THESIS ON INFORMATICS AND SYSTEM ENGINEERING C121

Competence Based Learning – Framework, Implementation, Analysis and Management of Learning Process

KADRI UMBLEJA

TTÜ
PRESS

TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Computer Systems

This dissertation was accepted for the defence of the degree of Doctor of Philosophy in Computer and Systems Engineering on 14.03.2017

Supervisor: Vello Kukk, Ph.D., Department of Computer Systems, Tallinn University of Technology, Tallinn, Estonia

Innar Liiv, Ph.D., Department of Software Science, Tallinn University of Technology, Tallinn, Estonia

Opponents: Jan M. Pawlowski, Ph.D., Institute of Computer Science, Ruhr West University of Applied Sciences, Mülheim an der Ruhr, Germany

Timo Tobias Ley, Ph.D., School of Educational Sciences, Tallinn University, Estonia.

Defence of the thesis: April 21, 2017, Tallinn

Declaration:
Hereby I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology has not been submitted for any academic degree.

European Union
European Social Fund
Investing in your future
DoRa
ARCHIMEDES

ISSN 1406-4731
ISBN 978-9949-83-088-6 (publication)
ISBN 978-9949-83-089-3 (PDF)

INFORMAATIKA JA SÜSTEEMITEHNIKA C121

Kompetentsipõhine õpe – raamistik, implementatsioon, analüüs ja õppeprotsessi juhtimine

KADRI UMBLEJA

TTÜ
KIRJASTUS

TABLE OF CONTENTS

LIST OF PUBLICATIONS

The work of this thesis is based on the following publications:

P1. K. Umbleja, V. Kukk, and M. Jaanus, "Answer evaluation in competence based learning," in ICEE 2011: An International Conference on Engineering Education, Belfast, UK, 2011.

P2. K. Umbleja, V. Kukk, M. Jaanus, A. Udal, and B. Gordon, "Analyzes of Competence Based Approach to Learning " in ICEE-2012, Turku, Finland, 2012, pp. 418–424.

P3. K. Umbleja, V. Kukk, and M. Jaanus, "Competence-based approach to learning," in Global Engineering Education Conference (EDUCON), 2013 IEEE, 2013, pp. 552-559.

P4. K. Umbleja, V. Kukk, M. Jaanus, and A. Udal, "New concepts of automatic answer evaluation in competence based learning," in 2014 IEEE Global Engineering Education Conference (EDUCON), 2014, pp. 922-925.

P5. K. Umbleja, "Students' grading control and visualisation in competence-based learning approach," in 2015 IEEE Global Engineering Education Conference (EDUCON), 2015, pp. 287-296.

P6. K. Umbleja and M. Ichino, "Predicting Students' Behavior During an E-Learning Course Using Data Mining," in Interactive Collaborative Learning: Proceedings of the 19th ICL Conference - Volume 2, Cham: Springer International Publishing, 2017, pp. 175-189.

OTHER RELATED PUBLICATIONS

O1. O. Shvets, V. Kukk, M. Jaanus, and K. Umbleja, "Home and remote laboratories - development of web based practical works for higher and vocational education," Journal of International Scientific Publications: Materials, Methods and Technologies, vol. 3, pp. 323-335, 2009.

O2. V. Kukk, M. Jaanus, K. Umbleja, and O. Shvets, "Developing integrated learning environment–analysis," in International Conference on Engineering Education, ICEE, 2010, pp. 18-22.

O3. O. Shvets, V. Kukk, M. Jaanus, and K. Umbleja, "Remote laboratories and methods of their application in the study process," Journal of International Scientific Publications: Materials, Methods and Technologies, vol. 4, pp. 60-67, 2010.

O4. M. Jaanus, V. Kukk, and K. Umbleja, "Integrating labs into learning environment," Elektronika ir Elektrotechnika, vol. 102, pp. 27-30, 2012.

O5. K. Umbleja, V. Kukk, and M. Jaanus, "Processing Answers in Competence-Based Learning," INNOVATIONS 2012: World Innovations in Engineering Educations and Research pp. 91-100, 2012.

O6. M. Jaanus, V. Kukk, K. Umbleja, B. Gordon, and M. Pikkov, "HomeLabKits -implementation and usage," in BEC 2012 : 13th biennial Baltic Electronics Conference, Tallinn, Estonia, 2012, pp. 331-334.

O7. V. Kukk and K. Umbleja, "Analysis of forgetting in a learning environment," in 2012 13th Biennial Baltic Electronics Conference, 2012, pp. 335-338.

O8. K. Umbleja, V. Kukk, M. Jaanus, and O. Shvets, "Laboratory Experience with Competence Based Learning," in ICEER 2013, Marrakesh, Morocco, 2013, pp. 487–494.

O9. M. Jaanus, A. Udal, V. Kukk, and K. Umbleja, "Using microcontrollers for high accuracy analogue measurements," Elektronika ir Elektrotechnika, vol. 19, pp. 51-54, 2013.

O10. K. Umbleja, M. Jaanus, V. Kukk, and A. Udal, "Interactive Distance Lab Solution for Robotic Arm," in ICEE/ICIT-2014 CONFERENCE PROCEEDINGS: ICEE/ICIT 14, Riga,Latvia, 2014.

O11. A. Udal, M. Jaanus, K. Umbleja, and R. Reeder, "The method of quantum optical communication based on entangled photon pairs," in 2014 14th Biennial Baltic Electronic Conference (BEC), 2014, pp. 33-36.

O12. M. Jaanus, A. Udal, V. Kukk, and K. Umbleja, "Implementation of the robot arm in the interactive learning environment," in Signals and Electronic Systems (ICSES), 2014 International Conference on, 2014, pp. 1-4.

O13. M. Jaanus, K. Umbleja, V. Kukk, and A. Udal, "Competence based interactive learning with HomeLabKits: Experience and work in progress," in 2014 IEEE Global Engineering Education Conference (EDUCON), 2014.
O14. K. Umbleja, "The first year experience of using LEGO Mindstorms robots in the Tallinn University of Technology outreach program for secondary and primary school learners," in New Technologies and Innovation for Global Business: ICEE 2015 International Conference of Engineering Education, Zagreb, Croatia, 2015, pp. 741-748.
O15. V. Kukk, K. Umbleja, and M. Jaanus, "Two-Dimensional Knowledge Model for Learning Control and Competence Mapping," in International Workshop on Learning Technology for Education in Cloud, 2015, pp. 16-27.
O16. M. Jaanus, A. Udal, V. Kukk, K. Umbleja, J. Gorbatsova, and L. Molder, "Improved C5D Electronic Realization of Conductivity Detector for Capillary Electrophoresis," Elektronika ir Elektrotechnika, vol. 22, pp. 29-32, 2016.
O17. K. Umbleja, "Can K-12 Students Learn How to Program with just Two Hours?," in International Workshop on Learning Technology for Education in Cloud, 2016, pp. 250-264.
O18. Ichino, M; Umbleja, K. (2016) Similarity and Dissimilarity Measures for Mixed Feature-Type Symbolic Data. International Series Studies in Theoretical and Applied Statistics, Springer International Publishing (in print)

AUTHOR'S CONTRIBUTION TO THE PUBLICATIONS

All the results in [P1]-[P4] were obtained by the author under the supervision of Dr. Vello Kukk. Results in [P6] are obtained by the author under the supervision of Dr. Innar Liiv and Dr. Manabu Ichino.

In [P1] author starts to develop automatic answer evaluation model that would be able to overcome drawbacks of then used answer evaluation method – limited feedback, repeated mistakes having same weight and multiply different mistakes, common typos being classified as mistakes etc. Author uses one set of tasks as examples and develops initial method that mimics student's answering process. It is then analysed by the author and it is found that the new proposed method overcomes the problems that the initial answer evaluation suffered from. Author also finds that switch to competence based method raises the number of tasks done by the students and their later attempts are remarkably higher than the first five ones and therefore justifies repetition as a part of the framework.

In [P2] the competence based learning framework developed by the author and Dr. Vello Kukk is analysed. Author covers usage of competences, results, usage of help materials, the effectiveness of algorithm proposed in [P1] and students' study behaviour in the analysis. Author finds that proposed method allows to extract more detailed information about the students that leads to more adequate control over learning. A coverage of competences on different ability levels is also covered. Author finds that coverage at that time is very poor with lack of tasks in lower levels. On the other hand, author finds that there has been improvements in results, algorithms and in students' study behaviours.

Competence based learning framework is described in detail by author in [P3]. Motivations why competence based learning was considered, is compiled by the author. The structure of courses and learning in competence based learning environment that was developed during the switch from topic based to competence based learning in collaboration by the author and Dr. Vello Kukk is described. The HomeLabKits were developed on the hardware side by Dr. Martin Jaanus. Automatic evaluation model is further developed by the author. Analysis of learning in the system is conducted by the author. In the end, author finds that motivations compiled before the switch were all achieved as showed by analysis. On the other hand, it is found by the author that the number of students who do not finish on time or drop out is still very high and competence based learning has a problem in form that students are not able to predict their time requirements properly.

In [P4] author further develops the model proposed in [P1] and [P3], to enable more detailed feedback, which is crucial for the learning process, and the analysis of the usage of new algorithms mimicking student's answering process. Furthermore, author presents a model of automatically synthesising evaluation algorithms using Rete algorithm. Author developed world rules for a small set of tasks in electrical engineering field for verification of usefulness of proposed model and found that the response time of automatic synthesis is slower than desired. Therefore, improved algorithm mimicking student's answering process remains in use in the e-learning system.

