 35.43%

However, in order to compare different algorithms (ran with different numbers of clusters, K), we need overall accuracy for each algorithm. Since the number of traces (or sequences) vary for all clusters, we computed weighted average prediction accuracy using Eq. (3.5).

$$\text{Prediction accuracy of Algorithm}_a = \frac{\sum_{i=1}^{K}(\text{Prediction accuracy of Cluster}_i \text{ x Traces in Cluster}_i)}{\sum_{j=1}^{K} \text{Traces in Cluster}_j} \tag{3.5}$$

D Base model

To compare the accuracy of predictive models developed using Markovian assumption with random guessing, we constructed base models for both datasets by adopting the notion of 'empirical probability' from statistics. This assumes that "*the probability of an event is the ratio of the number of outcomes in which a specified event occurs to the total number of trials*"[20]. In both datasets, highest frequent activity (from Table 3.17) is: "High Knowledge (HK)". Using this, base model prediction accuracy for both datasets is computed in Table 3.18.

We consider these probabilities as our base model for respective datasets, since, without having any other knowledge, the highest frequent activity is more likely to be predicted as the next activity by a predictive model. And we expect our sequential prediction models (or Markov predictions) to perform better than these base models for respective datasets.

[20]From: https://en.wikipedia.org/wiki/Empirical_probability

3.5.2.5 Experimental setup

All experiments related to model-based clustering in this work were performed using ClickCluct package of R [Melnykov, 2016], which actually provides implementation of the emEM algorithm. The algorithm converges if the difference between the log-likelihood of two subsequent iterations is less than $1e-10$. We used the same stopping criterion for the EM and K-EM algorithms, and, modified the existing code to implement the latter two methods. Following subsections explain the parameters used to construct mixture Markov models using the three algorithms, and, the method we used to select an appropriate cluster for each sequence (or trace) after convergence.

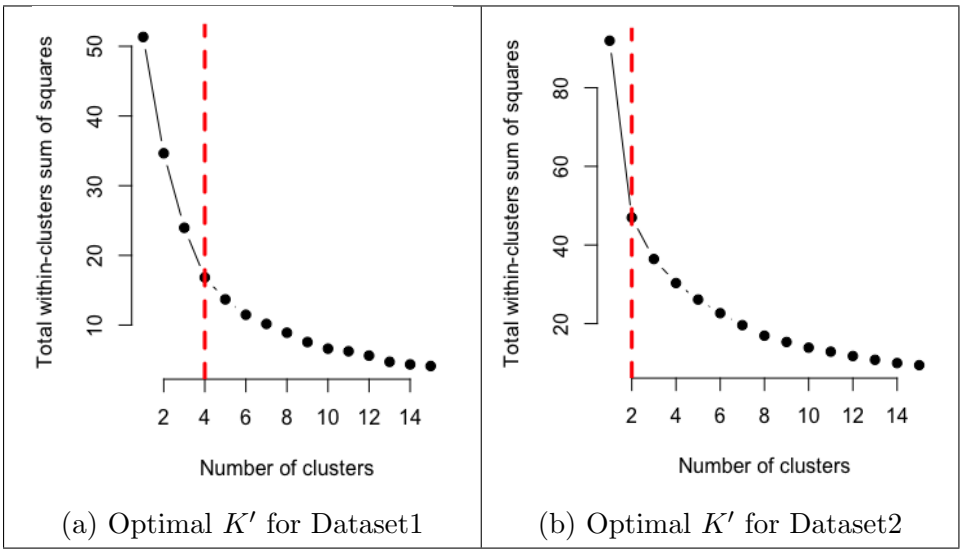
A Parameters for K-EM algorithm

First, we provide details of the K-means clustering as it was performed on both datasets, and, then we describe how the obtained results were used to initialize the EM algorithm. Once initialized, the usual EM algorithm was run on both datasets.

Step 1 — *Run K-means algorithm*: K-means is a widely used algorithm to perform unsupervised clustering to group data items having minimum sum of squared distances within clusters. However, it does not apply to categorical data directly [Huang, 1998]. Considering this limitation and the nature of our data, we computed proportion value for each pattern p_i per Login-Logout session using Eq. (3.1). And, K-means algorithm was run on both datasets as it was done in research study II, Section 3.4.2.2.

Briefly, the optimal number of clusters (K') for both datasets was obtained using NbClust method of R; which returned 4 and 2 values of K' for Dataset1 and Dataset2, respectively. To validate these optimal values, we employ the *elbow method* and plot these values, as shown in Fig. 3.11.

We can see that the optimal K' determined for both datasets using



(a) Optimal K' for Dataset1 (b) Optimal K' for Dataset2

Figure 3.11: Elbow method plots of optimal number of clusters *(obtained using NbClust method of R)* for K-EM algorithm – (a) Dataset1 (b) Dataset2

NbClust method are indeed good choices as also indicated by the elbow method. Next, the K-means algorithm was run on both datasets for 15 iterations with 25 initial random points to get stable clusters.

Step 2 — *Initialize EM with K-means results*: For EM initialization, we followed the steps as mentioned in Section 3.5.2.2; once initialized, the usual EM algorithm was run on both datasets using the respective initial models. More specifically, the following actions were taken to initialize the EM algorithm.

- Number of K mixtures are set to 4 and 2 for Dataset1 and Dataset2, respectively.
- Then, points 2(b) and 2(c) were performed as specified in Section 3.5.2.2.

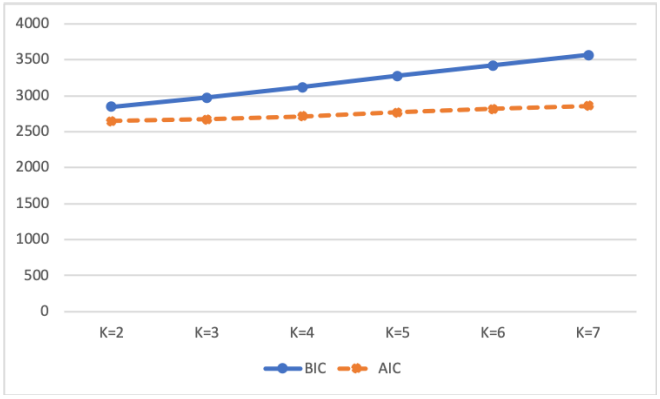
B Parameters for EM and emEM algorithms

Both algorithms were initialized randomly, however, first, we needed to determine the appropriate number of mixtures for both datasets. We computed BIC and AIC scores for Dataset1 and Dataset2 using models of a different number of mixtures, see Fig. 3.12 (values of K are shown on the horizontal axis).

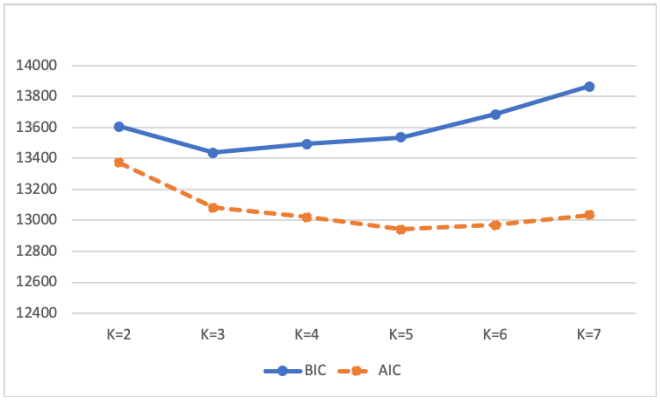
For Dataset1 (Fig. 3.12(a)), we can see that BIC and AIC scores increase with an increasing K value and both measures suggest that 2 is the optimal number of clusters. However, in case of Dataset2 (Fig. 3.12(b)), both measures disagree; that is, the lowest BIC score is achieved at $K = 3$ and the lowest AIC score is at $K = 5$. In such a situation, BIC-preferred model can be taken as a minimum size and AIC-preferred model as a maximum, and, any model can be picked within this range (preferably based on some other criteria) [Dziak et al., 2019]. In our case, the range for the optimal number of mixtures is 3 and 5, and we picked $K = 3$ arbitrarily. Thus, the EM and emEM algorithms were applied using $K = 2$ and $K = 3$ respectively on Dataset1 and Dataset2.

C Determining appropriate cluster for each sequence

All variants of the Expectation-Maximization algorithm (i.e., K-EM, EM, emEM) outputs a N x K matrix of estimated posterior probabilities wherein each cell contains: $z_{s,k}$ the probability that the s-th sequence belongs to the k-th mixture component. In other words, sequences are assigned to all the clusters with some probability distribution. However, once the algorithm has converged (or stopped), we need to assign each sequence to a single cluster. To determine the appropriate cluster for each sequence $s \in N$, we employ the most commonly used method for this purpose, that is, *Bayes decision rule* - which assigns sequences to the clusters based on the highest posterior probability.



(a) Optimal K for Dataset1



(b) Optimal K for Dataset2

Figure 3.12: Models comparison using AIC and BIC scores to determine optimal number of clusters for EM and emEM algorithms – (a) Dataset1 (b) Dataset2

3.5.2.6 Data analyses and results

In this section, we present results of model-based clustering performed on two real datasets using the three algorithms: EM, emEM, and K-EM. To compare the performance of these algorithms, we consider the following two measures of critical importance.

- Prediction accuracy of mixture Markov chains.
- Number of iterations required to reach convergence (or convergence rate for model training).

Table 3.19 and 3.20 contain the results of algorithms' next activity prediction accuracy of the three pre-mentioned methods on Dataset1 and Dataset2, computed using Eq. (3.5). Highest accuracy in comparison to K-EM method is shown in boldface. Prediction accuracy of each resultant cluster computed using Eq. (3.4) is provided in Appendix A.1. Remember that the optimal number of mixtures (K) for EM and emEM algorithms were determined based on information criterion (e.g. BIC and AIC measures). Whereas, for K-EM we relied on some very commonly used internal criteria (e.g. Silhouette-index, Beale-index, Dunn-index, etc.) to determine K' for the K-means algorithm. Specifically, for Dataset1, the optimal K is 2 for both EM and emEM algorithms and for K-EM is 4. And, for Dataset2 , the value of K mixtures is 3 for EM and emEM, and 2 for K-EM. Since both datasets are not very large, we, therefore, trained models using three different data distributions (i.e. 90, 85 and 80 percent) and computed next activity prediction accuracy on remaining (or test) data for providing detailed comparisons.

First, we compare the test data prediction accuracy obtained through the *Markovian* assumption using all variants of EM algorithm with *randomness* (i.e. base model) of respective datasets. As base model prediction accuracies were computed using complete datasets, we make comparison with

Table 3.19: Comparison of test data prediction accuracy of different algorithms for Dataset1 *(models constructed as described in Section 3.5.2.5; EM and emEM are run with K=2; K-EM with K=4 mixtures)*

Algorithm	**Dataset1 (197 traces)** Base model prediction accuracy = 40.76%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	59.70%	59.32%	57.63%
emEM	58.38%	63.45%	55.14%
K-EM	**64.40%**	**69.64%**	**68.39%**

Note: Algorithm accuracy value(s) of EM or emEM in boldface is higher or in tie with that of K-EM (after round-off).

Table 3.20: Comparison of test data prediction accuracy of different algorithms for Dataset2 *(models constructed as described in Section 3.5.2.5; EM and emEM are run with K=3; K-EM with K=2 mixtures)*

Algorithm	**Dataset2 (348 traces)** Base model prediction accuracy = 35.43%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	51.51%	50.68%	**56.22%**
emEM	52.50%	**53.74%**	**55.15%**
K-EM	**54.83%**	**53.97%**	54.40%

Note: Algorithm accuracy value(s) of EM or emEM in boldface is higher or in tie with that of K-EM (after round-off).

prediction accuracy of algorithms that comprises the effect of all clusters (see Eq. (3.5)).

As expected, results in Table 3.19 show that all algorithms achieve better prediction accuracy as compared to that of Dataset1 base model (40.76%) for all Train-Test ratios. Similarly, Table 3.20 shows the high performance of all algorithms in contrast to the base model of Dataset2 (35.43%). Hence, we conclude that Markov predictions, which in fact hold a sequential structure, perform better than random guess present in both datasets. In other words, a student's future behavior is better predictable from his/her recent behavior.

Next, we focus on comparing the prediction accuracy of the three algorithms ran on different train data proportions. In case of Dataset1, K-EM achieves higher next-activity prediction accuracy than EM and emEM algorithms (see Table 3.19) for all Train-Test distributions, that is, 64.40%, 69.64%, and 68.39% respectively for models constructed using 90, 85 and 80 percent train data. For Dataset2, our proposed method K-EM also performs better than both existing algorithms in most cases (i.e. 54.83% and 53.97% prediction accuracy for models trained using 90 and 85 percent data, respectively). While, EM and emEM algorithms achieve better prediction accuracy for a model trained using 80 percent data, i.e., 56.22% and 55.15%, respectively, versus 54.40% accuracy of K-EM. Nevertheless, based on these results, we can claim that the proposed K-EM method achieves better prediction accuracy than both EM and emEM algorithms on both datasets.