The problem of high dropouts and students' inability to predict their time needed to complete the course is considered by the author in [P5]. Grading in competence based system is described by the author and analysis is conducted by the author on how it has affected the results. Then author continues to describe five visualization tools that were developed to give students better overview of their process. First two visualization tools were developed by Dr. Vello Kukk during the process of switching from topic-based learning to competence based learning. Additional three visualisation tools were proposed and developed by the author. Feedback from students was collected by the author and analysed, which resulted in author discovering that students preferred familiar tools to new tools but ¾ of students agree that visualization tools have helped them to follow their progress. The data, on the other hand, does not show significant improvements in results.

In [P6] author continues to tackle the problem of high dropouts by compiling a list of features describing students' learning behaviour in the e-learning environment. PCA is used on a large set of features to find 9 most significant features to be used in development of student's study model. Then author applied general dis/similarity measure for histogram valued symbolic data, developed by the author to histogram valued descriptions of students to find clusters of similar students. Using clustering results, author developed student behaviour models for five major courses in e-learning environment. The model at one of the courses was then tested by the author to predict final grade and finishing time of new students taking the course. Results obtained by the author showed that developed models worked very well to predict final grade but did not yield similar result for predicting finishing time.

ABBREVIATIONS AND DEFINITIONS

Acquisition level	Level 77 from scale of 0 to 127 where it is assumed that the learner has acquired the competence
CBL	Competence Based Learning
Competence	Competence is defined as an elementary, atomic skill or knowledge that cannot be divided into smaller portion
Competence coverage	Competence coverage shows how many levels of possible difficulty levels for tasks have been covered with exercises for a specific competence.
Dendrogram	Dendrograms are graphical representations for the arrangement of the clusters produced by hierarchical clustering [1]
Difficulty level	Numeric value from scale 0-127 that is connected to the task and specific competence to reflect how advance is the usage of the competence in current task.
ECU	European Credit Unit
EDM	Educational Data Mining
HLK	HomeLabKit is portable kit containing all required equipment for doing lab experiments outside university laboratory.
Input competence	Competence that is always graded within the submission. Tasks can be called out only using input competences by the learning control
Knowledge	Knowledge is theoretical cognition acquired from learning materials.
OS	Operating System
PCA	Primary Component Analysis

Score/ability level	Numeric value from scale 0-127 that is connected to the learner and specific competence to reflect learner's proficiency with current competence.
SDA	Symbolic Data Analysis
Secondary competence	Competences that may be graded with the submission. Usually they are graded if specific mistake is made.
Skill	Skills are practical abilities that learner acquires as a result of learning process.
Summative grading	Commonly used grading method where multiply skills and knowledge are graded together with one mark.

1 INTRODUCTION

Education is the basis of success in every field. Technology offers many ways to improve and enhance education, learning and teaching, and it has been covered widely ([2-7] for example). The problem is that in order to apply benefits of technology to learning, it usually requires skills that majority of pedagogues do not have [8-10]. Therefore, it is the responsibility of computer scientists to step into the field of education and fill that cap to benefit everyone.

The first step in the evolution of learning is so called traditional classroom. Majority of higher education courses nowadays are still given by using this approach, when from week to week, the lecturer comes in front of the audience and delivers the material. The course ends with exams covering all material covered in the course. In some cases, course is supported with laboratories or practices and in some cases there are intermediate deadlines. The learning process rests mainly on the teacher and students must physically participate in the learning process [11]. Despite innovations in technology enabling alternative techniques for pedagogy, lecture formats continue to be the primary method for teaching adult learners [12]. Technology is just used to slightly enhance the passive learning method. For example, PowerPoint is used for showing slides.

The next step in evolution of learning is distance learning (sometimes called d-learning). Distance learning is defined as a method of studying in which lectures are broadcast or lessons are conducted by correspondence, without the student needing to attend a school or college [13]. For example, learning via television or radio is type of distance learning[14].

E-learning can be considered as a type of distance learning which focuses on building a cost-effective learning infrastructure, which in turn enables anytime, anywhere, self-paced, and interactive learning [15]. E-learning is defined as learning conducted via electronic media, typically on the Internet. It has been emphasized that successful e-learning depends on the self-motivation of individuals to study effectively [16]. E-learning has supported significant improvement in interactivity, collaboration, and delivery of online education [15].

E-learning relies heavily on application of technology to enhance the process. Most of the time, the learning process is supported using online/virtual learning management environments. Those are the places where materials are offered, information is shared and, in some cases, students have chances to do tests and practice. Most common of those kinds of learning environments are Moodle and Blackboard. Their popularity comes from accessibility, easiness to use and support offered. On the other hand, possibilities they offer are limited and for majority of times, they are just used to share lecture slides [17, 18]. There are more specific learning environments, like Ville [19], that offers more features and is able to fully support distant learning. ISC learning environment in Tallinn

University of Technology is one of the latest and is one of the first to implement constant learning process without teachers' interference.

Another factor in learning is the learning methodology used. Traditional classroom is usually associated with topic-based learning where every week a specific topic is covered. It has been found to be very ineffective, very passive and not engaging [20-23]. Furthermore, it has been found that traditional classroom fails to develop conceptual understanding of covered material [24].

Therefore, many researchers have focused on finding other kind of learning methodologies that would be more effective, more engaging and what would use the possibilities offered by technology more thoroughly. Examples of that kind of methodologies are flipped-classroom [12, 25], problem-based learning [26, 27] and serious play [28, 29].

A step further from those methods is so called competence based approach to learning. There are many different terms (competence based learning, competency based learning, competence driven learning, outcome based learning) to describe educational models with similar concept. They differ in details but share similar characteristics and assumptions that learning should be guided by predetermined outcomes [30]. Those outcomes are acquiring concrete skills and knowledge. Often, the outcomes are just called competences. Rozeboom defines the method as learner-centred education, integration of practice and theory and the integration of "head", "heart" and "hands" [31]. In current work, the term "competence based learning" (CBL) is used. Furthermore, CBL means a different thing for many researchers. For some, it is a system, for others it is a methodology, a form of assessment or a model for curriculum [32]. In current work it is considered to be a learning methodology. CBL has also been described as a shift from teaching towards learning [33].

CBL has recently gained momentum in educational community. 10 years ago very few people wrote or mentioned competence based approach in learning in general. It was mainly used in medical education [34]. Today, on the other hand, the term "competence" is used widely by many researchers (for example [35-39]) and the ideas of measuring learners' competences have been adapted from small elementary skills in a course to curriculum level. It has also been stated that using e-learning, distance learning and the development of competences are important factors of Bologna process [40].

The aim of this thesis is to formulate a full learning framework focusing on acquired new skills and knowledge as a result of learning, formulate methodology how to transform classical topic-based learning environment into proposed learning framework and use technology to support personal, undistributed and effective learning for nowadays students.

1.1 Problem Statement

The aim of the learning should be acquiring new knowledge and skills. Technology has advanced a lot in the last decade and young people have been eager to catch up with the innovations. Education, on the other hand, is slow to catch up with the improvements [41, 42]. Classical classroom with lectures remains the primary method for teaching in higher education [12] despite being found ineffective and passive [20-23]. E-learning has gained momentum and virtual learning environments are used frequently by higher education institutions [41]. Sadly, most of the time their full potential is not used and they are just used as repositories to store materials [17, 18].

Technology has high potential for improving and enhancing learning if applied deliberately. E-learning offers possibilities to reach large crowds without too much extra workload as popularity of Massive Open Online Courses have shown [43].

E-learning can also be used to personalize learning in ways that is not possible in classical classroom. Every student has a different background and abilities. Classical classroom assumes that everyone has similar background and is able to learn in the same space. It means that for some, the learning tempo is too much and for others the material is too easy. Furthermore, nowadays students are gathering information all around them. Many of them have gained skills and knowledge from workplace as majority of university students are employed. In Estonia, more than 60% of students were working while studying in 2011 [44]. It can be assumed that the number has risen with years. Lately there have been more initiatives to consider knowledge and skills acquired outside universities but it is still not very widespread. In e-learning, student can learn in their own tempo and from the level they are actually on. On the other hand, high employment means that students have to share their time and effort between studies and workplace. This makes it hard for them to attend on campus classical lectures. Again, e-learning can be beneficial in that situation.

There are many different learning methodologies to replace classical classroom and topic-based learning. Competence based approach has lately gained a lot of momentum, due to big initiatives like Bologna and Tuning, concentrating on measuring learning outcomes. Despite many authors doing research in the competence/output oriented learning, the field has so far been heavily concentrating on curriculum design. Few authors have considered implementing this methodology in course levels. Furthermore, despite many authors considering the methodology, there is no general definition neither for CBL nor competence. Most of the definitions are very vague and lead to vaguely defined competences for the methodology that makes the assessment imprecise and summative. Therefore, a definition for competence that would allow clearly defined competences, is required.

In addition, one of the biggest problems so far, has been inadequate assessment of competences. Some of the problems have been raised from the vague definitions of competences used. Others, like the same weight of repeated and new mistakes, are common problem in assessment in general, not only in CBL or in e-learning.