We also compare convergence rates of the three algorithms on models trained using 90 percent data in Table 3.21, showing the best value(s) in comparison to K-EM method in boldface. Results show that K-EM method requires the least number of iterations for training the model in contrast to the original EM algorithm for both datasets. However, contradictory results are obtained for the emEM algorithm, that is, K-EM performs better than emEM for Dataset1, while the opposite is true for Dataset2 (emEM performs

Table 3.21: Comparison of convergence rates of the training models constructed using 90% train data *(no. of mixtures same as in Table 3.19 and 3.20 for Dataset1 and Dataset2, respectively)*

Algorithm	**Dataset1** (197 traces)	**Dataset2** (348 traces)
EM	180	79
emEM	121	**26**
K-EM	**68**	31

better).

In summary, our results based on both evaluation measures show that K-EM outperforms the original EM algorithm, as well as it achieves better prediction accuracy than emEM algorithm; whereas, contradictory findings are observed for convergence rates of the two algorithms. Although, we can claim that overall the proposed K-EM method achieves better results when tested on two real datasets (one small and another of medium size). However, one may have a concern here that the algorithms are compared with different number of mixtures (i.e. EM and emEM is run with $K = 2$ and K-EM with $K = 4$ for Dataset1, and $K = 3$ for EM and emEM, $K = 2$ for K-EM for Dataset2); and, this could be a potential reason for different results. Therefore, we performed further experiments to compare the results of the proposed K-EM method with two pre-existing algorithms using the same number of mixtures (as used by the K-EM) for the respective dataset.

Table 3.22 and 3.23 contain new results of comparisons between the three algorithms based on prediction accuracy for Dataset1 and Dataset2, respectively, using the same number of clusters as that of K-EM. Results from Table 3.22 show that emEM algorithm performs better than K-EM with 90 percent train data. However, in all other cases, K-EM performs better than both algorithms on Dataset1. Therefore, we can conclude that

Table 3.22: Comparison of test data prediction accuracy of different algorithms for Dataset1 *(models constructed using the same number of clusters as of K-EM, i.e. K=4 mixtures, for all algorithms)*

Algorithm	**Dataset1 (197 traces)** Base model prediction accuracy = 40.76%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	62.43%	57.99%	55.22%
emEM	**67.66%**	62.93%	58.31%
K-EM	64.40%	**69.64%**	**68.39%**

Note: Algorithm accuracy value(s) of EM or emEM in boldface is higher or in tie with that of K-EM (after round-off).

the proposed method also obtains better accuracy using the same number of mixtures. For Dataset2, K-EM performs better than emEM algorithm with all Train-Test ratios. While, EM performs equally likely to K-EM with 90% train data and slightly better for the remaining two data distributions, i.e. 54.55% versus 53.97% for 85 percent train data and 55.25% versus 54.40% for 80 percent train data, (see Table 3.23).

Table 3.24 shows the convergence rate of the three algorithms ran with the same number of clusters on 90 percent train data. Our method converges faster than EM algorithm for Dataset1 and emEM performs better than K-EM. While, for Dataset2 , EM algorithm performs better than the K-EM which in return performs better than the emEM algorithm.

In Table 3.25, we present a summary of all the results obtained previously using the three algorithms applied to two datasets with (1) the optimal number of mixtures for the respective algorithms, and, (2) using the same number of mixtures as of K-EM. The table shows results of the two evaluation measures used in this work: (1) test data prediction accuracy (computed

Table 3.23: Comparison of test data prediction accuracy of different algorithms for Dataset2 *(models constructed using the same number of clusters as of K-EM, i.e. K=2 mixtures, for all algorithms)*

Algorithm	**Dataset2 (348 traces)** Base model prediction accuracy = 35.43%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	**54.96%**	**54.55%**	**55.25%**
emEM	53.13%	51.32%	51.25%
K-EM	**54.83%**	53.97%	54.40%

Note: Algorithm accuracy value(s) of EM or emEM in boldface is higher or in tie with that of K-EM (after round-off).

Table 3.24: Comparison of convergence rates of the training models constructed using 90% train data *(no. of mixtures same as in Table 3.22 and 3.23 for Dataset1 and Dataset2, respectively)*

Algorithm	**Dataset1** (197 traces)	**Dataset2** (348 traces)
EM	105	**18**
emEM	**65**	36
K-EM	68	31

on three different train data proportions), (2) convergence rate (of models trained with 90% train data). We highlight the 'best' and 'second-best' performances in order to compare the performance of K-EM with both EM and emEM algorithms. Clearly, results show that the next activity prediction accuracy of K-EM method took lead over randomly initialized EM and emEM algorithms in most cases. EM algorithm performs slightly better than the K-EM method for Dataset2 with an equal number of clusters. Similarly, K-EM resulted in better convergence rate than EM algorithm in most cases and remains in competition with that of emEM algorithm.

A Visualizing and Interpreting Students' Problem-Solving Behaviors

Fig. 3.13 and 3.14 contain Markov models[21] of each resultant cluster for Dataset1 and Dataset2 , respectively; obtained using the K-EM algorithm on 90% train data. States of the Markov chains (shown by circles) represent six discrete engagement/disengagement behavioral patterns and the size of each state is proportional to its percentage in the respective cluster to show dominant pattern(s) in the respective problem-solving sessions[22]. The thickness of each edge is proportional to the transitional probability between respective states (scaled by a constant factor). Transition probabilities greater than 32% are displayed only to highlight prominent behaviors. In the followings, we interpret the problem-solving behaviors of the students as reflected by the states and transition probabilities of a first-order Markov chain for each obtained cluster.

[21] All plots were drawn using r-igraph: https://igraph.org/r/ .

[22] Furthermore, states are filled with different colors to highlight their meanings. For example engagement behavior reflected with either confidence level is represented by two states, FG and LE, which are given the same color (yellow) in the images. Similarly, states representing disengagement behaviors: KG and NI, are shaded with the same color (blue). High knowledge (HK) and low knowledge (LK) states are differentiated with gray and white colors, respectively; *see colored pictures in online PDF version.*

Table 3.25: Results summary of the three algorithms applied on two datasets, compared using: (1) the optimal number of mixtures for respective algorithms, (2) the same number of mixtures as of K-EM

Dataset	**Algorithm**	**Evaluation Criteria**			
		Test Data Prediction Accuracy			**Convergence Rate for 90% Train Data**
		90% train data	85% train data	80% train data	
Dataset1[a]	EM	*		*	
	emEM		*		*
	K-EM	**	**	**	**
Dataset2[a]	EM		*	**	
	emEM	*	**[c]	*	**
	K-EM	**	**[c]		*
Dataset1[b]	EM				
	emEM	**	*	*	**
	K-EM	*	**	**	*
Dataset2[b]	EM	**[c]	**	**	**
	emEM	*			
	K-EM	**[c]	*	*	*

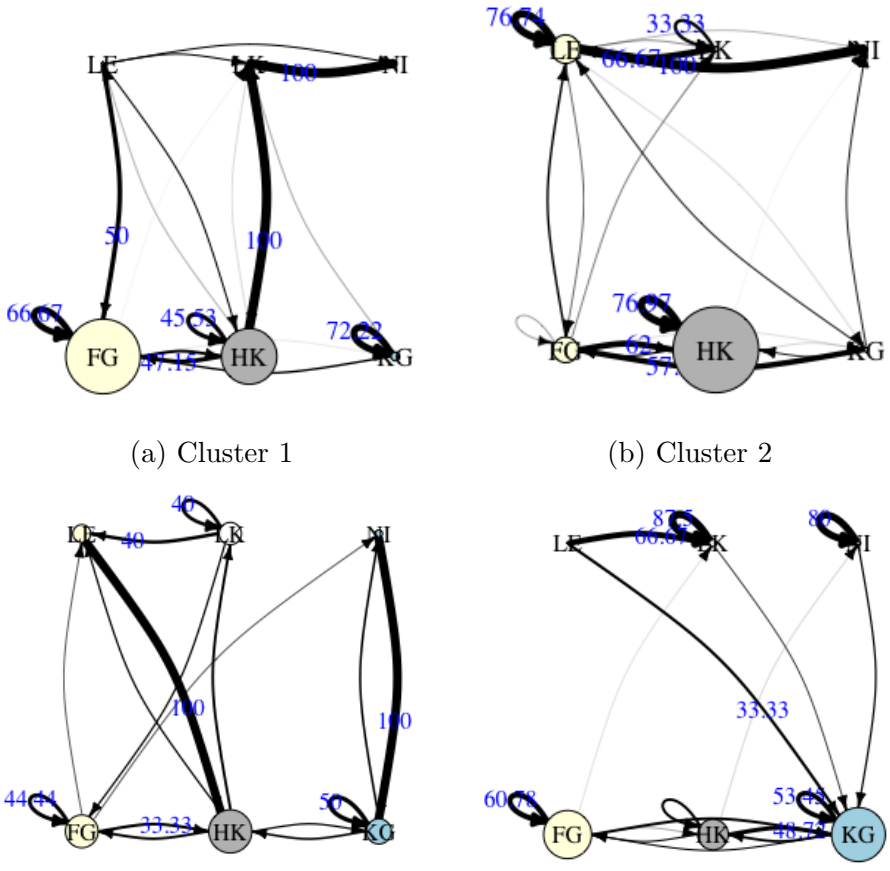
[a]The optimal number of mixtures are used by the respective algorithms.

[b]Same number of mixtures are used by all the algorithms as of K-EM.

** Best performance

* Second best performance

[c]Both methods performed equally likely.



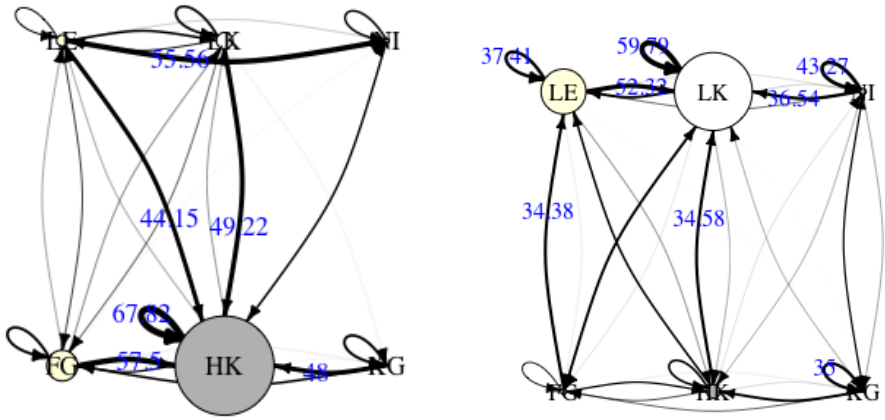
(a) Cluster 1

(b) Cluster 2

(c) Cluster 3

(d) Cluster 4

Figure 3.13: Four obtained Markov chains for Dataset1 : (a) Cluster 1: *77 traces* ; (b) Cluster 2: *56 traces* ; (c) Cluster 3: *7 traces* ; (d) Cluster 4: *38 traces* ; The size of each state is proportional to its percentage in the cluster and thickness of each edge is proportional to the transitional probability between respective states (scaled by a constant factor).



(a) Cluster 1 (b) Cluster 2

Figure 3.14: Two obtained Markov chains for Dataset2 : (a) Cluster 1: *159 traces* ; (b) Cluster 2: *155 traces* ; The size of each state is proportional to its percentage in the cluster and thickness of each edge is proportional to the transitional probability between respective states (scaled by a constant factor).

Students' behaviors in Dataset1 In Fig. 3.13(a), Cluster 1 represents the largest subgroup of traces in Dataset1 (i.e. 43%) and depicts positive engagement of the students during assessment. It contains traces reflecting fill-knowledge gap (FG) activity as the most dominant behavior followed by the high knowledge (HK) behavior. In FG state, students attempt to fill their knowledge gap(s) through detailed feedback (mainly) for wrong answers [Maqsood and Ceravolo, 2019]. A high transition probability of self-loop on FG activity (66.67%) shows that the students in case of wrong response(s) majorly focused on learning from the (detailed) feedback available for each submitted problem. Also, in cases when students show high knowledge (HK), they moved to the FG activity for incorrect answers. Finally, transitions from upper-half to lower-half of the chain show a change in the students' confidence level from low to high in respective knowledge states. This behavior is indeed desirable, that is, under-confident students should gain confidence in the subject domain over time.

The second largest subgroup of traces found in Dataset1 is shown by Cluster 2 in Fig. 3.13(b), which comprises of 31% students' problem-solving sessions. It contains sessions of the students having high knowledge (HK) in the subject domain who gave more correct answers with high confidence (see the state's size and high probability self-loop transition on HK activity, i.e. 76.97%). Students also depicted highly engaged behavior during the assessment as reflected by another frequent activity, fill-knowledge gap (FG). A high probability transition from FG to HK activity (62%) is a reflection of the students' engagement during the assessment. Similarly, positive engagement is found for responses given with low confidence, that is, through a high probability self-loop on learn (LE) state and more incoming transitions.

Cluster 3, representing only 4% of the traces, reflects mixed behaviors of the students who attempted problems with varying behaviors in general. High knowledge (HK) activity is slightly more prominent, followed by high fill-knowledge gap (FG) and knowledge-gap (KG) states in decreasing order.