Despite many researches using CBL, mainly in curriculum level, there has been almost no analysis of usage or the effectiveness of the model. Also, it has been rarely adopted in e-learning environments. No clearly defined framework for CBL applying e-learning benefits to its fullest, could be found from the literature.

The problem of high dropouts is connected to the problem of student employment. As students have two different obligations, studying may become their secondary concern. E-learning generally tries to support changed learner by offering possibilities to study anywhere and anytime. Sometimes that is not enough and majority of the work is left close to the final deadline. In CBL additional problem is that the amount of work required from the student is not linear and depends on personal performance. This complicates things even further and dropouts can actually increase with CBL. Therefore, to help students to follow their progress during the course with CBL is important research question with yet no adequate solutions.

Educational data mining is a huge field where different data mining principles and methods are applied in educational context. Those algorithms have been extensively used to analyse performance in e-learning environments [45-47]. Different models have been used to describe learner's profiles and progress, like Bayesian networks [48]. Most of these models require complicated calculations or scrutinising training. Symbolic data analysis has been rarely used so far, mainly due to the lack of general methods for all types of data. For example, there is a general method for dissimilarity (and therefore for clustering) for interval valued data [49] but there is no general method for histogram valued data. On the other hand, histogram values give most freedom to describe study behaviour features with high granularity.

1.2 Methodology

The research methodology for developing a framework for CBL in e-learning consisted of literature review to identify key factors common in previous works about CBL and to identifying shortcomings of current state of research in CBL, analysing the state of current e-learning environment using classical topic-based learning and stating goals for the switch in learning methodology. Based on set goals, additional literature review was conducted to identify possible solutions and approaches that could be interweaved into the framework.

To verify the effectiveness of proposed framework and to identify possible shortcomings, educational data mining principles are used in data analysis. Both hypothesis testing and exploratory data analysis are used.

1.3 Author's Contribution

The main contributions of the author in this thesis are:

- Development and description of competence based learning framework in the context of e-learning with granular grading, memory model and personal learning path (P3).

- Development of answer evaluation algorithms mimicking students' behaviour (P1, P4)
- Detailed analysis of different aspects of proposed framework (P2).
- Development of three different visual progress following tools and analysis of their usefulness (P5).
- Development of general dis/similarity measure for symbolic histogram valued data with different bin numbers and bin widths.
- Development of student learning behaviour models using developed histogram valued dissimilarity measure to predict final grade (P6).

1.4 Thesis Outline

This thesis is organized in the following way:

Chapter 2 introduces competence based learning and describes the current state of the art. As the learning methodology relies heavily on definition of competence and no single generally accepted definition exists, multiply of different definitions proposed by researchers are introduced. That will lead to the current definition of competence based learning used in a thesis. Differences between classical topic based learning and competence based learning are then considered. Chapter continues with the overview of previous research in the area of competence based learning, finding that competence based learning has been rarely used in e-learning and majority of research in competence based learning has been in the area of curriculum design. This chapter forms the background for the learning methodology, using literature review.

Chapter 3 described the state of the e-learning system ISC before CBL was implemented in the learning management system. The reasons why CBL was desirable are analysed and goals for the switch from classical topic based learning to novel learning methodology are set. Chapter continues with presenting framework for CBL in e-learning environment and steps from switching from topic based to competence based learning are defined. Assessment in competence based learning is described in detail with new method for answer evaluation proposed. This chapter forms the theoretical base for the implemented learning methodology.

Chapter 4 contains in depth analysis on how learning has changed with the shift in learning methodology. First impressions by students and teaching staff are covered and suggestions offered for less painful switch from learning methodology. Analysis covers the usage of competences in the system and their coverage of different ability levels. Students' learning time is analysed in depth to make sure that the proposed framework won't make them do unreasonable amount of work. Different aspects of students' study behaviour are covered. Effectiveness and usage of the proposed method for answer evaluation is also analysed. This chapter confirms the effectiveness and usefulness of the proposed methodology.

Chapter 5 continues the development of e-learning environment, as it was found during the analysis that dropouts and students finishing on time have not changed as desired. Therefore, multiply process visualisation tools are developed to support the learning process. The tools are described in detail and feedback for their effectiveness is collected from the students. Feedback is compared with data logged by the system. This chapter proposes solutions to overcome dropout problem but data analysis shows that visualization tools did not achieve their goals as students wanted clear conclusions not the data to draw the conclusions by themselves.

Chapter 6 continues tackling high dropouts problem of CBL as visualization did not offer adequate solution. Therefore, prediction model for student behaviour is proposed. Symbolic data analysis with different variable types is covered as it is found that students' study behaviour is most naturally described in that form of data. As no general method for histogram valued data with different bins and bin widths currently exists, a general method is developed. Clustering using proposed method is defined using dissimilarity measure and cluster compactness measure derived from the dissimilarity. To verify the correctness of the method, it is validated on two commonly used datasets and results are found to be similar with previously recorded results in the literature. A prediction model for student study behaviour in CBL is then developed and tested on one course using CBL framework during the whole semester. The results achieved from prediction and cluster composition are analysed.

2 COMPETENCE BASED LEARNING

The idea of competence based learning is both old and involving. Many details about the methodology are still being worked out and one single best practice has not yet been formed. It has been historically adopted by a narrow set of researchers and educators [50]. The first programs based on competence based approach were introduced in USA during the 60-s [50]. Despite the fact that competence based programs have been part of the educational systems in many countries for years, they have mostly been applied in professional and vocational training [51]. Only recently has it found application in higher education – mainly due to the growing gap between the academic curricula and the actual demands from businesses and society [52]. Due to the rapid development of technology, learning methods and learners have changed a lot since web technologies have made learning more adaptable and more accessible [53].

2.1 Definitions

There is no single definition for CBL as all educational models have similar characteristics but they differ in details, as explained previously. What those models have in common is that they lie heavily on term "competence". As there is no single universally accepted definition for the method, there is also no single definition for the term "competence".

The word competence is derived from Latin word "Competere" which means to be suitable [54]. The concept of competence was originally developed in psychology, referring to the individual's ability to respond to certain demands placed on them by their environment [54].

McClelland defines competence as the knowledge, skills, traits, attitudes, self-concepts, values, or motives directly related to job performance or important life outcomes and shown to differentiate between superior and average performers [55].

Klarus defines competence as mix of skills, attitudes and knowledge that makes the employee or graduate successful in society in his/her profession [31].

Parry defines competence as a cluster or related knowledge, skills, and attitudes that reflects a major portion of one's job (a role or responsibility), which in turn correlates with performance on the job, that can be measured with well-accepted standards, and can be improved with training and development [56].

Sampson and Fytros analyse various definitions of competence in literature and offer their own definition for competence as a set of personal characteristics (e.g. skills, knowledge, attitudes) that an individual possess or needs to acquire, in order to perform an activity within a specific context. Performance may range from the basic level of proficiency to the highest levels of excellence [54].

Sánchez-Ruiz and others also cover a broad variety of definitions and argue that educators usually define competences as indicators of profits, knowledge and capacities; employers and economists, however, associate them to with performance, productivity, efficiency and professionalism [57].

Other definition states that by competence, a good performance is defined in a diverse, authentic contexts based on the integration and activation of knowledge, rules and standards, techniques, procedures, abilities and skills, attitudes and values [58].

There have been few attempts to establish coherent terminology but those have had little impact to date [54, 57, 59, 60]. Weinert states that there is no basis for a theoretically grounded definition or classification from the seemingly endless inventory of the ways the term competency is used [61].

For current thesis, CBL is defined as a knowledge based methodology which concentrates on measuring what a person can actually do as a result of learning [62].

CBL contrasts classical topic-based learning where learning is focused on certain topics (e.g. Ohm's Law as part of Physics course) and then it is checked if student, in general, can solve exercises or knows theory relevant to that topic. In CBL, those topics are divided into smaller units, called competences, which should be acquired throughout the course. In current thesis, atomic portions of theory (elementary skills of knowledge atoms that cannot be divided into smaller units) have been extracted from previously used wider topics that are the foundation of proposed methodology [63].

To clarify CBL approach, the following example can be used. In classical "topic-based" approach, if simple mathematical equation x= (4*8+10)/2 is analysed, it is only checked if the x is correct. Intermediate steps like addition, multiplication and division used to get the final answer are ignored. In CBL, student's skills on adding, multiplying and dividing are also analysed and graded. If final answer given by student is 11 (instead on correct answer 21), it can be deducted that student made a mistake with multiplication part of the task (adding 4 and 8 instead of multiplying). Dividing and adding skills can be graded with high marks but multiplying skill with a low mark as that was the actual competence where student's knowledge was inadequate [63].

It has been recorded in a literature that faculty members have been the chief critics of CBL approach [39]. Some have seen qualification frameworks as an intrusion into the learning process [32]. Others expressed concern that CBL represents a deconstructionist approach to learning that fails to foster deep and reflective engagement [64]. Neem argues that institutions that offered direct assessment models of CBL, while utilizing course mentors and recognizing prior learning, were a variation of the correspondence school model [65]. Ford states that there are so many definitions and implementations for CBL that it is impossible to compile a list of pros and cons [39]. She also points out that effectiveness of CBL depends largely on results achieved in the context of goals set.

2.2 History and Previous Works

Despite recent significant attention to competence based approach to learning it is not a new concept [39].

The origins of competence based approaches go back almost 90 years [34, 39]. The term competence in education was first mentioned in the 1960s where it was associated with a report in the USA on the effectiveness of teacher performance [32]. It is considered to represent a fourth generation of competence models [32, 39]. The generations are described by Brown in [32] and with every generation the focus moves from process to outcomes [39].

Competence based approach owes its historical development to "objectives in education" movement with emphasis on minimum standards and performance competencies on all educational levels [66, 67]. That also prompted the implementation of several competency-based programs, ranging from small-scale projects, such as developing interview skills in residents, to large-scale endeavours, such as creating a competency-based curriculum for first-year psychiatry residents, or a baccalaureate program for physical therapy students. As early as 1972, the American Board of Pediatrics published one of the first comprehensive documents on this subject, entitled "Foundations for Evaluating the Competency of Pediatricians" [66].

Jones and Voorhees [68] examined fourth and fifth generation competency-based programs targeting adult learners in the USA. They found that most programs in postsecondary education focused on development and transferability of competency or outcome-based curricula in specific disciplines and to a lesser extent, specific workplace skills and institutional effectiveness[39].

Competency based approach has been widely covered in medical education [21, 34, 64, 69] but Carraccio, Wolfsthal, Englander, Frerentz, and Martin found in their literate review [66] that efforts to apply competency-based models to medical education and training were essentially stalled at the conceptual level. They found that most reform efforts centred on identifying general competencies and learning outcomes. According to the authors, both the failure to link curriculum and residency training to specific competencies and the lack of valid assessment tools and methods for evaluation of competencies limited the extent to which medical and health-related education providers were able to operationalize competency-based learning approaches[39].

Through the 1990s much of the literature focused on the debate surrounding the evaluation of competence [66] as typically the competence based approach has been covered in curriculum level [39].

Development of (online) e-learning, advances in learning analytics and adaptive learning technologies and the operationalization of direct assessment models to entire college degree programs signalled an evolutionary shift toward a sixth generation of competency-based education models [39]. Goals of increased productivity, effectiveness, and demonstrable outcomes have prompted expanded global interest in the development of new major competency-based education initiatives [38]. The sixth generation has been characterised by increased emphasis on direct assessment of competencies rather than instructor-led courses [70]. Direct assessment does not imply to be automatic and majority of cases still relays on compiled competence/learning outcome descriptions that are open for different interpretation. Also curriculum mapping against standards has been brought out as trend in sixth generation [70].

During the past decade there has been an increase in the recognition of learning outcomes and competency based education and testing as well as greater use of learning outcomes and competency models for shaping higher education curricula in the US [71]. In US, the main motivator for reform has been US Department of Education with its response to demand from the industry for more educated and qualified workforce [72]. In EU the change has been led by Bologna process. Although the concepts of learning outcomes and competencies were not mentioned in the original 1999 Bologna Declaration or in the Prague Communiqué 2001, they have appeared in every new ministerial Communiqué since 2003 [51]. The learning outcomes are described as "as one the basic building blocks of European higher education reform" [73].

CBL has been covered widely in curriculum development and mapping. Early model on how competences are affecting curriculum development in Australia have been covered by Brown [32]. It uses competency standards for jobs to derive statements on what a person is required to do in a job and specifies the standard on how this should be done. Curriculum document is then derived that states learning outcomes and criteria for evaluation. The third step is training delivery that includes all the ways learning is going to take place. It is also mentioned in the paper that learning outcomes or competences can be previously acquired or gained outside official learning (for example, at the workplace) [32].

Klein-Collins has compiled extensive overview of how CBL has been used in USA on degree programs [70]. She covers mostly two types of curriculums designed in general CBL approach. Majority of programs covered offer traditional course-based framework with competence framework embedded into it. She calls that approach very basic as the focus is on a few intended learning outcomes. Few higher education institutions covered by her are allowing students to earn degree by showing proficiency in a set of predetermined competencies. Students can do it by taking courses or preparing portfolios that allow learning from variety of sources, including work and other life experiences [70]. The latest also allows students to complete programs faster than with a traditional method – therefore time is not fixed for a student [39].

Kuh and others [74] provide an overview of current state of learning outcome assessment in USA. They found that with 84% of cases, colleges and universities have defined learning outcomes for undergraduate students. They find it to be the first step towards developing competency based curriculums. They find that students' assessment results are shared more widely now internally than it used to be but they also state that there is a long way to go for complete student competence portfolio. They emphasise that universities and colleges have to use those assessment outcomes to guide institutional actions. All those described methods of assessment require interference from institution (for example, exams have to be marked by teaching staff).

Carraccio and others cover the development of CBL, mainly in the medical education in USA [66]. They concentrate on the changes in curriculum development due to CBL. They found that CBL usually consists of four states: competence identification, determination of competence components and performance levels, competence evaluation and overall assessment of the process [66].

Mulder and other cover how competencies have been used in national curriculum development in EU countries [37]. Four countries – England, Germany, France and Netherlands are covered in detail. It is found that all four countries have developed standards and methods for CBL but in reality they have been heavily criticized. For example, CBL in England suffers from very vaguely and generally defined competences and therefore they do not have any discriminative power in assessments [37]. In Germany, the main problem area is the assessment of competences. France also suffers from proper assessment in addition to the fact that all assessment is carried out in simulated environments. In Netherlands knowledge, skill and attitudes are divided in implementation of CBL and emphasis is on skill part; knowledge is left out of focus.

Rozeboom covers the steps of developing a curriculum for CBL, using example from Netherlands [31]. He emphasises that the development of the curriculum should start with identifying required skills, knowledge and attitudes from the point of view of labour market. Only then can competences and professional situation based on those competences be defined. Competences are also considered in a development of curriculums in Poland [75, 76] and in Germany [77].

Tuning methodology is a project and initiative from EU to develop and reform curriculums to be comparable [78]. One of the characteristics of this method is the shift from input-based to output-based learning [79]. Tuning defines competence very vaguely and uses two classes of competences - generic and subject specific [79]. The method has been adopted by many researchers. For example, Sánchez-Ruiz and others cover CBL with Tuning in Spain [57]. They describe how they transformed Tuning ideas for non-engineering education to engineering curriculums by using questionnaires for academics, graduates and employers. They identified 85 educationally relevant competences. He argues that although many educational institutions display themselves with a competency-based educational concept, on the level of concrete educational programs and practices this philosophy does not materialize [57].

Tuning method has also been adopted widely outside EU [79], for example in Africa [80-82] or in Latin-America [83].

Curriculum mapping against standards has been covered by Ford [39]. She finds that most of the time, in the institution and program level, competence frameworks are broad statements of learning outcomes and serve as the standard against which specific outcomes are assessed and measured [39]. She brings out examples of standards developed around learning outcomes that institutions can adapt to their own requirements. She also says that Bologna Process can be seen as a framework supporting outcome oriented learning. All those standards covered by Ford are very conceptual, “high-level” and using vaguely defined competences.

Ganzglass and others offer an approach for creating a qualifications framework that would enable postsecondary institutions to reliably and consistently award educational credit for non-credit workforce education and training, regardless of where and how the training occurred [84]. They recommend building national competence based framework, lower barriers between credit- and non-credit bearing institutions and linked data system that would offer comprehensive picture of students’ learning outcomes.

Ewell [85] covers a development of standard and competency based framework for USA higher education that is now used in more than 400 universities and colleges in 45 states [39]. He emphasizes that in order to make the framework effective and actually working, institutions and their faculties will need to develop a comprehensive record-keeping system for posting, housing, and manipulating data on what competencies have been actually mastered by students at the levels claimed [85]. He insists that all the taught competences have to be covered in multiple ability levels and the assessment has to be able to evaluate students’ abilities adequately. He suggests that assessment should not be periodical but be embedded into competency framework.

Luimula and Roslöf cover how competences were mapped for a game development course in Finland [86].

Pata and others analyse how competences are mapped to curriculum in Estonia [87]. They emphasise importance of self-regulation as part of personal competence maturing cycle in their three cycle model for curriculum maturing. They also state importance of overview of student’s process. Competences mapped into curriculum in this work seem to be quite vague.

Baumgartner in Singapore has developed one of the best frameworks for CBL that does not cover only curriculum level but continues to describe how the framework is also applied to courses [36]. Her model consists of three levels of competences. First level has 8 very generic competence areas in the field of Information Technology. The second level contains small set of sub-concepts. Third level was added, according to her, to clarify second level competences in more detail. The core competences are used for course design and assignments in courses. The framework Baumgartner applies, does not use e-learning but uses technology to support course and curriculum design by having repositories [51]. She has found that despite numerous engineering curriculums having defined competences connected to them; in many cases the competences do not have any practical relevance in the course design or delivery [51].

CBL has covered less in context on technology supported learning. Sampson and Fytros cover CBL in technology enhanced learning [54]. They find that even though the term CBL has been used for some time, it has only recently caught the interest of researchers for combining it with e-learning. They analyse three standards (IEEE RCD, IMS RDCEO, HR-XML) for describing competences and find them lacking in aspects required to describe all possible competences adequately.

Koper and Specht also cover CBL in virtual learning point [88]. They state that CBL first appeared in e-learning around 2005. They also found that CBL has mostly been used in curriculum design level. They concentrate on integrating different levels (training, schools, university) of learning to facilitate lifelong learning. They also concentrate on three previously mentioned standards of describing competences and describe their plan to offer facilities that support the creation, storage, use, and exchange of formal and informal knowledge and learning resources in detail.