In Cluster 4 (containing 21% of the total traces), disengaged behavioral pattern, that is, 'knowledge gap' (KG) is the most prominent activity with more incoming transitions (including one with 48.72% from HK state) as well as self-loop (53.45%). This shows that the students having high confidence in wrong answers did not request the available feedback which could have helped them in answering more answers correctly. Hence, traces in this cluster reflect their disengagement during the assessment. Another observable activity in this cluster is fill-knowledge gap (FG) with a high probability self-loop transition (60.78%) which shows recurrent learning approach of the students having little knowledge, as explained earlier.

Students' behaviors in Dataset2 Cluster 1 represents 51% of the traces of high knowledge students (see the size of HK state in Fig. 3.14(a)). The state has a high probability of self-loop (67.82%) and a transition probability of 57.5% from FG to HK that shows the engagement of the students during the assessment. Furthermore, incoming transitions from the LK and LE states to HK state show positive behavior of the students during the assessment. This shows that students who initially started solving problems with low confidence gained confidence in their knowledge. Again, this kind of behavior is desirable that students answering questions with low confidence improves their confidence level over time and gives more correct answers with a high confidence level. Students' behaviors observed in this cluster are very similar to those of the second largest subgroup of Dataset1, that is, Cluster 2.

Cluster 2 captures 49% of the sessions reflecting engagement behaviors of low confident students (i.e. LK and LE states are prominent with decreasing order). Existence of the self-loop (59.79%) and incoming transitions to LK state show that the students acquire a correct knowledge of the subject domain but they have doubts about it [Gardner-Medwin and Gahan, 2003]. Learning (LE) is the second frequent activity observed in this cluster which shows engagement behavior of the students and lately they gave correct

answers with a high ratio (see 52.32% transition from LE to LK state). Transitions from the lower-half of the Markov chain to respective states in the upper-half reveal a change in the students' confidence from high to low at a very early stage.

In summary, visualization of the resultant mixture Markov models provides substantial insights about the students' problem-solving behaviors in both datasets. Through these plots, a class teacher can better understand the strengths and weaknesses of the different subgroup of the students. For example, this could be a point of concern for the class teacher to further investigate the potential reason(s) for the high ratio of traces (49%) with low confidence observed in Cluster 2 of Dataset2. In our opinion, it could be either due to the (high) difficulty level of the posed questions or the perceived toughness of the course by the students, which made them felt low confident about their (correct) knowledge. Similarly, some students having high confidence in wrong responses depicted disengaged behavior during the assessment (see Fig. 3.13(d) – Cluster 4 of Dataset1), and they need special attention of the class teacher in order to identify possible difficulties they faced during confidence-based assessment.

3.5.3 Related Work

3.5.3.1 Measuring Student Engagement

There are several methods used in the existing literature for data collection and estimating students' engagement behavior. For example, Chapman [2003] reported a number of alternative methods used by the researchers, including: students' self-report engagement level (through questionnaires), checklists and rating scales - done by the teachers, direct observations of students in a class, (students') work sample analyses (e.g. project, portfolio, etc.), and, case studies. As mentioned earlier, our focus is on analyzing students' interactions data recorded by a computer-based assessment system. Therefore, in the following, we discuss attributes and methods used to mea-

sure student engagement by related works only which have taken students' logged data as an input.

Hershkovitz and Nachmias [2009] referred to engagement as an attribute of motivation during learning and used Hierarchical clustering algorithm to identify the best attributes that mapped on existing theories of motivation. They identified the following two variables to determine student engagement: *time on task percentage* and *average session duration*. Cocea and Weibelzahl [2009] also linked engagement with students' motivation in a subject or domain and estimated it using: *frequency* and *effort (or time) spent* on both reading pages and quizzes attempted by the students as they interacted with three different learning environments. Students' sessions were labeled as 'engaged' or 'disengaged' by human experts based on a set of rules defined earlier from manual analysis of the data [Cocea and Weibelzahl, 2007]. Eight data mining techniques were then used to construct a prediction model for student (dis)engagement, for example, Bayesian nets, Logistic regression, Decision tree, etc. Their supervised approach relied on pre-analysis of the data performed by human experts to identify a suitable length of traces which is data-dependent. Hence, the re-usability of the implemented method is reduced extensively. Whereas, we adopted an unsupervised approach using a probabilistic model that takes care of traces of different lengths.

Beal et al. [2006] adopted the notion of students' active participation in a current task and classify students' problem-solving activities into five different levels of engagement using: *response correctness*, *time spent per problem* and *help usage*. Hierarchical clustering was applied to proportion scores of these patterns to analyze students' use of an intelligent tutoring system (ITS). Another experimental study presented in [Brown and Howard, 2014] uses on-/off- task notations to refer to engaged and disengaged behaviors, respectively. Specifically, they used *response correctness*, *time on task* and *triggered events (i.e., keyboard strokes and/or mouse movements)*; attributes to label students' actions as engaged or disengaged. Engagement

is considered as one of the affective states in [Pardos et al., 2014] which is determined using *number of correct answers*, *proportion of actions in a time frame*; *number of reattempts*, *hints requested* and *fail on first attempt*. Human experts' (in field) observations were synchronized with student logged data to define a mapping between recorded interactions and various affective and behavioral states observed by the experts. Eight classification methods including Decision trees, Naive Bayes, Step regression and others were used to build a model for automatic detection for each effective state separately.

The literature review shows the potential of students' logged interactions to determine their level of involvement in the learning process. However, the classification of students' problem-solving activities into engagement/disengagement behaviors depends on the problem domain and collected data attributes. As mentioned earlier, we used a classification scheme defined in [Maqsood et al., 2019] for mapping students' problem-solving activities into six behavioral patterns reflecting their engagement and disengagement during confidence-based assessment. Our work is distinguished from prior works as we have analyzed sequential traces of students' interactions to understand their progression from one behavioral state to another using more sophisticated probabilistic model.

3.5.3.2 Modeling and Predicting Humans' Behaviors using Probabilistic Methods

Although several techniques have been presented in the literature to extract meaningful information from students problem-solving traces recorded by computer-based learning environments, for example: clustering [Beal et al., 2006, Boroujeni and Dillenbourg, 2018, Hershkovitz and Nachmias, 2009, Köck and Paramythis, 2011], classification [Cocea and Weibelzahl, 2009, 2011, Maqsood et al., 2019, Pardos et al., 2014], evolutionary method [Romero et al., 2004], Bayesian network [Muldner et al., 2011], etc. Our focus on a family of probabilistic approaches used to model and/or pre-

dict human behavior. In this section, we discuss some applications of different methods specifically including Markov chain, hidden Markov model and mixture of Markov chains.

Authors in [Taraghi et al., 2015] modeled students' question answering patterns (i.e. right or wrong answer) using second-order Markov chains to construct their profiles. Another application of Markov chains to capture and predict users' behaviors is given in [Khalil et al., 2007], where each trace contains a user's navigational pattern on a website. A simple K-means algorithm is used to group users having similar web navigation behaviors. Each cluster is then represented by a Markov chain and a user's future behavior is predicted accordingly. Their work is limited as it restricts a user's behavior to be represented by only one Markov chain. Whereas, our approach of clustering similar Login-Logout sessions using mixture Markov chains allows the flexibility of capturing a change in a student's behavior from one session to another. Furthermore, model-based clustering is a more sophisticated method to group traces of different lengths in contrast to distance-based clustering approaches like K-means and Hierarchical clustering algorithms [Cadez et al., 2003] used in some prior works, e.g., [Khalil et al., 2007, Taraghi et al., 2015].

Simple Markov chains are restricted to observable data only, whereas, sometimes it is important to identify underlying hidden information to represent internal cognitive behaviors of the users. Hidden Markov Model (HMM) is another very popular probabilistic approach amongst researchers to analyze and model humans' behaviors, where the hidden or latent states overcome the pre-mentioned limitation of Markov chains. For example, Beal et al. [2007] captured students' problem-solving behaviors using HMM where latent states reflect their different levels of engagement (i.e. low, medium, high) with an ITS. Also, in [Fok et al., 2005] a classification model is developed using a hidden Markov model to characterize students showing different content access preferences while interacting with an e-learning system.

Bouchet et al. [2013] used the Expectation-Maximization (EM) algorithm to cluster students' profiles participating in a self-regulated learning environment. Although resulted clusters reveal distinct problem-solving behaviors of the students, sequential ordering of the activities is not considered by the authors which may have offered useful insights to further distinguish between students and improve system adaptation. Cadez et al. [2003] also utilized model-based clustering to analyze web navigation patterns of a website users where each trace contains sequential ordering of web pages accessed by a user. Their approach is quite related to that of ours in a way that they also used a mixture of first-order Markov chains to model and analyze sequential categorical data representing users' dynamic behaviors. However, our method is a modification to the original EM algorithm which improves the prediction accuracy for each resultant cluster.

Recent work on understanding students' procrastination behavior [Park et al., 2018] has utilized model-based clustering where each mixture component follows a Poisson distribution to show students' activities in an online course. Hansen et al. [2017] also used a mixture of Markov chains to model the dynamic behaviors of the students captured by an e-learning system. Their proposed method estimates mixture components (i.e. first-order Markov chains) using a modified K-means clustering algorithm. Authors made a similar assumption that students behaviors may change over time and thus performed activity sequences analyses at the session level, which associates multiple Markov chains with an individual student representing his/her different problem-solving sessions. Despite having some similarities, our approach is an extension to the standard EM algorithm which is more accurate for estimating the likelihood of related sequential traces and generates (a mixture of) Markov chains with better prediction accuracy.

3.5.4 Discussions

Model-based clustering is a probabilistic method to generate a finite mixture of Markov models to represent underlying distributions of the data through different mixture components (or clusters). Each mixture component is represented by a first-order Markov chain. Expectation-Maximization (EM) algorithm is a well-known method to perform model-based clustering on multivariate categorical time series. However, the quality of the obtained clusters is dependent on its initialization [Michael and Melnykov, 2016], which is performed randomly in the original algorithm.

In this work, we employed model-based clustering to model and predict students' engagement and disengagement behaviors through their logged interactions during confidence-based assessment. We proposed a new method to identify a mixture of Markov models for discrete data by initializing the EM algorithm using the results of K-means clustering algorithm and named it as "K-EM", see Section 3.5.2.2 for details. To predict students' next activity behavioral pattern, we make the *Markovian* assumption that a student's future behavior is dependent on his/her most recent behavior only and not on the previous history. Experiments are carried out on two real datasets (i.e. Dataset1 and Dataset2) containing sequentially ordered discrete data items representing students' problem-solving behaviors in the confidence-based assessment, sample data is shown in Table 3.16.

The proposed K-EM method is compared with two existing algorithms, namely: EM and emEM. The three algorithms are applied on both datasets and compared using the following two evaluation measures: (a) prediction accuracy, (b) convergence rate. Our results (summarized in Table 3.25) show that the next activity prediction accuracy of K-EM method outperforms both pre-existing algorithms in most cases. K-EM also converges faster than the EM algorithm, however, contradictory results are obtained in comparison to emEM algorithm. Additionally, we compared the overall prediction accuracy of these algorithms (computed using Eq. (3.5)) with the chance

of random guessing present in each dataset (i.e. 'base models' described in Section 3.5.2.4 – paragraph D) and found that our Markov predictive models perform much better; for example, see results in Table 3.19 and 3.20. This finding confirms that the students' interactions data mapped onto their behavioral patterns, hold some structural information which is better captured through sequential ordering.

Although there is some criticism on initializing the expectation-maximization algorithm based on other clustering methods (like K-means, Hierarchical clustering) [Michael and Melnykov, 2016]; this approach shows better results in our study using the two real datasets, see Section 3.5.2.6 for a detailed view on different obtained results. In fact, the good prediction accuracy of each resultant cluster shows the potential usability of our proposed method, see detailed results given in Appendix A.1. As mentioned by the authors, critics come from the fact that this approach relies on the results of another clustering method which may impose some restrictions on the resultant mixture components. Yet, it is considered as a practical alternative to random initialization of the original EM algorithm [Gupta et al., 2011, Michael and Melnykov, 2016]. Our experiments using categorical sequential time series result in better prediction accuracy and requires less number of iterations to reach convergence, in contrast, to randomly initialized approaches. However, the size and few numbers of discrete states are the limitations of both datasets used in this work. Hence, application of the K-EM method on a larger dataset with more discrete states is essential to validate this approach.

Given the heterogeneous nature of students' behaviors (as shown in Fig. 3.13 and 3.14), increasing the prediction accuracy for each obtained cluster is another future challenge. A naive approach to further improve the prediction accuracy is to use a higher order of Markov chains for representing clusters (that is, a mixture of higher-order Markov models), where, in a k-order Markov model the probability of a future state depends on k

previous states. But, an increase in the accuracy would come with a cost of an increase in time and space complexity which is not favorable especially if the developed model is to be implemented in an online setting (e.g., an adaptive system).