López-Fernández and Tovar have developed learning management system that offers students different materials and activities to develop variety of competences [89]. They concentrate on the motivation issue of using e-learning environment and not so much how the competences are achieved and assessed in their system.

Jiang describes computer-based online graduate program [90]. It consists of courses that can be enrolled online. Every course has quizzes and a project, from which student competences are deducted. They use Moodle and Blackboard that have limited personalization and no CBL options. They also state that due to the fact that students are not able to achieve competences in projects, 2-4 sample projects are offered, meaning that instead of achieving the competences, they are copying offered samples. Furthermore, they have not described how their learning process is led by competences.

Cheng describes self-regulated competence based e-learning environment for workplace training [91]. The system allows every learner to set up their own learning goals. Social learning and networking is also embedded to the system. CBL has been used as a pedagogical model for curriculum development and for learning resource management. Wang continues the development [92] and offers a model where every competence (she names them key performance indicators) has a name, rating criteria and value. Value indicates learner’s proficiency level and rating criteria is set up to assess related performance indicators at different proficiency levels [92]. She also did an evaluation of the effectiveness of the prototype system with a target and control group and found target group performing better in four measured fields (reaction, learning, behaviour and results) [93]. Sadly, she has not explained how the quizzes are assessed or what kinds of quizzes are offered in addition to self-assessment, peer-assessment and supervisor-assessment. One screenshot shows multiply choice question but there is no indication on what other kind of tasks (if at all) are available while using the system.

Bremgartner and Francisco Netto propose a strategy based on CBL that allows to analyse, update and recommend student profiles in Moodle [94]. They thrive to achieve personalized learning that finds students' weak spots and concentrates on them. At first, they use their proposed strategy on simulator only. Later they continue the development and test the improvements on a single course in real learning [95]. Despite showing promising results on real course test, they have not published if the system was ever really implemented into learning process in their institution.

Moodle, one of the most popular virtual learning environments, started offering basic CBL support since Moodle 3.1 released in May 2016. It allows teacher to define competences taught and connect competences with activities. It also allows manual competence grading for student by teacher [96].

Kuhn, Gold and Abke describe a method on how defined competences should lead to proper teaching actions [40]. At first, after competences are defined, they should lead to teaching aims that in turn will lead to teaching/learning methods. Proper experiments and tasks can then be developed and proper actions taken.

Importance of practical competences have also been covered on the aspect of designing specific courses with hands-on activities. For example, Hu and others developed hands on experience in mechatronic course [97]. Bredol and Harth improved Physical Chemistry course with different activities by using principles of CBL [98]. Yi and others developed measurement and control lab course with student centred approach in China to achieve practical competences for students. They also covered the competences achieved by teachers giving the courses [99].

In Germany, Margaritis and others have worked years to develop a competence model for computer science field teachers [35, 100, 101]. Their model consists of two sub-models: pedagogical content and non-cognitive competence models. They use expert interviews and literature review to derive 79 competences. The competences were, again, quite vaguely defined and unspecific (for example: "Textbooks - The teacher is in a position to select appropriate material, like textbooks, scripts or films for her/his lesson and to provide that material to her/his students. They are also able to create an advanced organizer together with the class" [100])

As can be seen, CBL has been covered widely in curriculum level and less in technology enhanced learning (e-learning). Many of the previous works have concentrated on describing competences in official capacity with very little focus on practical application of offered methods. Majority of previous works have also covered only single aspects of CBL. No fully functional framework that concentrates on competence identification, assessment, students' competence portfolio creation and management could be found in the literature.

3 APPLYING COMPETENCE BASED APPROACH FOR E-LEARNING SYSTEM

This chapter introduces ISC e-learning environment with its current state and features before competence based approach was applied. Motivations and reasons behind the switch in learning methodology are covered and framework for CBL is formulated. Also, instructions from transforming topic based classical learning environment into CBL are developed. Finally, automatic answer evaluation and grading, essential feature for non-disturbed CBL learning process, is covered with two different approaches applied.

This chapter is based on publications P1, P3, P4 and P5.

3.1 Background

3.1.1 E-learning System

ISC is an e-learning management environment used in the Department of Computer Control, Tallinn University of Technology (TUT) from the end of 1990s [102]. The system consists of exercises verifying theoretical knowledge and lab experiments for practical skills. It is used in the teaching of many different courses in the field of computer science; not only in TUT, but also in other institutions in Estonia. Some of the courses have overlapping material. The system started with the classic "topic-based" approach, where the set of tasks were divided into discrete topics [103].

At the beginning, the system was a place to store materials and share information (similar to early development of Moodle learning environment). Soon, it was also used to submit reports and home tasks. By 2003, learning was moved to e-learning environment, meaning that tasks and lab experiments were now done online with early automatic evaluation. Evaluation compared student's given answer with a set of possible answers with feedback connected to them. Those answers also had a grade associated with them (in scale from 0 (totally wrong) to 3(totally correct)). If no match was found, grade 0 was used. All students' activities from the system are logged – including all submissions and results. Tasks were grouped into similar topics (like Ohm Law). Every group had a scale with 128 levels associated with it, showing how well the topic has been acquired by the learner. Colour codes were used to indicate levels visually.

In 2005 courses started to consist of groups of exercises and first version of mobile lab kit, HomeLabKit, was developed allowing students to solve laboratory experiments outside classroom. The system also dropped deadlines and learning became self-regulated. By 2007, there were more than 15 000 tasks in the system.

For 2010, the system was fully web-based. Courses consisted of topics that had hierarchical structure so that harder topics cannot be learned before basic topics have been learned. Level 77 from 128-level scale was used as a point of "topic has been learned". After every submission in the topic, in the case of correct answer, the level of knowledge for current topic was raised. The raise was not linear. At lower levels the raise had larger step. At higher levels the step got narrower and in highest levels the advancement was one level at the time. The drop due to the wrong answer was implemented in similar manner – in lower levels the drop was not big but mistakes in higher level had huge impact. The learning environment was used to teach multiple courses in a field of electrical engineering and computer science. At the time there were two major courses, with more than 100 students a semester, taught with the systems – Signal and Circuit Theory and Operating Systems.

In 2010 fall semester CBL was introduced into ISC learning environment.

3.1.2 HomeLabKits

Hands-on laboratory experiments are most effective and important part of modern engineering education. Laboratories should be places where connection between theoretical knowledge and real world is made [104]. They develop students' experimental skills, ability to work in teams and communicate effectively, learn from failure, and be responsible for their own results [105]. It is also a form of active learning [106]. All of them are higher levels of Bloom taxonomy that are considered to be most important factors in showing that material has been acquired [107].

The problem with distance learning has been that it cannot offer the same level of laboratory experiments as classical on the campus learning [108]. Engineering education has to take advantage of developing technologies and traditional laboratories can benefit from the internet. Classical labs can be replaced by simulations/virtual laboratories or remote laboratories can be used where lab equipment is accessed remotely via internet. Although, simulations or virtual laboratories can be also used to overcome the disadvantages of traditional laboratories, any simulation is simply a model of a physical process, which is just an approximation that cannot reproduce every aspect of the real phenomenon [109]. Distance laboratories do let students perform real experiments with unexpected results but they do not provide the user with a feeling of real presence in the lab [110]. To offer alternative approach for distance laboratories, HomeLabKits were developed.

The first generation of HomeLabKits (HLK) was introduced in 2005 during classical topic based learning. In 2007, second generation of HLK was developed [102].

HomeLabKit is a box containing everything needed to perform lab experiments. Currently they are used in electrical engineering, measurement and microcontroller programming courses. Those kits contain AC and DC sources, multimeters, some electronic components (resistors, capacitors, and inductors) and specific course based test devices [111]. HLKs are flexible as their content can be varied and these variations are reflected by software so experiments may be specific for any kit. Devices used in kits are all individual with component parameters stored in the database [103]. All components in the kits are packed into strong transportable case as seen Figure 3.1

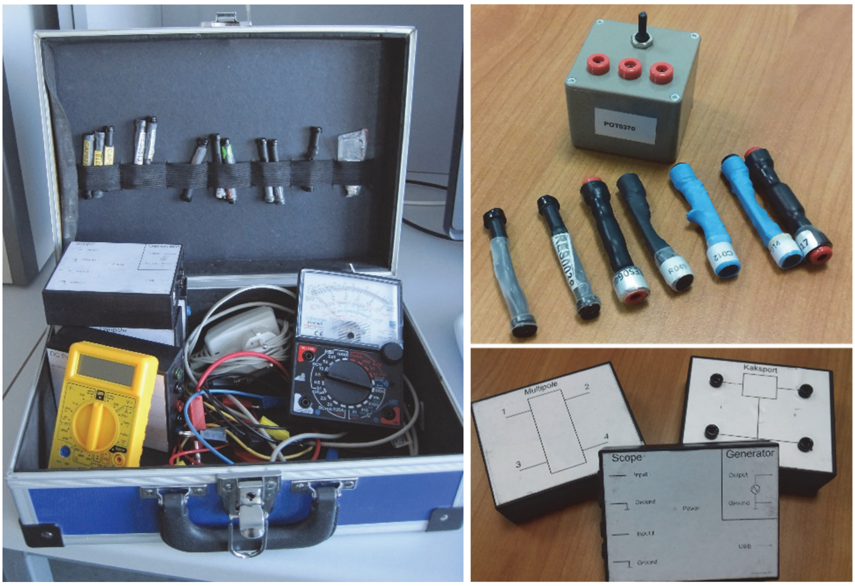


Figure 3.1: HomeLabKit for bachelor students and some of its components

Labs kits have two possible scenarios of usage. The first is that students use lab kits in laboratory with offered assistance. The second scenario is that student comes to campus, borrows the kit for a few days (ordinary students) or for few months (distance learners) [111].

Usually, freshmen are nowadays without any practical experience with electrical devices, measurement instruments etc. Most of the students start with a couple of hours in the lab but after that they will take the kits home, declaring that the freedom of choosing the learning time is extremely important for them. The average lending time is about 2-3 days, except distant learners in other locations, who can take the kit for some months. Currently there are around 70 kits available for lending or usage in the laboratory [111].

3.1.3 Motivations

The reasons why CBL was considered to replace classical topic based learning in ISC system are the following:

- The aim of the learning should be acquiring new concrete knowledge and skills and being able to analyse and apply them. We saw that topic based approach did not deliver this goal.
- New technologies. Nowadays students are very capable of adapting to new technology and expect those technologies to be used in learning too. Universities and teaching seems to adapt to changes more slowly even though students are expecting it [103]. For example, e-learning is one of those changes, and it has become almost de facto standard for learning in

higher education nowadays [112-115]. Smart devices are now being frequently used by learners and also much more often in the learning process [11, 114, 116, 117]. This all affects the design of learning resources – screen becomes smaller, learners interact differently with their devices, computational power is shrinking again and it cannot be expected that learner is able to connect extra devices to USB ports.

- Other commitments. Majority of the students work while they study [118]. Others have responsibilities for their families or have young children. All that limits students' ability to attend lecturers on the campus. Old-fashioned specific times and dates for specific lab experiments were not suitable for them any more as they were unable to attend them. They also tended to miss deadlines associated with classical topic-based approach and therefore opted to drop from the course. CBL with e-learning gives them flexibility to cope with private life, professional life and learning commitments.
- Changed learner. It is not only technology that has changed – the learner has also changed. Many students are not so much interested in learning new skills as part of the learning process, rather than just getting the final grade. Also, the grade itself does not matter – it is just the passing that matters. Student has changed from learner to a client or a king who should have everything in a way that suits him/her and who is never responsible for his/her shortcomings.
- Easy way out. Another change in the behaviour manifests in what students are learning. If they have option of choosing between more complex laboratory experiment or easier (by students' assumption) theoretical task, they take the easier theoretical task. They are very creative when required to find excuses for themselves and very harsh on mistakes made by others.
- Cramming in the end. In the majority of the courses, students tend to cram for the exams, concentrating all the learning to the last minute, a phenomena researched by many [119-121] . They go over large amount of information in a short period of time and are able to repeat and use it shortly afterwards to the best of their abilities. After the specific event (like an exam or a class test), the majority of them will not use that information again in close future and it is just forgotten, making the learning void [103]. With CBL and applying forgetting model, this unbeneficial behaviour can be overcome
- One size fits none. Every student is different and big classroom lecturers do not take that into account. Students tend to work with different speed and have different previous knowledge and interests. It is impossible to take every student's specifics into consideration when interacting with large community at the same time. Personal learning enables students to work on their own time on their own speed without the pressure of running out of time [103].

- Summative grading. In topic-based approach only one grade is used that represent the knowledge of some wider topic. It does not measure elementary competence but instead the sum of different elementary competences[111]. CBL enables to have personal picture of every student's learning behaviours and current abilities. That gives the teacher a better understanding of class level, what the students already know (even before they come to the first class, if they have used that system before) and where the gaps in their knowledge are. It saves time, avoids repeating material student already knows (but what sadly happens quite often) and gives the teacher a better idea how to plan course material between lectures [103].

All those factors come into play when CBL is considered. CBL can be adapted (and has been adapted) in classical/non e-learning but it is most suitable and can be used to its fullest when implemented in e-learning form. Therefore, it is well suited to be used with online learning that is accessible any time and from any place.

CBL also concentrates heavily on acquired outcomes. It should put all the emphasis what learner can actually do as the result of learning. So, CBL has been seen as a “solution” for students' changed attitude that they just want to pass the courses with minimal effort and time.

3.1.4 Goals

The goals hoped to be achieved with CBL are following:

- Acquisition of new concrete skills and competences. Learned competences are moved from short term memory to long term memory. Students know how and when to apply those competences.
- Repetition. Cramming in the end of the course for the exam does not serve the goal of acquiring new competences firmly. It would be more beneficial if the work is evenly divided and repetition will make sure that the skill is reinforced [103].
- Practical skills. Using learning control and course composition, it can be made sure that learner also has to solve and acquire practical skills by doing lab experiments.
- Overview of learners' abilities. Recording every competence separately and grading them independently will yield more detailed picture of single learner's abilities (for both the learner and the teacher) but also gives teaching staff overall overview of class process in elementary competence level. As more information is available, more deliberated and well considered decisions can be made.
- Accessibility. Fully web based learning environment facilitates learners desire to be connected 24/7 without time or location constraints so that having other commitments besides learning could never be used as an excuse for not filling the course requirements.

- Personalised learning – using e-learning and CBL allows finding most suitable exercises for a specific learner. Learner should study material he/she has not acquired yet as long as that learner needs. Also, course composition can be used to facilitate more personal approach – not all competences have to be acquired, leaving room for a learner to choose what he/she wants to study in advance topics.

3.2 Framework

The aim of this section is to offer a full description of CBL framework, describe the methodology of applying CBL to e-learning environment with all the benefits that technology can offer and describe the process of transforming classical learning environment into CBL.

3.2.1 Competence Based Approach

Competence-based learning is defined as knowledge based methodology which concentrates on measuring what a person can actually do as a result of learning [62]. Competence is defined as an elementary, atomic skill or knowledge that cannot be divided into smaller portion. Skills are practical abilities that learner acquires as a result of learning process. Knowledge is theoretical cognition acquired from learning materials. For example, assembling electrical circuit is a skill and knowing Ohm's Law formula is knowledge.

The definition of elementary competence comes from literature review where the main complaint has been that competences are very broadly defined and are ambivalent [37, 39, 57, 70, 79, 85]. Using elementary or atomic parts of skills and knowledge required makes sure that competences are defined as unambiguously as possible. It also makes it easier to assess them properly.

The aim of CBL is to enable learners to acquire concrete competences as a result of learning. Therefore, learning is controlled/lead by competences. Student indicates the competence he/she wants to learn and proper exercise with desired competence is found.

3.2.1.1 Grading

CBL requires different approach to grading than classical topic-based mythology. In topic-based learning, only one grade is given as a result of an assessment that covers a certain topic or a large portion of material. That one number, grade, is used to represent knowledge of all different elementary skills and knowledge used by the learner. Therefore, it could be said that grading is summative.

Summative grading has many downsides. First of all, it lacks granularity, making it impossible to have a clear and detailed picture of what the learner has actually learned. It is hard to summarize from the summative grading results which parts of the material were easier and which should be covered in more detail. Summative grading also cannot be easily used as an input for other courses due it showing no clear and detailed picture of student's specific abilities.

In order to have a detailed overview of student progress, grading has to be in-depth and extract detailed results. All competences have to be graded separately.

To achieve the desired granularity and fine-detail results, every competence in the system has its own score/ability level that represents student progress in acquiring that specific atomic competence. It means that when a competence is a part of more than one course, only one ability level is used – just for the competence, independent of the course.

When student solves exercises, every competence that is used by the student to solve the task is graded separately and independently, meaning that some competences can go up (if they were used correctly in the task) in the same time when others can go down (when those competences were required in places where mistakes were made) [63].

In order to facilitate that kind of fine detail grading without extra workload for the teaching-staff, automatic answer evaluation is used. It is achieved by using complex algorithms mimicking student answering process. If the final answer is wrong, algorithm tries, step-by-step, to find the place where mistake was made and identify the reasons behind it. Algorithm also generates proper feedback according to the mistake made. This kind of algorithm is a fair evaluator, lacking subjective bias and is always guaranteed to grade similar answers the same way [63].

CBL works best with high level of resolution for results. Therefore, instead of using common 5 or 6 grade systems for grading, scale of 128 different ability levels, similar to the scale used before CBL, is used to avoid discretization. Level 0 means that student has no knowledge of that competence and level 127 represents full and solid knowledge. Level 77 is called acquisition level to represent the point where it is assumed that after passing that benchmark learner has acquired the competence with minimum proficiency. It is expected, that at level 77, student is able to apply acquired competence (Bloom's taxonomy level 3 [122]) and use it in further basic exercises correctly. Also, it is expected that student continues to fortify the knowledge about that specific competence and acquire higher competences' and Bloom's taxonomy levels by continuing working on the system [63].

Level 77 is derived from historic submission data as psychologically acceptable learning goal and point where statistically so called "3dB" rule applies [123]. This rule means that learner's success rate should be at level -3dB. In other words, probability of correct answer should be close to 0.707 (2 correct tasks in a group of 3 tasks). This rule has been used in the system since 128-level scale was introduced.