Visualization of the resultant mixture Markov models reveals very useful insights for class teachers about students' problem-solving behaviors, as discussed in Section 3.5.2.6 – paragraph A. Implementation of these plots in an online assessment tool would provide easy access to various analytics to class teacher(s) who can identify strengths and weaknesses of the students, and, may modify teaching strategies accordingly. Also, the developed method can be implemented in an adaptive system which can automatically identify students with undesirable behavior (using our predictive model) and offers personalized feedback to diverse groups of the students. However, it may be difficult to provide any assistance in some cases, e.g. Cluster 3 of Dataset1 that shows mixed behaviors of the students (see Fig. 3.13(c)). Here, we also highlight that the two larger subgroups of both datasets (i.e. Cluster 2 and Cluster1 respectively of Dataset1 and Dataset2) reveal very similar behaviors of the students belonging to different populations. This is very promising for constructing a mixture of Markov models representing the most common behaviors of the students through different mixture components, which can be identified by the domain expert(s). And, each new student can then be assigned to a suitable mixture component after collecting his/her problem-solving behaviors. Evaluating the prediction accuracy and testing this model on different populations is also a point of investigation for future work.

Additionally, plots of the resultant mixture components for both datasets (shown in Fig. 3.13 and 3.14) reveal that the students depicted different problem-solving behaviors in different Login-Logout sessions. Thus, in agreement to [Hansen et al., 2017], we conclude that it is advantageous to analyze students' interactions at a lower-level, i.e. Login-Logout sessions. And,

mixture Markov chains yielded through model-based clustering is a useful mechanism to capture students' diverse behaviors. Furthermore, students' personalized behavioral profiles can be easily constructed by extracting the resultant mixture components for each specific student that reflect his/her behavior in different Login-Logout sessions – all combined into a vector of related Markov models.

4

Conclusive Remarks

In this last chapter of the thesis, we provide conclusive remarks in detail through the following sub-topics.

- In the first section, we provide a brief overview of the need and purpose of conducting this research work.
- The second section contains our conclusions based on the results and findings of the research studies performed to answer the research questions.
- The third section highlights the contributions of this research work.
- Subsequently in the next section, we discuss the limitations of our methodology and provide some recommendations for improvements.
- Finally, we discuss potential future work directions of this thesis.

4.1 Thesis Overview

This doctoral research work aimed to analyze and model students' behavioral dynamics in confidence-based assessment, which drive assessment taking students' confidence levels in addition to their answers to the questions.

The confidence level specified by a student reflects his/her expectancy about the knowledge (demonstrated through a recently submitted problem), which could be either accurate or inaccurate. Thus, confidence-based assessment offers additional parameters related to students' performance in contrast to traditional assessment which solely relies on response's correctness. The difference between expected and actual performance was exploited by Bruno [1995] to ascertain students' knowledge level. Hunt and Furustig [1989] considering the confidence measure on a binary scale (i.e. high or low level), defined four knowledge regions: *uninformed* (wrong answer given with low confidence), *doubt* (correct answer given with low confidence), *misinformed* (wrong answer given with high confidence), and, *mastery* (correct answer given with high confidence). We referred to these regions as 'confidence-outcome categories' and renamed them after Vasilyeva et al. [2008]. We believed that these confidence-outcome categories have much more potential to provide useful insights about students' productive and unproductive problem-solving behaviors [Maqsood and Ceravolo, 2018].

Investigating students' intended behaviors become even more crucial when they are interacting with computer-based learning environments due to the non-presence of a human teacher, who is generally good at recognizing different needs of the students through observations. Student engagement reflects their active or inactive involvement in an ongoing task or process. Several attempts have been made to determine student engagement using both theoretical and non-theoretical approaches (discussed in detail in Section 2.1.1). We identified that behavioral and online engagement are the most suitable terms to differentiate between productive and unproductive behaviors through analyzing activities performed by the students while interacting with computer-based learning/assessment systems.

Despite having several benefits of confidence-based assessment over traditional assessment approach, the study of the existing literature shows that no attempt is made to analyze problem-solving behaviors of the students in

relation to the four pre-mentioned knowledge regions. This gap laid down the basis of our first research question – *RQ-1: What behaviors can be used to determine student engagement/disengagement in confidence-based assessment?* However, determining student engagement is not just limited to the identification of a set of parameters which categorize problem-solving behaviors. But, the need to construct a mechanism that can extract and capture these behaviors from trails of sequential activities performed by students, becomes essential to represent each individual learner. Thus, our second research question was – *RQ-2: How can we model these behaviors to construct students' behavioral profiles?*

In the following section, we provide details of our findings and conclusions made to answer the two research questions.

4.2 Conclusions

To answer our first research question, RQ-1, our first objective (i.e. Objective I) was to identify the potential parameters which can be used to define student engagement and disengagement behaviors. The computer-based assessment tools used for experimentation and data collection provided task-level (corrective) feedback to the students which contains correct solution along with an explanation (i.e. elaborated feedback), to help students to fill their knowledge gap(s). Our findings from the first research study (presented in [Maqsood and Ceravolo, 2019]), revealed that the students after providing distinct confidence-outcome category answers show a different response towards the available corrective feedback. In other words, some students tried to learn from their mistakes through the corrective feedback in case of wrong answers given with high or low confidence, however, some other students completely ignored the feedback and just focused on attempting different questions.

Our results also indicated that feedback-seeking has a positive impact on students' confidence level and response correctness in the subsequent

attempts. That is, students who sought feedback relatively gave more correct answers in the next question, and, under- and over-confident students adjusted their confidence level accordingly. For example, under-confident students who were in *doubt* about their knowledge and gave the correct answer with low confidence, increase their confidence level to high after seeking feedback. Vasilyeva et al. [2008] also observed similar positive impact of feedback-seeking on students' confidence level. We also investigated the correlation between feedback reading time and answers given with distinct confidence-outcome categories, but, we did not find any useful results.

We conclude that task-level feedback which plays a key role in fostering students' learning, offers useful information about their problem-solving behaviors in confidence-based assessment. Whereas, seeking and utilization of available feedback is much dependent on students' engagement level [Mory, 2004, Timmers et al., 2013], which may vary over time. Therefore, it is important to investigate the response of students towards the available feedback as they progress in the ongoing assessment process to determine their interest or involvement.

As the next step, we defined a scheme to classify students' activities into engagement or disengagement behaviors [Maqsood et al., 2019], which was our second objective (i.e. Objective II) for RQ-1. The six identified behavioral classes not only differentiate students based on their response correctness but also consider their confidence level and response towards the available corrective feedback. We reasoned that the combination of these three problem-solving parameters reflects the students active or inactive participation in the assessment process. Besides, these engagement/disengagement behavioral patterns provide information about students' confidence accuracy and knowledge level.

'Time spent' on different activities is a critical variable for computer-based assessment because typically a system does not allow to distinguish between time actually spent on a task or time where the user is perform-

ing some other activity (also known as "off-task" behavior). For similar reasons, it has been used as an integral component of many student engagement detection models (see summarized parameters of different models in Table 2.2). Our proposed student engagement/disengagement model is more generalized as it does not rely on time limits defined by human experts based on data analyses performed on collected data.

Our experiment to identify groups of similar behaviors resulted in four clusters, three of them defining different positive engagement behaviors and one related to negative engagement or disengagement behaviors. We also explored the correlation of these behavioral groups with students' actual performance. Although no significant performance difference was observed between all the behavioral groups, a meaningful difference in performance scores in confidence-based assessment was seen in the groups showing engagement versus disengagement behaviors. The results also showed that high and low-performance students of the class relate differently to these engagement and disengagement behaviors. Based on these results, we conclude that the proposed scheme of classifying students' activities has the potential to identify their positive and negative engagement behaviors during confidence-based assessment.

Similarly, to answer the second research question, RQ-2 (given at the end of Section 4.1), from the study of various data mining and machine learning methods, we found that probabilistic approaches are more favorable for modeling and predicting sequential traces of human activities. The resultant models also support an easy interpretation of users' intended behaviors. Therefore, our third objective (i.e. Objective III) was to develop a technique to model, analyze and predict students varying engagement and disengagement behaviors using probabilistic models. This leads us to the development of a new approach for model-based clustering which produces K mixture Markov models. And, it is assumed that each input observation sequence is generated by one of these mixture components (or clusters)

representing different probability distributions. Our proposed method, we called it K-EM, achieved better prediction accuracy and convergence rate in comparison to the two pre-existed algorithms when tested on two real datasets.

The visualization of the resultant Markov chains revealed dynamic behaviors of the students within different problem-solving (or Login-Logout) sessions. More interestingly, we found some similar behaviors between Dataset1 and Dataset2 which were collected from students studying in different countries and they solved problems of different subjects[1]. To be more precise, one of the Markov chains resulted from both datasets were quite similar (i.e. Cluster 2 of Dataset1 and Cluster 1 of Dataset 2, shown in Fig. 3.13 and 3.14). This shows that there are some common problem-solving behaviors which are independent of the subject domain and the type of posed questions. This further highlights the potential of our engagement/disengagement classification scheme. Cocea and Weibelzahl [2011] and Tan et al. [2014] have also shown empirically that students' engagement detection model can be compared across different domains.

Furthermore, comparison between the accuracy of the Markov predictive models[2] and random guess present in both datasets[3], showed that students problem-solving behaviors hold a structure that is better represented through temporal ordering between different activities. That is, students future behaviors are more dependent on previous behavior and predictions made using Markov models achieved better accuracy.

In our approach, we performed data analyses at the trace level, that is, each unique Login-Logout session is treated separately which contains the collection of temporally ordered activities performed by a student during one

[1]The Dataset1 was collected from undergraduate students of a Pakistani university who solved code-tracing problems, while, the Dataset2 was collected from students studying in an Italian university who solved multiple choice questions.

[2]Markov predicted models are constructed using the *Markovian* assumption as described in Section 3.5.2.4 – paragraph C.

[3]That is, base models of both datasets constructed in Section 3.5.2.4 – paragraph D.

session. We assumed that a student's behavior may change from one Login-Logout session to another, and thus a session/trace is the lowest possible level to understand the behavioral dynamics of the students. To determine the validity of our approach, we compared engagement/disengagement behavioral groups at student and trace level. Student level contained the count of all activities performed by each student in different Login-Logout sessions. As expected, the next activity prediction accuracy was higher at trace level than that at student level.

Based on these findings and results, we conclude that students depict different problem-solving behaviors during confidence-based assessment that reflect their engagement or disengagement. These distinct behavioral patterns can be used to identify students showing unproductive behaviors leading to a decrease in their performance outcomes, poor confidence accuracy and/or disinterest in the assessment process. However, students behaviors may vary within and across different Login-Logout sessions, therefore, representing their trails of activities and predicting future behaviors accurately is challenging. This tells us it requires the development of sophisticated methods that can represent drift in students' behaviors at both levels (i.e. within and across different sessions). We are also optimistic that students' engagement behaviors leading to their confidence accuracy and knowledge outcomes can be improved through personalized feedback that promotes self-reflection. This is discussed along with other future work directions of this research in Section 4.5.

The research questions, objectives and outcomes of this thesis are summarized in Table 4.1.

4.3 Contributions

This research work contributes to elevate the existing state of educational research and educational data mining domains. To be precise, despite having several benefits over the traditional assessment approach, confidence-based

Table 4.1: Thesis summary

Research questions	**Research objectives**	**Outcomes**
RQ-1: What behaviors can be used to determine student engagement/disengagement in confidence-based assessment?	Objective I – Investigate the correlation of different student performance parameters with confidence-based assessment.	Identified new problem-solving parameters for student engagement detection in confidence-based assessment.
	Objective II – Define a scheme to classify students' activities into engagement or disengagement behaviors.	A new scheme was proposed to categorize students' activities into six engagement/disengagement behaviors.
RQ-2: How can we model these behaviors to construct students' behavioral profiles?	Objective III – Model, analyze and predict students varying engagement and disengagement behaviors using probabilistic models.	A novel approach was developed to estimate mixture Markov model for multivariate categorical data. We also build a classifier for mixture Markov models that assigns new input observation(s) to the best mixture component.

assessment has not gained popularity in a large community. It has been investigated so far by educational theorists and many case studies are available in the current literature, that enlighten the effectiveness of this two-dimensional assessment paradigm on students' knowledge retention [Adams and Ewen, 2009, Hunt, 2003], performance outcomes [Nietfeld et al., 2006] and engagement [Boekaerts and Rozendaal, 2010, Lang et al., 2015].

Our work contributes to extend its applications into computer-based assessment, which allows access to thousands of students at a time. In addition to that, this work introduces confidence-based assessment to a larger community of educational data mining researchers who aim to analyze students data recorded by computer-based learning/assessment systems to understand the diverse needs of the students. The capability of data mining and machine learning algorithms are fully utilized by these researchers to reveal interesting insights about students' learning outcomes and their intended behaviors and make future predictions, which are otherwise impossible.

In this work, we conducted three research studies to investigate students problem-solving behaviors during confidence-based assessment. In fact, it is the very first attempt to exploit students' logged interactions to categorize their behaviors in confidence-based assessment. The proposed scheme is a novel approach that defines six distinct categories representing students' engagement and disengagement behaviors [Maqsood et al., 2019] based on three important problem-solving parameters [Maqsood and Ceravolo, 2019]. This new set of engagement/disengagement behavioral patterns not only offers information about students' varying problem-solving behaviors but at the same time it represents their knowledge level and confidence accuracy. Furthermore, our findings show that the proposed scheme is associated with students' real performance in the class [Maqsood et al., 2019], which reveal the potential of this approach. We hope that this multifaceted engagement detection model opens new avenues of further investigation and implementation in computer-based assessment systems.

Another distinctive contribution of our work is in the field of The proposed method introduces a new approach for initializing the Expectation-Maximization algorithm, which plays a critical role in estimating model parameters of resultant mixture components. In this regard, several approaches have been proposed for the Gaussian Mixture Model that describe the probability distribution of continuous data. However, the current literature shows that very few attempts are made for categorical data and so we believe that our method is a useful addition to the existing collection. We also build a classifier for mixture Markov model that estimates the most suitable mixture component for new input observation sequences. None of the existing packages available for mixture Markov model in R have implemented the classifier. Our code implementation of the classifier in R is shared on GitHub[4] platform so that other researchers may benefit.

Some other researchers, for example Cadez et al. [2003], Hansen et al. [2017], Park et al. [2018] have also used mixture Markov models to model and interpret users' interactions with computer-based environments. But, none of these works have evaluated prediction accuracy of their models, which is a precondition for developing intelligent systems [Desmarais and Baker, 2012]. Additionally, our method is a modification to the original EM algorithm which resulted in distinct Markov chains having promising accuracy to predict a student's next (or future) activity behavior.

4.4 Discussion

This section discusses the main limitations of our study and proposes a set of recommendations that can guide future researches in the domain of educational data mining.

[4]https://github.com/r-maqsood/Mixture-Markov-Models-R

4.4.1 Limitations

There are some limitations of this research work and the obtained results, as discussed below.

- The first limitation is that the two real datasets used in this work were of small to medium sizes with six discrete labels representing different engagement/disengagement behaviors (details of the two datasets and experimental studies are given in Section 3.2). Additionally, the second dataset (Dataset2) was only available to us by the time of third research study, and so we could not validate the results of the first and second research study. The datasets we consider came from two contexts denoted by different general and teaching cultures, i.e. Pakistan and Italy. This gives support to the generality of our findings, according to the principles of cross-validation. It is however clear that full cross-validation of our results would require additional experimental confirms.

- Another limitation of our work is related to the classification scheme we proposed to categorize students' problem-solving activities into engaged or disengaged behaviors (see Section 3.4.1). The classification scheme only considers feedback-seeking or no-seeking behavior into account while differentiating between student engagement and disengagement. Our concern is that there is a possibility that a student just clicks on the feedback page for curiosity (or let's say by mistake), or do not spend sufficient time to read and process the presented information. We ignored feedback reading time, since, no significant difference was found in feedback reading times for problems solved with distinct confidence-outcome categories (see Section 3.3.3.2, paragraph E, for details). Although the generality of the proposed scheme is increased by not defining strict time limits (defined by human experts, e.g. as it is done in [Beal et al., 2006, Brown and Howard, 2014, Joseph, 2005]),

for reasons discussed in Section 3.4.3; we believe that it also introduces some noise in the data where we misinterpret the student(s) problem-solving behavior. An alternative approach is suggested in the next subsection.

- From our experience with the Dataset2, we observed that longer Login-Logout sessions length is a limiting factor in obtaining high prediction accuracy for the student's future behavior. As shown in Table 3.15, the maximum number of solved problems in the Dataset1 and Dataset2 were 6 and 39, respectively; and the average number of solved problems were respectively 5 and 17. The next activity prediction accuracy results of different obtained clusters for both datasets are given in Appendix A.2 (Table A.1 and A.2). The results with 90% train data show that the highest prediction accuracy achieved for both datasets, using our proposed K-EM method, was: 83.33% for the Dataset1; and, 58.54% for the Dataset2.

  In our opinion, these longer problem-solving sessions also hide some interesting but relatively less frequent behavioral patterns depicted by the students. The visualization of the four resultant Markov chains of the Dataset1 reveals very useful insights about the students' behaviors, see Fig 3.13. Specifically, first three resultant clusters captured engaged behaviors of the students revealing the following behavioral groups: (i) students who mostly gave correct answers with high confidence which show their 'mastery' in the subject domain, (ii) high confident students who have knowledge of the subject but mostly gave wrong answers and showed interest in fill-knowledge gap activity, and (iii) shows the smallest group of mixed engagement behaviors. While the fourth cluster showed disengagement behaviors of high confident students who gave wrong answers and mostly did not try to fill their knowledge gaps. However, in the case of the Dataset2, the two resultant clusters were mainly separated by the activities performed with

high or low confidence level (see Fig 3.14). And, this could be a potential reason for obtaining less prediction accuracy for the Dataset2.

4.4.2 Recommendations

In this section, we list down some recommendations for researchers interested in carrying out a similar study.

- As mentioned earlier, categorizing students' problem-solving behaviors based on different time conditions (identified by human experts) restrict the application of the developed scheme to other domains. And, completely ignoring the time limits can introduce noise in the data. A compromise could be to define a minimum threshold value for the time required to perform some activity. For example, in our work, feedback-seeking/no-seeking behavior plays an important role in differentiating between students' engaged or disengaged behaviors. Given that students could request corrective feedback to fill their knowledge gap(s) which determines their intention or involvement in the assessment process. We can assume that it requires at least 10 seconds to read the content presented on the feedback page, and therefore, students who spent less time than this threshold value did not intend to open the page or read the feedback completely. Thus, it is not appropriate to relate their behavior to some engaged class (associated with high and low confidence), rather a new class called 'curious' may be introduced. We hope that with this new class, a clear performance difference can be seen in high and low performing students in relation to different problem-solving behavioral groups; which were somehow not very significant in our results (as discussed in Section 3.4.2.3).

- We urge other researchers to carry out multidisciplinary and cross-cultural experimental studies to implement educational frameworks and evaluate their generalizability through data exploration, which is

one of the growing concerns for educational researchers [Jensen et al., 2019]. The results that we achieved from experimenting with Pakistani and Italian students are very encouraging. The experiments were designed for relatively different subjects and different type of questions (see details in Section 3.2), but the underlying assessment model was the same (given in Fig. 3.1). Interestingly, we found that some common behaviors were depicted by the students from both populations, as already discussed in Section 4.2. We believe that if more evidence(s) can be obtained about the generality of some behaviors, it would be possible to develop a model that comprises of common behavioral states and the same model can be used without hesitation for new students (from different populations) to classify and construct their behavioral profiles. The findings of a recent work by Jensen et al. [2019] are encouraging and provide the evidence for constructing generalized models using students' interactions data. Furthermore, Cocea and Weibelzahl [2011] and Tan et al. [2014] have also shown empirically that students' engagement detection model can be compared across different domains.

- As mentioned in the limitations, we suggest limiting the maximum number of problems to be solved in a single Login-Logout session between 10 to 15. So, the students' varied behaviors do not get absorbed into more frequent and general behaviors, which also degrades the performance of the predictive model (see Section 4.4.1). Furthermore, students are more likely to become disengaged with longer (tutoring) sessions [Arroyo et al., 2007]. On the other hand, in our experience, short length problem-solving sessions challenge both the model training or learning capacity of the adopted technique and the validity of the obtained results. The least prediction accuracy obtained by the smaller dataset used in our work (i.e. Dataset1 with 5 problems solved on average) was 47.83%, whereas the other dataset (i.e. Dataset2 with 17

problems solved on average) achieved 51.12% accuracy at minimum[5].

4.5 Future work

One absolute future work direction is to implement the developed method of modeling students' behavioral dynamics in a real-time environment – that is, in an 'intelligent' assessment tool which classifies students' behaviors as they interact with the tool during assessment. In this work, we carried out multi-purpose research to identify crucial parameters reflecting students' engagement/disengagement and constructed their behavioral profiles. However, due to time restrictions, we could not implement and test our methodology in a real-time system. We also suggest supporting visualization of students approximate behaviors during assessment through Markov chains in the intelligent tool. The visualizations will allow class teachers to understand the strengths and weaknesses of their students at student and class level.

Another factor for restraining us to put our methodology into action was the development of personalized feedback that should be offered to the students depicting varying engagement/disengagement behaviors during assessment, which itself is a complete research topic. Therefore, designing and evaluating personalized feedback for different engagement/disengagement behaviors identified in our research for confidence-based assessment is a future challenge. We are optimistic that students' engagement behaviors leading to their confidence accuracy and knowledge outcomes can be improved through personalized feedback that promotes self-reflection. One such attempt is made by Hench [2014], who developed a graphical feedback mechanism that helped students to determine their confidence level for each question accurately by referring to the question's difficulty level and an associated degree of confidence inaccuracy by other students. Ar-

[5]The prediction accuracy results reported here were obtained using K-EM method with 90% train data, see Table A.1 and A.2 in Appendix A.2

royo et al. [2007] has also provided practical evidence to remediate students' disengaged behaviors using open-learner models in a mathematics tutoring system. But, instead of adopting a typical approach in which students are delivered personalized feedback from the set of pre-designed hard-coded feedback responses; we suggest constructing an incremental approach like the one proposed in [Höhn and Ras, 2016], that prepares the content of the feedback by considering various factors into account at run-time.

Although our proposed method, K-EM which estimates mixture Markov models for multivariate categorical data, achieved better prediction accuracy and convergence rate than the two traditional random initialization approaches (i.e. EM and emEM). There is still room for improving its prediction accuracy and validating the results on larger datasets. A naive approach to further improve the prediction accuracy is to use a higher order of Markov chains for representing clusters (that is, a mixture of higher-order Markov models), where, in a k-order Markov model the probability of a future state depends on k previous states. But, an increase in the accuracy would come with the cost of an increase in time and space complexity which is not favorable especially if the developed model is to be implemented in a real-time online setting (e.g., an intelligent/adaptive system). However, utilizing and comparing other machine learning methods for predicting students' future behavior can be an insightful future work.

To promote students' learning and to assist them during problem-solving, many of the computer-based learning/assessment systems also provide 'help' to students. The help can be offered in different ways, for example: as scaffolding questions, (bottom-out) hints, clues, etc. Proper utilization of the available help depends on different factors correlating to a student's intention during problem-solving (e.g., time spent per problem, amount and time spent seeing problems, etc.) [Arroyo et al., 2004]. Some researchers have considered positive and negative help-seeking behaviors to categorize students' problem-solving activities into engaged or disengaged behaviors,

e.g. [Beal et al., 2006, Pardos et al., 2014, Tan et al., 2014].

In this respect, another direction to extend our work is by including students' usage of 'check previous solution' activity in the engagement detection model. 'Check previous solution' activity is shown in the assessment model (Fig. 3.1) which was implemented in the two computer-based assessment tools used for data collection. The activity provides detailed answers to previously submitted questions by the student, which may help him/her to answer the current question. However, all students may not utilize this available feature and so it can be useful to further differentiate between engagement behaviors of high versus low performing students.

A

APPENDICES

A.1 Appendix A

As mentioned in Section 3.2.2, we used two different computer-based assessment tools namely, CodeMem and QuizConf, for performing two experimental studies. Details of both experimental studies are given in Section 3.2. Both tools recorded students' interactions during confidence-based assessment in which they were asked to specify confidence level (as high or low) before submitting a solution. The assessment model implemented in both tools, that define the navigational structure between different activities a student may perform and used for data collection, is shown in Fig 3.1. The data collected from CodeMem and QuizConf tools were named as Dataset1 and Dataset2, respectively. A sample of collected raw data is already shown in Fig. 3.2.

CodeMem which was used in our first experimental study, is a pre-existing tool developed by a team of three students[1] from National University of Computer and Emerging Sciences, Pakistan. The tool was developed for evaluating code tracing skills of students learning C/C++. More specifically, for a given code snippet, students are required to fill a trace

[1]Under the supervision of the principal investigator of this research study. The tool was modified to incorporate objectives of the experimental work.

table showing the correct order of executable line(s) of code and updated value of variables and/or expressions contained in each particular line of code. Fig. A.1 shows screenshot of a sample question given to a student in CodeMem tool.

QuizConf – built by the principal investigator of this research work – was used in the second experimental study. The tool was designed to facilitate students for assessing their problem-solving skills in an introductory programming course. The tool asks multiple choice questions from students for a given code flow diagram, displayed on the same page. Fig. A.2 shows screenshot of a sample question given to a student in QuizConf tool.

As shown in the screenshots, student's confidence level associated with each answered question was obtained using separate submit buttons for 'high' and 'low' confidence. This was done so that students make a conscious choice of their confidence level before submitting an answer. Furthermore, students were also allowed to 'quit' a problem (using Quit button) which can be done any number of times before submitting a final answer for any question.

As mentioned in Section 3.3.2, both tools offered two types of feedback to the students: (i) an implicit 'knowledge of result' feedback which informs a student immediately of his/her response's correctness (i.e. either correct or incorrect); and, (ii) 'corrective feedback' that provides correct solution along with detailed feedback for a recently submitted problem, which was available on a student's explicit request. In Fig. A.3 and A.4, we provide screenshots of knowledge of result feedback pages from CodeMem and QuizConf tools, respectively. After receiving this implicit feedback for each submitted answer, a student may or may not request corresponding corrective feedback for a particular question. And, a student's feedback seeking/no-seeking behavior towards the corrective feedback has played a central role in our engagement/disengagement classification scheme based on his/her response's correctness and the confidence level (see Section 3.4.1 for details).