3.2.1.2 Memory Model

Important part of the learning control is forgetting as one of the ultimate goals of learning is remembering [124]. To make sure the learned competences are strongly acquired, memory model is used.

When a competence is acquired at first, it is moved to short term memory where it fades if it is not repeated over and over again. Duration of short term memory is around 30 seconds [125]. Only with repetition, the competence, learned skill or knowledge moves from short term memory to long term memory. At first, the repetition has to be more frequent. With time, the repetition can become less frequent as competence is more firmly attached. Human forgetting has been studied thoroughly and H. Ebbinghaus was the first to show with his experiments that retention is very fast in the beginning, quite slow later and repeating is important to memorization [126].

To mimic the process of student forgetting, memory model is used. Memory model achieves two well-known principles of learning – repetition and spacing. Repetition takes care that the new competence is moved from short term memory to long term memory and spacing effects assures that learning is greater when studying is spread out over time[126].

The model uses learner's competence ability level as an input along with the time difference between last submission and current time. As a result, a new ability level returned is lower than the initial level, reflecting the forgetting that has taken place between last submission for this competence and now.

Ebbinghaus defined forgetting using exponential formula:

$$R = e^{-\frac{t}{s}} \tag{3.1}$$

where R is memory retention, s is relative strength of the memory and t is time [126].

The following form of the formula is used for the memory model [127]:

$$L(t) = L(\infty) + \big(L(0) - L(\infty)\big)e^{-\frac{t}{s}} \tag{3.2}$$

where L(0) is initial level at t=0, L(∞) is limit level.

Wixted followed up on Ebbinghaus with defining forgetting by using power formula that describes human memory more accurately [127]. Wixted power definition is the following:

$$L(t) = L(0)\left(1 + \frac{t}{\tau}\right)^{-p} \tag{3.3}$$

where L(0) is initial ability level, τ is time constant, t is time difference between current moment and last time competence was solved, p is decay parameter [128]. τ is set for 14 days by default. This is based on a published data, recommendations and processing previous data using exponential forgetting formula[123, 129].

Equation 3.3 is considered to be the preferred option by several authors ([127, 130, 131] etc.) for having one clear advantage: only one parameter determines the forgetting curve as time constant can be set to constant. Exponential form is determined by two parameters which must be varied; that is substantially more complex, especially finding final level L(∞) [123].

At first, exponential version of memory model was used but the analysis showed that power formula reflects more adequately how students are forgetting. For every student's competence, there is a p connected to it to reflect how fast this particular skill is forgotten by a specific student. The initial state of p is 1. P is recalculated recursively after every submission according to the correctness of that submission. The graphical representation of P can be seen in Figure 3.2. If P is smaller than 1, the forgetting is less steep. If P is larger than 1, the forgetting becomes steeper.

Decay constant is used to lower result (ability level) over time. Yellow dot shows ability level at the time of submission. As time passes, the ability level is lowered. When new submission is done at green dot, the ability level of considered competence is remarkably lower than at yellow dot. At the point of new submission, tangent line of forgetting curve is taken and it's slope recorded. Ability level is recalculated according to the correctness of the submission and new decay constant is found using the slope of the tangent line.

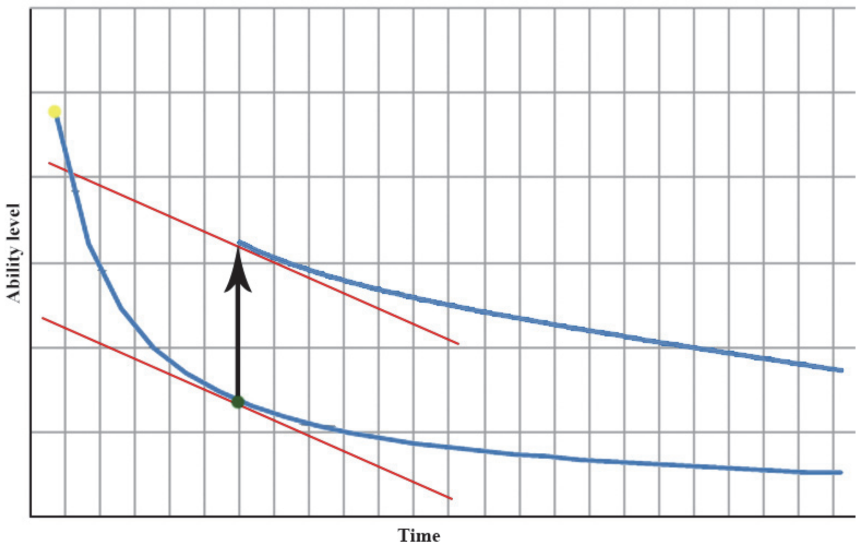


Figure 3.2: Graphical representation of decay parameter

As memory model forces learners to repeat the exercises, there has to be enough different tasks. Learners get very frustrated if they have to repeat the exact same task in short period of time as it has been observed in learning process when some competences did not have adequate coverage.

In order to facilitate repetition by a student numerous times, tasks have to be small and should not take too long to solve.

3.2.1.3 Personalization

All students have a different background, previous experience, interest and needs. Therefore, learning (sequence) should also adapt to student's persona and support the learner in a way it is needed for that specific student. The learning path should lead to maximal material acquisition and effectiveness of learning at the same time [132]. Learners previous knowledge and competences should also be considered [133].

First step of personalization, historically, was dropping deadlines (self-regulated learning). Abolishing deadlines also supported the benefit of e-learning: an ability to learn when and where the learner wishes to do so. Students have many different learning styles when time is considered. Some learn fast, being able to do a lot of tasks in a small timeframe. Others need time to concentrate. Some want to complete the course as soon as possible. Others tend to spread the material over the whole semesters. Some students only start to work during the final weeks.

The second important factor for personalization is finding a suitable task for a learner – learning process control. In CBL, learning is guided by a competence - learner selects a competence he/she wants to study. For every competence, learner has an ability score connected to it (also used by the memory model). Therefore, tasks should also have difficulty levels connected to it so that most suitable match between current learner's level (with memory model applied) and task's difficulty could be found. For better learning control, other factors should be also considered. There should be a list of previous tasks, solved using a competence selected, that are excluded from matching to make sure the same task is not repeated over and over again. This could be taken even further: a weight could be calculated for every task in consideration reflecting how many times the learner has already solved it. That weight could widen the gap between levels if task has been solved many times, or narrow it if task has not been previously solved by the learner. Also, the "3dB" rule could be considered so that student will always get some more challenging tasks if he/she is in a correct solving streak or the challenge could be adjusted downwards if learner tends to answer multiple tasks wrongly in a row.

Third factor of personalization is what learner is learning. On topic based learning every topic should be covered by the learner. In every course, there are competences that are very important and should not be missed but there are also other competences that are more "voluntary". Usually those are more complex, high-end, or very specific competences. To facilitate learners' interests and let learners have a choice, all courses, consisting of competences and weights that reflect their importance in a course, should have a slight overlap so that to achieve a maximum result, not all the competences in the course should be needed to be covered. Weights also allow factoring in the importance of core competences – it is not possible to pass the course without acquiring those competences.

Fourth factor of personalization is what learner has already learned. As many students learn the system in multiple courses and some of the courses have overlapping competences, they may already have done a portion of the course before starting it. Also, as every student can access all the competences and learn on their own time, it was not penalized in any way if the learner already had completed a part of the course before declaring it. Then, it started to appear that students started to abuse this, declaring courses with very similar content to get extra credit units without working. Therefore a value added model was included [134] which takes student's previous state into account and makes sure everyone is gaining new competences and knowledge during declared courses. This, again, personalizes the course even further.

Final grade is also part of the personalized approach to let learner choose what he/she is learning. As course consists of competences that have weights and learner's level connected to it, final grade becomes a function of those parameters, letting the learner to decide how well which competences to learn. The more competences with higher ability levels, the higher the grade.

3.2.2 Transferring Topic Based Learning Environment to Competence Based Approach

Now a new methodology for transforming topic-based e-learning environment for CBL is proposed.

All tasks from classical topic based learning approach will be transformed into CBL. Firstly, competences from current tasks in the systems are extracted. They should be extracted in a way that they check only elementary skill or knowledge used in a task every time it is solved. These are so called "input-competences". Every task/lab experiment is now connected to a list of competences used in them.

If it becomes visible that a task has too many competences connected to it, it would be recommended to split the task into multiple tasks contrary to suggestions found from the literature where it was argued that context-rich, multifaceted problems help students to develop more sophisticated skills [24]. In a proposed approach, smaller tasks take less time, their evaluation is straighter forward and they are repeatable. It should also be considered that the tasks would not take too long to solve: it has been found that students' attention span declines with a minute [135], the optimum time to solve a theoretical task is around one minute. For lab experiments this time can be longer.

All tasks also require more complex answer evaluation and grading – all those "input competences" have to be graded separately meaning that their scores can go up and down independently. Therefore, for every task/lab experiment and every input competence in them, it has to be defined how to check if the skill or knowledge has been properly used. It should also be noted that grading should not just be fully wrong or fully correct. Other intermediate steps should also be used to give a more precise picture of the learner's abilities. At the end of this step, every task has a list of competences connected to it and every competence for every task has a formula or an evaluation method how to evaluate its application by a learner in a current task. Those evaluation methods may be common for a same competence in multiple tasks.