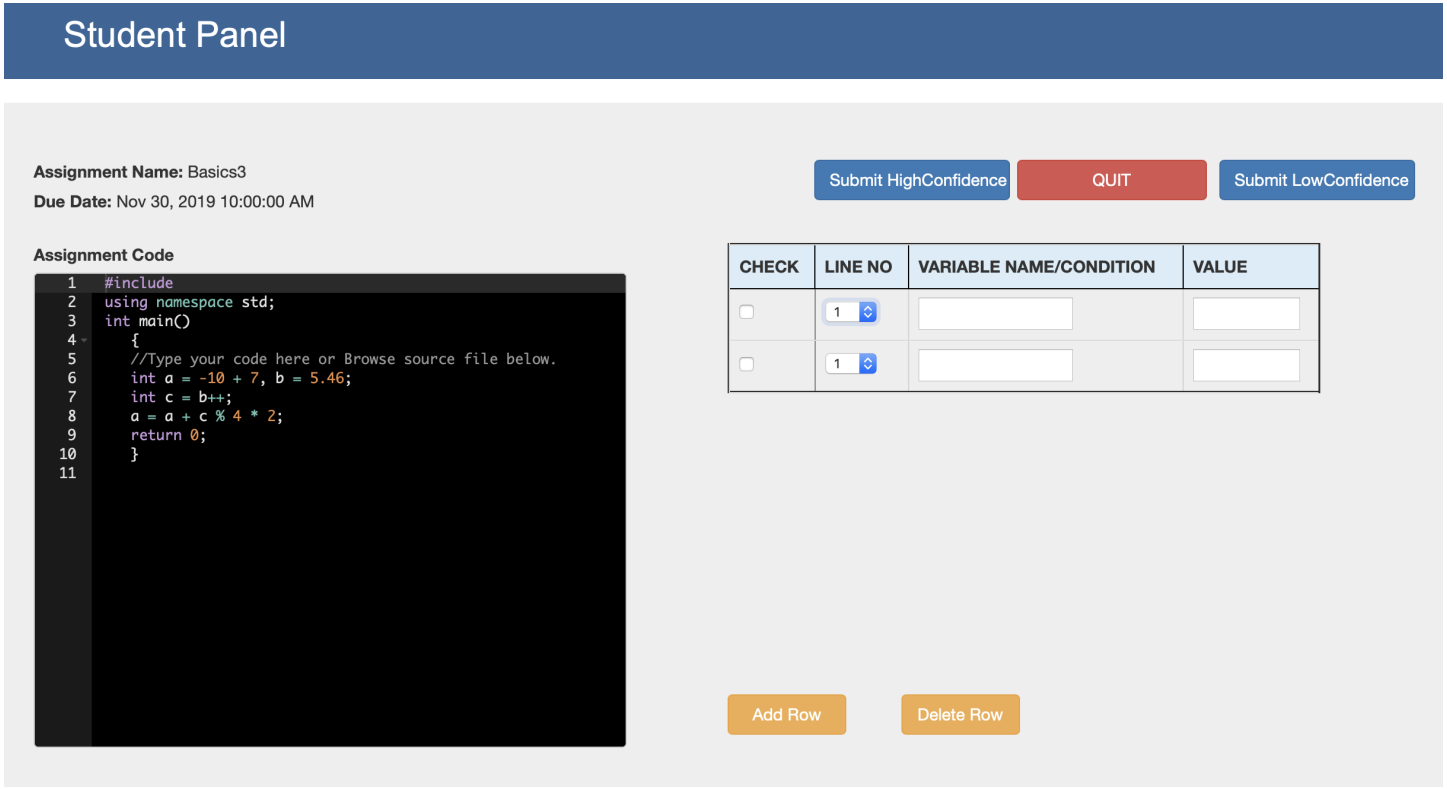


Figure A.1: A sample question from CodeMem tool

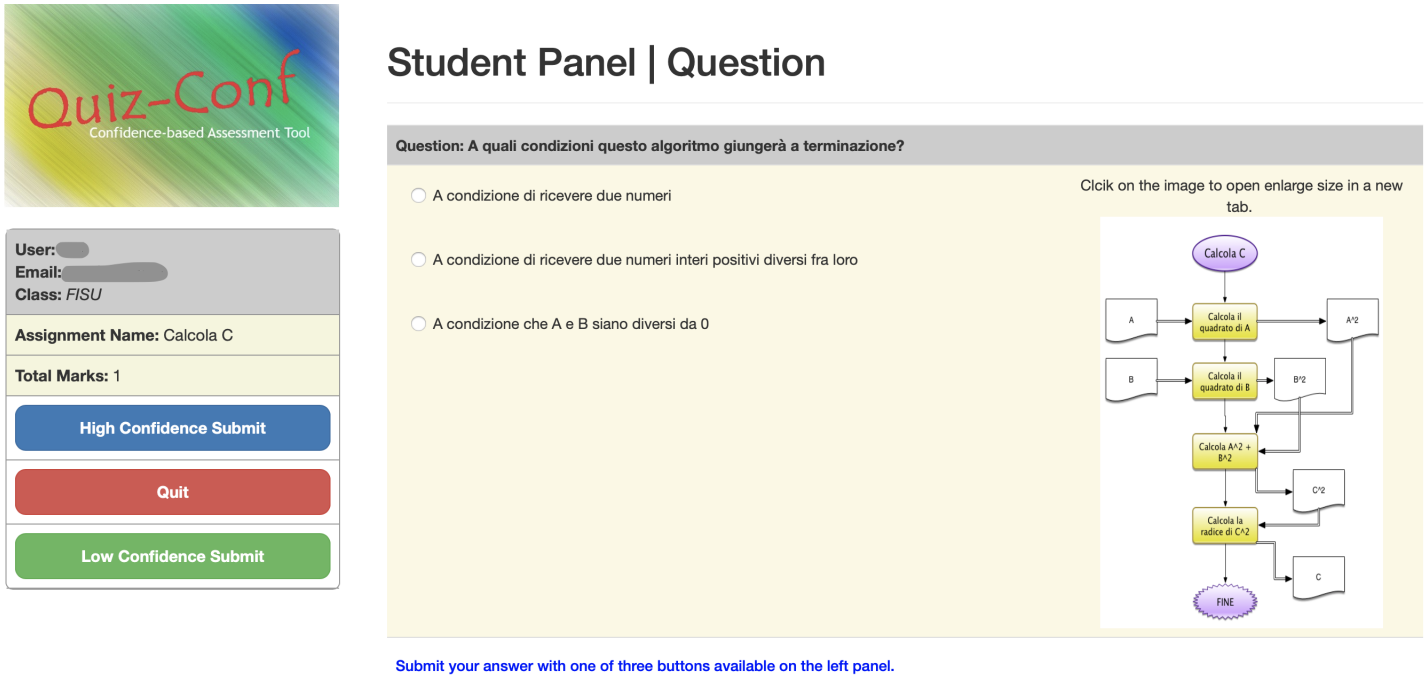


Figure A.2: A sample question from QuizConf tool

An example of corrective feedback offered in both CodeMem and QuizConf tools are respectively shown in Fig. 3.3 and 3.4.

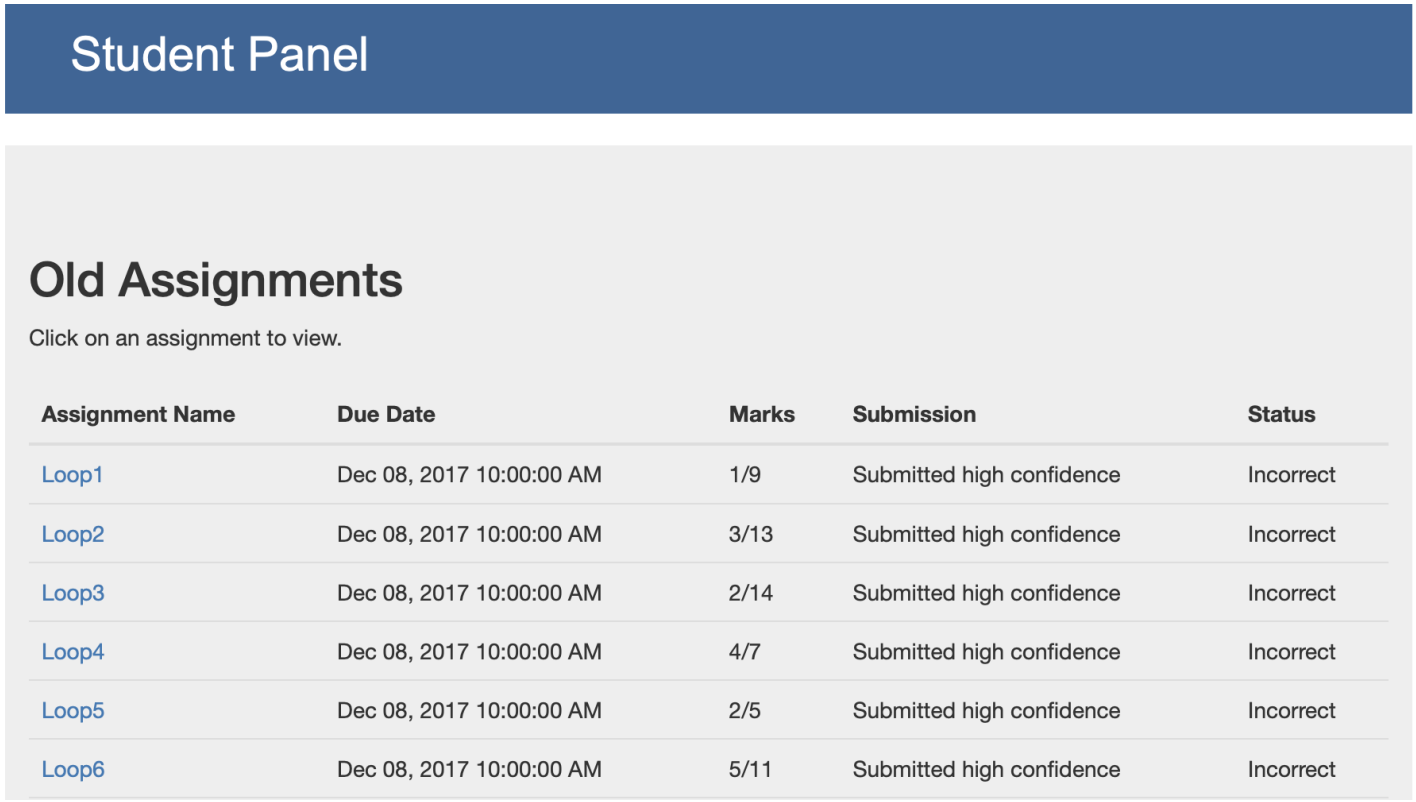


Figure A.3: An example knowledge of result feedback page from CodeMem tool

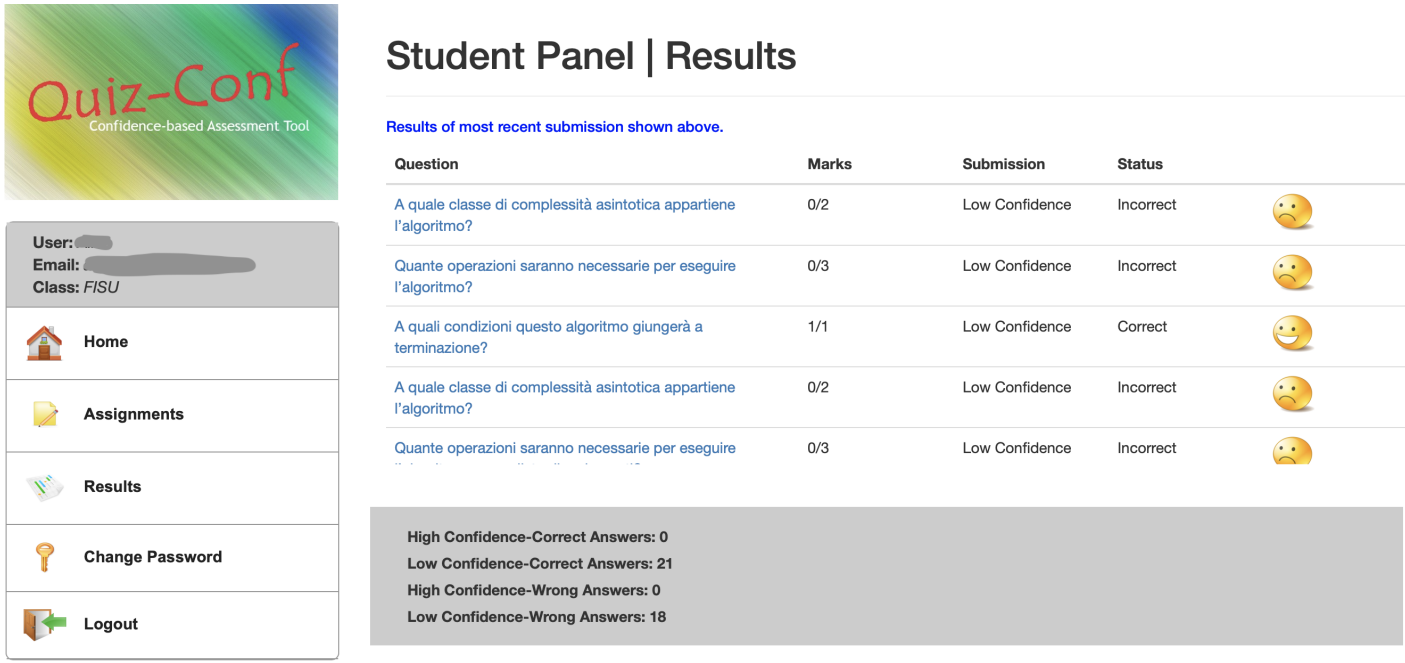


Figure A.4: An example knowledge of result feedback page from QuizConf tool

A.2 Appendix B

Here we provide detailed results of the model-based clustering performed in research study III (as described in Section 3.5.2). The results presented below show a comparison of prediction accuracy of the three clustering algorithms: EM, emEM and our proposed K-EM method, performed on the two datasets (Dataset1 and Dataset2). The next activity prediction accuracy of each obtained cluster is computed using Eq. (3.4). Table A.1 and A.2 contain prediction accuracy of the clusters obtained using optimal K for respective datasets, as mentioned in Section 3.5.2.5. Table A.3 and A.4 show prediction accuracy of the clusters obtained using the same number of K as that of the K-EM algorithm for respective datasets.

Table A.1: Comparison of test data prediction accuracy of different clusters obtained using the three algorithms for Dataset1 *(models constructed as described in Section 3.5.2.5; EM and emEM are run with K=2; K-EM with K=4 mixtures)*

Algorithm	**Dataset1** (197 traces) Base model prediction accuracy = 40.76%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	Cluster 1 (8 traces) = 61.29% Cluster 2 (11 traces) = 58.54%	Cluster 1 (16 traces) = 61.9% Cluster 2 (13 traces) = 56.14%	Cluster 1 (21 traces) = 57.14% Cluster 2 (18 traces) = 58.21%
emEM	Cluster 1 (9 traces) = 57.89% Cluster 2 (10 traces) = 58.82%	Cluster 1 (15 traces) = 66.67% Cluster 2 (14 traces) = 60%	Cluster 1 (10 traces) = 56.25% Cluster 2 (29 traces) = 54.76%
K-EM	Cluster 1 (6 traces) = 47.83% Cluster 2 (5 traces) = 83.33% Cluster 3 (2 traces) = 60% Cluster 4 (6 traces) = 66.67%	Cluster 1 (15 traces) = 67.19% Cluster 2 (2 traces) = 66.67% Cluster 3 (7 traces) = 66.67% Cluster 4 (5 traces) = 82.35%	Cluster 1 (21 traces) = 64.77% Cluster 2 (2 traces) = 66.67% Cluster 3 (9 traces) = 67.44% Cluster 4 (7 traces) = 80.95%

Table A.2: Comparison of test data prediction accuracy of different clusters obtained using the three algorithms for Dataset2 *(models constructed as described in Section 3.5.2.5; EM and emEM are run with K=3; K-EM with K=2 mixtures)*

Algorithm	**Dataset2** (348 traces) Base model prediction accuracy = 35.43%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	Cluster 1 (11 traces) = 58.74% Cluster 2 (10 traces) = 29.13% Cluster 3 (13 traces) = 62.6%	Cluster 1 (20 traces) = 64.15% Cluster 2 (18 traces) = 29.82% Cluster 3 (14 traces) = 58.25%	Cluster 1 (15 traces) = 73.86% Cluster 2 (26 traces) = 47.17% Cluster 3 (28 traces) = 55.18%
emEM	Cluster 1 (8 traces) = 34.29% Cluster 2 (12 traces) = 56.69% Cluster 3 (14 traces) = 59.31%	Cluster 1 (17 traces) = 60.94% Cluster 2 (12 traces) = 40.34% Cluster 3 (23 traces) = 55.41%	Cluster 1 (15 traces) = 38.55% Cluster 2 (27 traces) = 61.25% Cluster 3 (27 traces) = 58.28%
K-EM	Cluster 1 (17 traces) = 58.54% Cluster 2 (17 traces) = 51.12%	Cluster 1 (29 traces) = 59.03% Cluster 2 (23 traces) = 47.58%	Cluster 1 (35 traces) = 59.45% Cluster 2 (34 traces) = 49.21%

(models constructed using the same number of clusters as of K-EM, i.e. K=4 mixtures, for all algorithms)

Algorithm	**Dataset1** (197 traces) Base model prediction accuracy = 40.76%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	Cluster 1 (4 traces) = 83.33% Cluster 2 (7 traces) = 60.71% Cluster 3 (3 traces) = 50% Cluster 4 (5 traces) = 55.56%	Cluster 1 (15 traces) = 66.18% Cluster 2 (empty) = N/A Cluster 3 (10 traces) = 63.89% Cluster 4 (4 traces) = 12.5%	Cluster 1 (4 traces) = 37.5% Cluster 2 (14 traces) = 52.94% Cluster 3 (2 traces) = 50% Cluster 4 (19 traces) = 61.18%
emEM	Cluster 1 (5 traces) = 55% Cluster 2 (9 trace) = 68.57% Cluster 3 (1 traces) = 60% Cluster 4 (4 traces) = 83.33%	Cluster 1 (10 traces) = 61.11% Cluster 2 (15 traces) = 58.46% Cluster 3 (4 traces) = 84.21% Cluster 4 (0 traces) = –	Cluster 1 (11 traces) = 57.5% Cluster 2 (2 traces) = 66.67% Cluster 3 (1 traces) = 50% Cluster 4 (25 traces) = 58.33%
K-EM	Cluster 1 (6 traces) = 47.83% Cluster 2 (5 traces) = 83.33% Cluster 3 (2 traces) = 60% Cluster 4 (6 traces) = 66.67%	Cluster 1 (15 traces) = 67.19% Cluster 2 (2 traces) = 66.67% Cluster 3 (7 traces) = 66.67% Cluster 4 (5 traces) = 82.35%	Cluster 1 (21 traces) = 64.77% Cluster 2 (2 traces) = 66.67% Cluster 3 (9 traces) = 67.44% Cluster 4 (7 traces) = 80.95%

Table A.4: Comparison of test data prediction accuracy of different clusters obtained using the three algorithms for Dataset2 *(models constructed using the same number of clusters as of K-EM, i.e. K=2 mixtures, for all algorithms)*

Algorithm	**Dataset2** (348 traces) Base model prediction accuracy = 35.43%		
	Train-Test ratio: 90-10	Train-Test ratio: 85-15	Train-Test ratio: 80-20
EM	Cluster 1 (17 traces) = 52.53% Cluster 2 (17 traces) = 57.38%	Cluster 1 (29 traces) = 57.4% Cluster 2 (23 traces) = 50.96%	Cluster 1 (29 traces) = 53.56% Cluster 2 (40 traces) = 56.47%
emEM	Cluster 1 (19 traces) = 62.85% Cluster 2 (15 traces) = 40.82%	Cluster 1 (24 traces) = 63.13% Cluster 2 (28 traces) = 41.2%	Cluster 1 (33 traces) = 64.17% Cluster 2 (36 traces) = 39.41%
K-EM	Cluster 1 (17 traces) = 58.54% Cluster 2 (17 traces) = 51.12%	Cluster 1 (29 traces) = 59.03% Cluster 2 (23 traces) = 47.58%	Cluster 1 (35 traces) = 59.45% Cluster 2 (34 traces) = 49.21%

BIBLIOGRAPHY

Timothy M Adams and Gary W Ewen. The importance of confidence in improving educational outcomes. In *25th annual conference on Distance Learning and Teaching*, 2009.

Hirotogu Akaike. Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike*, pages 199–213. Springer, 1998.

Ivon Arroyo, Tom Murray, Beverly P Woolf, and Carole Beal. Inferring unobservable learning variables from students' help seeking behavior. In *International Conference on Intelligent Tutoring Systems*, pages 782–784. Springer, 2004.

Ivon Arroyo, Kimberly Ferguson, Jeffrey Johns, Toby Dragon, Hasmik Meheranian, Don Fisher, Andrew Barto, Sridhar Mahadevan, and Beverly Park Woolf. Repairing disengagement with non-invasive interventions. In *In proceedings of the 13th International Conference on Artificial Intelligence in Education (AIED), R. Luckin et al.(Eds), Marina del Rey, July 2007*, volume 2007, pages 195–202, 2007.

Ryan SJd Baker and Lisa M Rossi. Assessing the disengaged behaviors of

learners. *Design recommendations for intelligent tutoring systems*, 1:153, 2013.

Albert Bandura. Self-efficacy: toward a unifying theory of behavioral change. *Psychological review*, 84(2):191, 1977.

Robert L Bangert-Drowns and Curtis Pyke. A taxonomy of student engagement with educational software: An exploration of literate thinking with electronic text. *Journal of Educational computing research*, 24(3): 213–234, 2001.

Carole Beal, Sinjini Mitra, and Paul Cohen. Modeling learning patterns of students with a tutoring system using hidden markov model. In *In proceedings of the 13th International Conference on Artificial Intelligence in Education (AIED), R. Luckin et al.(Eds), Marina del Rey, July 2007*, 2007.

Carole R Beal, Lei Qu, and Hyokyeong Lee. Classifying learner engagement through integration of multiple data sources. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, page 151. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

Christophe Biernacki, Gilles Celeux, and Gérard Govaert. Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. *Computational Statistics & Data Analysis*, 41(3-4):561–575, 2003.

Jeff A Bilmes et al. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510):126, 1998.

Monique Boekaerts and Jeroen S Rozendaal. Using multiple calibration indices in order to capture the complex picture of what affects students' accuracy of feeling of confidence. *Learning and Instruction*, 20(5):372–382, 2010.

Alejandro Bogarín, Rebeca Cerezo, and Cristóbal Romero. A survey on educational process mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(1):e1230, 2018.

Mina Shirvani Boroujeni and Pierre Dillenbourg. Discovery and temporal analysis of latent study patterns in mooc interaction sequences. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*, pages 206–215. ACM, 2018.

FRANÇOIS Bouchet, Jason M Harley, Gregory J Trevors, and Roger Azevedo. Clustering and profiling students according to their interactions with an intelligent tutoring system fostering self-regulated learning. *JEDM— Journal of Educational Data Mining*, 5(1):104–146, 2013.

Patrice Bouvier, Karim Sehaba, and Élise Lavoué. A trace-based approach to identifying users' engagement and qualifying their engaged-behaviours in interactive systems: application to a social game. *User Modeling and User-Adapted Interaction*, 24(5):413–451, 2014.

LaVonda Brown and Ayanna M Howard. A real-time model to assess student engagement during interaction with intelligent educational agents. Georgia Institute of Technology, 2014.

James Bruno. Information reference testing (irt) in corporate and technical training programs. *UCLA*, 1995.

James Bruno, Charles Smith, Patrick Engstrom, Timothy Adams, Kevin Warr, Michael Cushman, Brian Webster, Frederick Bollin, et al. Method and system for knowledge assessment using confidence-based measurement, February 9 2006. US Patent App. 11/187,606.

Karina M Burns, Nicholas R Burns, and Lynn Ward. Confidence—more a personality or ability trait? it depends on how it is measured: A comparison of young and older adults. *Frontiers in psychology*, 7:518, 2016.

Igor Cadez, David Heckerman, Christopher Meek, Padhraic Smyth, and Steven White. Model-based clustering and visualization of navigation patterns on a web site. *Data mining and knowledge discovery*, 7(4):399–424, 2003.

Elaine Chapman. Alternative approaches to assessing student engagement rates. *Practical Assessment*, 8(13):1–7, 2003.

Malika Charrad, Nadia Ghazzali, Véronique Boiteau, and Azam Niknafs. Nbclust package: finding the relevant number of clusters in a dataset. *UseR! 2012*, 2012.

Mihaela Cocea and Stephan Weibelzahl. Can log files analysis estimate learners' level of motivation? In *LWA*, pages 32–35. University of Hildesheim, Institute of Computer Science, 2006.

Mihaela Cocea and Stephan Weibelzahl. Cross-system validation of engagement prediction from log files. In *European Conference on Technology Enhanced Learning*, pages 14–25. Springer, 2007.

Mihaela Cocea and Stephan Weibelzahl. Log file analysis for disengagement detection in e-learning environments. *User Modeling and User-Adapted Interaction*, 19(4):341–385, 2009.

Mihaela Cocea and Stephan Weibelzahl. Disengagement detection in online learning: Validation studies and perspectives. *IEEE Transactions on Learning Technologies*, 4(2):114–124, 2011. doi: 10.1109/tlt.2010.14.

Paul R Cohen and Carole R Beal. Temporal data mining for educational applications. *Int. J. Software and Informatics*, 3(1):31–46, 2009.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

Michel C Desmarais and Ryan S Baker. A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction*, 22(1-2):9–38, 2012.

John J Dziak, Donna L Coffman, Stephanie T Lanza, Runze Li, and Lars Sommer Jermiin. Sensitivity and specificity of information criteria. *bioRxiv*, page 449751, 2019.

Apple WP Fok, Hau-San Wong, and YS Chen. Hidden markov model based characterization of content access patterns in an e-learning environment. In *2005 IEEE International Conference on Multimedia and Expo*, pages 201–204. IEEE, 2005.

Philippe Fournier-Viger, Jerry Chun-Wei Lin, Rage Uday Kiran, Yun Sing Koh, and Rincy Thomas. A survey of sequential pattern mining. *Data Science and Pattern Recognition*, 1(1):54–77, 2017.

Chris Fraley and Adrian E Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *The computer journal*, 41(8):578–588, 1998.

Rita Francese, Ignazio Passero, Giuseppe Scanniello, and Genoveffa Tortora. Improving student's self-efficacy using an adaptive approach. In *The international workshop on distance education technologies (DET 2007) of the 13th international conference on distributed multimedia system*, pages 149–154, 2007.

Jennifer A Fredricks, Phyllis C Blumenfeld, and Alison H Paris. School engagement: Potential of the concept, state of the evidence. *Review of educational research*, 74(1):59–109, 2004.

Anthony R Gardner-Medwin. Confidence-based marking: encouraging rigour through assessment. In *Journal of Physiology*, volume 567, page WA10, 2005.

Anthony R Gardner-Medwin and Mike Gahan. Formative and summative confidence-based assessment. 2003.

Jiří Grim. Em cluster analysis for categorical data. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 640–648. Springer, 2006.

Charles Miller Grinstead and James Laurie Snell. *Markov Chains*, chapter 11, pages 405–450. American Mathematical Soc., 2012.

Julio Guerra, Shaghayegh Sahebi, Yu-Ru Lin, and Peter Brusilovsky. The problem solving genome: Analyzing sequential patterns of student work with parameterized exercises. In *Educational Data Mining 2014*, 2014.

Maya R Gupta, Yihua Chen, et al. Theory and use of the em algorithm. *Foundations and Trends® in Signal Processing*, 4(3):223–296, 2011.

Siti Suhaila Abdul Hamid, Novia Admodisastro, Noridayu Manshor, Azrina Kamaruddin, and Abdul Azim Abd Ghani. Dyslexia adaptive learning model: Student engagement prediction using machine learning approach. In *International Conference on Soft Computing and Data Mining*, pages 372–384. Springer, 2018. doi: 10.1007/978-3-319-72550-5_36.

Christian Hansen, Casper Hansen, Niklas Hjuler, Stephen Alstrup, and Christina Lioma. Sequence modelling for analysing student interaction with educational systems. In *Proceedings of the 10th International Conference on Educational Data Mining (2017)*, pages 232–237, 2017.

John Hattie and Helen Timperley. The power of feedback. *Review of educational research*, 77(1):81–112, 2007. doi: 10.3102/003465430298487.

Thomas L Hench. Using confidence as feedback in multi-sized learning environments. In *International Computer Assisted Assessment Conference*, pages 88–99. Springer, 2014.

Arnon Hershkovitz and Rafi Nachmias. Learning about online learning processes and students' motivation through web usage mining. *Interdisciplinary Journal of E-Learning and Learning Objects*, 5(1):197–214, 2009.

Sviatlana Höhn and Eric Ras. Designing formative and adaptive feedback using incremental user models. In *International Conference on Web-Based Learning*, pages 172–177. Springer, 2016.

Zhengyu Hu. *Initializing the EM algorithm for data clustering and subpopulation detection*. PhD thesis, The Ohio State University, 2015.

Zhexue Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery*, 2(3):283–304, 1998.

Darwin P Hunt. The concept of knowledge and how to measure it. *Journal of intellectual capital*, 4(1):100–113, 2003. doi: 10.1108/14691930310455414.

Darwin P Hunt and H Furustig. Being informed, being misinformed and disinformation: a human learning and decision making approach. *Technical Report PM*, 56:238, 1989.

Emily Jensen, Stephen Hutt, and Sidney K D'Mello. Generalizability of sensor-free affect detection models in a longitudinal dataset of tens of thousands of students. In *Proceedings of the 12th International Conference on Educational Data Mining (2019)*, pages 324–329, 2019.

E Joseph. Engagement tracing: using response times to model student disengagement. *Artificial intelligence in education: Supporting learning through intelligent and socially informed technology*, 125:88, 2005.

Geetha Kanaparan. *Self-efficacy and engagement as predictors of student programming performance: An international perspective*. PhD thesis, Victoria University of Wellington, New Zealand, 2016.

Faten Khalil, Hua Wang, and Jiuyong Li. Integrating markov model with clustering for predicting web page accesses. In *Proceeding of the 13th Australasian World Wide Web Conference (AusWeb07)*, pages 63–74. AusWeb, 2007.

Mirjam Köck and Alexandros Paramythis. Activity sequence modelling and dynamic clustering for personalized e-learning. *User Modeling and User-Adapted Interaction*, 21(1-2):51–97, 2011.

Raymond W Kulhavy and William A Stock. Feedback in written instruction: The place of response certitude. *Educational Psychology Review*, 1(4): 279–308, 1989. doi: 10.1007/bf01320096.

Andju Sara Labuhn, Barry J Zimmerman, and Marcus Hasselhorn. Enhancing students' self-regulation and mathematics performance: The influence of feedback and self-evaluative standards. *Metacognition and Learning*, 5 (2):173–194, 2010.

Charles Lang, Neil Heffernan, Korinn Ostrow, and Yutao Wang. The impact of incorporating student confidence items into an intelligent tutor: A randomized controlled trial. *International Educational Data Mining Society*, 2015.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

Jay Magidson and Jeroen Vermunt. Latent class models for clustering: A comparison with k-means. *Canadian Journal of Marketing Research*, 20 (1):36–43, 2002.

Rabia Maqsood and Paolo Ceravolo. Modeling behavioral dynamics in confidence-based assessment. In *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)*, pages 452–454. IEEE, 2018.

Rabia Maqsood and Paolo Ceravolo. Corrective feedback and its implications on students' confidence-based assessment. In *Technology Enhanced Assessment 2018 - Communications in Computer and Information Science (CCIS)*, pages 55–72. Springer, 2019.

Rabia Maqsood, Paolo Ceravolo, and Sebastián Ventura. Discovering students' engagement behaviors in confidence-based assessment. In *2019 IEEE Global Engineering Education Conference (EDUCON)*, pages 841–846. IEEE, 2019.

James H McMillan and Jessica Hearn. Student self-assessment: The key to stronger student motivation and higher achievement. *Educational Horizons*, 87(1):40–49, 2008.

Marina Meilă and David Heckerman. An experimental comparison of model-based clustering methods. *Machine learning*, 42(1-2):9–29, 2001.

Volodymyr Melnykov. Clickclust: An r package for model-based clustering of categorical sequences. *Journal of Statistical Software*, 74(i09), 2016.

Volodymyr Melnykov, Ranjan Maitra, et al. Finite mixture models and model-based clustering. *Statistics Surveys*, 4:80–116, 2010.

Semhar Michael and Volodymyr Melnykov. An effective strategy for initializing the em algorithm in finite mixture models. *Advances in Data Analysis and Classification*, 10(4):563–583, 2016.

Edna H Mory. Adaptive feedback in computer-based instruction: Effects of response certitude on performance, feedback-study time, and efficiency. *Journal of Educational Computing Research*, 11(3):263–290, 1994. doi: 10.2190/ym7u-g8un-8u5h-hd8n.

Edna H Mory. Feedback research revisited. *Handbook of research on educational communications and technology*, 2:745–783, 2004.

Kasia Muldner, Winslow Burleson, Brett Van de Sande, and Kurt VanLehn. An analysis of students' gaming behaviors in an intelligent tutoring system: predictors and impacts. *User Modeling and User-Adapted Interaction*, 21(1-2):99–135, 2011.

Privault Nicolas. *Discrete-Time Markov Chains*, chapter 4, pages 89–113. Springer Undergraduate Mathematics Series, 2013. doi: https://doi.org/10.1007/978-981-13-0659-4_4.

John L Nietfeld, Li Cao, and Jason W Osborne. The effect of distributed monitoring exercises and feedback on performance, monitoring accuracy, and self-efficacy. *Metacognition and Learning*, 1(2):159, 2006.

Paul Novacek. Confidence-based assessments within an adult learning environment. *International Association for Development of the Information Society*, 2013.

Christoph Pamminger, Sylvia Frühwirth-Schnatter, et al. Model-based clustering of categorical time series. *Bayesian Analysis*, 5(2):345–368, 2010.

Zach A Pardos, Ryan SJD Baker, Maria San Pedro, Sujith M Gowda, and Supreeth M Gowda. Affective states and state tests: investigating how affect and engagement during the school year predict end-of-year learning outcomes. *Journal of Learning Analytics*, 1(1):107–128, 2014. doi: 10.18608/jla.2014.11.6.

Jihyun Park, Renzhe Yu, Fernando Rodriguez, Rachel Baker, Padhraic Smyth, and Mark Warschauer. Understanding student procrastination via mixture models. In *Proceedings of the 11th International Conference on Educational Data Mining (2018)*, 2018.

Mykola Pechenizkiy, Nikola Trcka, Ekaterina Vasilyeva, Wil van Aalst, and Paul De Bra. Process mining online assessment data. In *Educational Data Mining 2009*, 2009.

David W Petr. Measuring (and enhancing?) student confidence with confidence scores. In *Frontiers in Education Conference, 2000. FIE 2000. 30th Annual*, volume 1, pages T4B–1. IEEE, 2000.

Jiezhong Qiu, Jie Tang, Tracy Xiao Liu, Jie Gong, Chenhui Zhang, Qian Zhang, and Yufei Xue. Modeling and predicting learning behavior in moocs, 2016.

Cristóbal Romero, Sebastián Ventura, and Paul De Bra. Knowledge discovery with genetic programming for providing feedback to courseware authors. *User Modeling and User-Adapted Interaction*, 14(5):425–464, 2004.

Cristobal Romero, Sebastian Ventura, Mykola Pechenizkiy, and Ryan SJd Baker. *Handbook of educational data mining*. CRC press, 2010.

Teomara Rutherford. The measurement of calibration in real contexts. *Learning and Instruction*, 47:33–42, 2017.

Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

Isabella Seeber, Ronald Maier, Paolo Ceravolo, and Fulvio Frati. Tracing the development of ideas in distributed, it-supported teams during synchronous collaboration. In *Proceedings of the European Conference on Information Systems (ECIS)*, 2014.

David H Shanabrook, David G Cooper, Beverly Park Woolf, and Ivon Arroyo. Identifying high-level student behavior using sequence-based motif discovery. In *Educational Data Mining 2010*, 2010.

Valerie J Shute. Focus on formative feedback. *Review of educational research*, 78(1):153–189, 2008. doi: 10.3102/0034654307313795.

Lazar Stankov, Suzanne Morony, and Yim Ping Lee. Confidence: the best non-cognitive predictor of academic achievement? *Educational Psychology*, 34(1):9–28, 2014.

Lazar Stankov, Sabina Kleitman, and Simon A Jackson. Measures of the trait of confidence. In *Measures of personality and social psychological constructs*, pages 158–189. Elsevier, 2015.

Nancy J Stone. Exploring the relationship between calibration and self-regulated learning. *Educational Psychology Review*, 12(4):437–475, 2000.

Ling Tan, Xiaoxun Sun, and Siek Toon Khoo. Can engagement be compared? measuring academic engagement for comparison. In *EDM*, pages 213–216, 2014.

Behnam Taraghi, Anna Saranti, Martin Ebner, Vinzent Mueller, and Arndt Grossmann. Towards a learning-aware application guided by hierarchical classification of learner profiles. *J. UCS*, 21(1):93–109, 2015.

Mike Thelwall. Computer-based assessment: a versatile educational tool. *Computers & Education*, 34(1):37–49, 2000.

Caroline F Timmers, Jannie Braber-Van Den Broek, and StéPhanie M Van Den Berg. Motivational beliefs, student effort, and feedback behaviour in computer-based formative assessment. *Computers & education*, 60(1):25–31, 2013. doi: 10.1016/j.compedu.2012.07.007.

Anders Tolver. An introduction to markov chains. *Department of Mathematical Sciences, University of Copenhagen*, 2016.

Fabienne M Van der Kleij, Theo JHM Eggen, Caroline F Timmers, and Bernard P Veldkamp. Effects of feedback in a computer-based assessment for learning. *Computers & Education*, 58(1):263–272, 2012. doi: 10.1016/j.compedu.2011.07.020.

Fabienne M Van der Kleij, Remco CW Feskens, and Theo JHM Eggen. Effects of feedback in a computer-based learning environment on students' learning outcomes: A meta-analysis. *Review of educational research*, 85(4):475–511, 2015. doi: 10.3102/0034654314564881.

Ekaterina Vasilyeva, Mykola Pechenizkiy, and Paul De Bra. Tailoring of feedback in web-based learning: the role of response certitude in the assessment. In *Intelligent Tutoring Systems*, pages 771–773. Springer, 2008. doi: 10.1007/978-3-540-69132-7_104.

Krista Lee Vogt. *Measuring student engagement using learning management systems*. PhD thesis, University of Toronto, Canada, 2016.

Barry J Zimmerman. Attaining self-regulation: A social cognitive perspective. In *Handbook of self-regulation*, pages 13–39. Elsevier, 2000.

Barry J Zimmerman. Becoming a self-regulated learner: An overview. *Theory into practice*, 41(2):64–70, 2002.