Then, after input-competences have been mapped with evaluation methods, for every task, previous submissions with their frequencies are found from the historic data. It could be assumed that the correct answer is the most frequent one.

Other forms of answers reveal most common misconceptions and errors done by students. It could be said that wrong answers are the most valuable information where a lot of knowledge can be extracted. Now another set of competences is extracted. Those are called "secondary-competences". They may not be visible all the time the task is solved. They may only be graded at certain situation when a specific mistake is made. Those "secondary-competences" may be "input-competences" for some other tasks or they may be totally new competences that were not extracted previously.

Evaluation methods should also be defined by "possible-competences". They should also have a condition when they are evaluated.

At the end of this step, a complete list of competences is compiled. All of those competences can be "input-competences" for some tasks and may also appear as "secondary-competences" for other tasks. Every task has their own list from the compile competence list of "input-competences" and another list of possible "secondary-competences" that may or may not appear in a current task. All those competences connected to a task have a method for evaluation.

For every student, there is now a skill level connected to every competence in a complete competence list. A scale with high definition is recommended to give more precise overview and higher level of differentiation.

From now on the learning is not guided by topics but by competences. Learner chooses a competence he or she wants to acquire and a suitable task is found. The task is allocated by using the following information: competence, student's current knowledge level of that competence and a list of previous tasks solved with that competence. The suitable tasks should not be too hard (have matching or similar level), should have a required competence as an input-competence for a task and should not be in a list of 5 last tasks solved with this competence. The last criteria is derived from learners' frustration when they get the same exercise over and over again.

In order to find a task with suitable complexity, a level to reflect it has to be assigned for every input competence in every task. Initial levels can be assigned by a teaching staff or they could be extracted from learning history. The latter is preferred as teaching staff may under or overestimate task's complexity but logs reflect the true state of the task's complexity. Those complexity levels should be updated regularly, for example, after every semester, according to learning history.

Now a suitable task can be found for a competence which accords to learner's personal state.

In order to have a variety of tasks with different levels of complexity for every competence, input competences' coverage should be considered. The coverage means that every competence should have at least a certain number of tasks where that competence is part of "input-competence" list. Furthermore, there should be sufficient tasks for different levels. Not every level should have more than five tasks (criteria from task allocation) for every competence but a competence should have at least 5 very easy, 5 easy, 5 medium, 5 hard and 5 very hard tasks. Therefore, every competence, at a bare minimum, should have 25 tasks connected to it. It is highly recommended to have more in order to give variety to learning. If competences are identified with less than a bare minimum number of tasks, then new tasks have to be formed.

The next step would be defining courses. Courses will consist of a list of competences connected to a course. Every competence in a course will have a weight. The weight reflects its importance for course as a competence may appear in more than one course. The weights could be described as a percentage or a milli-credit unit. It is recommended that for further personalization the courses should have 10-20% overlap so that the learner has an ability to choose what he or she wants to learn.

If there are competences that should be acquired during the course but no competence has been defined nor any tasks connected to it, then a new competence should be added and at least a bare number of tasks created for them. As there is no historical information on task level, it is a responsibility of the teaching staff to define the complexity level. In this case, the complexity levels should be updated quite often at the beginning, according to the logs. Also, possible “secondary-competences” should be checked from submitted answers.

Learning materials could also be connected to competences so that if learner needs additional help during studies, they can find associated materials by competence. Connecting materials with competences also shows shortcomings when there are competences which are not connected to any material.

Therefore, the steps for transferring from topic-based learning to CBL are following:

- Extracting input-competences from existing tasks. If required, tasks should be split into smaller tasks.
- Evaluation for every input competence is defined using granular grading.
- Previous wrong answers are analysed to extract secondary competences that may appear in a task.
- Evaluation and condition of grading for those secondary competences is defined.
- Every task is connected to a list of input and secondary competences.
- Connecting complexity level to every input competence in every task.
- Every learner will be given an ability level for every competence in the system.
- Developing a method for finding tasks based on competence, student’s ability level in the competence and recent tasks solved.
- Checking competence coverage in general – if required, new additional tasks are added.
- Competence coverage according to grading scale regions is checked – if required, new additional tasks are added.

- Defining courses as a set of competences. Every competence in a course has a weight.
- Connecting materials to competences.

As a result of proposed process, a fully functional CBL environment can be achieved where learning is controlled by competence and shortcomings in areas of a course that where not covered with tasks or materials can be identified.

3.3 Mapping Competences and Competence Coverage

The process of transforming classical topic-based e-learning environment to CBL started with identifying the competences. The competence, in current case, was defined as atomic skill or knowledge that is used to solve a task that cannot be divided further into other competences.

The contribution of the thesis author for this step was to map out competences for Operating System from classical topic based course to CBL. Other courses were mapped by relevant teaching staff.

Operating System course, as a topic-based system, consisted of 4 modules (Process Scheduling, Data Types, Memory Management and Paging) with 26 classes and 8571 tasks.

Modules contained larger topics in a course. Classes distinguish types of tasks. All the tasks inside the class have same structure but different values. For example, class could be “Find x. x=A+B”. Inside a task those A and B would be defined. For example, a task could be “Find x. x=4+5”.

Therefore, it can be said that before CBL, Operating System course had a very limited pool of different types of tasks (26 for 4 modules, on average 6.5 different exercise types). That was considered by the author to be not sufficient. Therefore, one goal of transition was to come up with more varying exercise types.

As a result of mapping competences, 35 input-competences where extracted. Additional 13 competences were added to a course to cover areas that were not previously covered. Currently the OS course consists of 44 input competences (some of them have been dropped during the years as course has changed) as can be seen in Table 3.1. Additionally, there are 47 secondary-competences possible in OS course tasks. All tasks in OS course are theoretical tasks. No tasks were split during the process.

Table 3.1: Input competences with their coverage

Input competence	Nr of tasks
Bit2Byte	43
Byte2Bit	48
CPU scheduling, FCFS	307
CPU scheduling, RR	1182
CPU scheduling, SJN	665
CPU scheduling, SRTN	569
Dec2Binary	40
Dec2Hex	116
Dec2Oct	156
Hashing	303
Hashing with offset	303
Hashing: collisions	303
Hex2Dec	143
Memory manager BF	1115
Memory manager FF	1964
Memory manager WF	1118
Memory: Only and all pages/segments from virtual to physical	629
Mod calculation	347
Numbers in Words	506
Oct2Dec	158
Page replacement	929

Page replacement: FIFO	453
Page replacement: LRU	476
Page table: offset calculation	190
Page table: page calculation	330
Page table: page transfer	330
Page table: pages transferred completely	140
Page table: virtual to physical memory calculation	329
PageTables	330
Paging	2239
Paging: swap in	2239
Paging:swap out	929
Process scheduling: Can only read or write	1441
Process scheduling: CPU	3343
Process scheduling: More than one process can read	800
Process scheduling: Only one process can write	1283
Process scheduling: Priorities	1237
Process scheduling: Reading & writing	1441
Process scheduling: Resources	3347
Process scheduling: Resources are independent	1281
Process scheduling: Waiting	3340
Processes	3361
SignedByte2Dec	136
Bit values	51

Currently there are 12 299 distinct tasks and 59 classes connected to competences for Operating Systems, meaning that around 4000 additional tasks and 33 classes were created during the transition process to expand the course to cover more of the relevant topics, match additional input competences created and fix any coverage issues that arose.

Some of the processes were automated. For example, as tasks inside the class had similar structure, input competences could be assigned on class level. The connection between input competence and exercise is still recorded in task level as some fine tuning was needed in some cases to match specific nature of the task and its data. Secondary competences were all hand assigned after analysing log files of previous wrong answers.

Majority of the input competences in OS had adequate coverage with variety of tasks from the topic-based form. There were few with borderline coverages like “Dec2Bin”, “Bit2Byte” and “Byte2Bit” competences. Also competence “Bit values” had borderline coverage for some levels before additional tasks were added. The coverage for levels was found in following way: 128-scale is used for tasks’ difficulty level. It was divided into 8 regions with 16 levels in each. Number of tasks inside those regions was found. The bare minimum was considered to be 5 tasks and therefore every competence should have at least 40 tasks for adequate coverage.

In the total process of transforming the e-learning environment, 341 competences were first derived with 289 of them being input competences. Additional competences were later added and by the end of 2016 spring semester, there were a total of 539 competences with 459 of them being input competences in the system. Currently there are 40 116 active theoretical tasks and 942 active lab experiments. In 2012 April, there were around 28 000 tasks as many of the coverage issues were not fixed by then[136].

During the initial transition, it was found that in more than 700 cases, some of the coverage regions were not filled. So, on average, for an input competence there were 2.18 regions not adequately filled. Surprisingly, the biggest cap in exercises was the lowest part of the scale. The first region, levels from 0 to 15 that should contain basic exercises, was the area with biggest coverage issues. The problem, when investigated, arose from the fact that in some of the competences that contained advanced concepts, the teaching staff found it hard to form simple tasks that would fit into lower difficulty scales.

3.4 Answer Evaluation

Grading and evaluation of student inputs are the most important parts of the CBL approach. Everything else, all the theory and concepts can only work and be beneficial in CBL framework if granular competence separation and independent grading is achieved.

ISC system takes “authentic” learner’s input, meaning that even if there are some multiple choice items, the majority of tasks ask the student to fill in the answer.

The answer can be one integer or float number, multiple numbers such as a result of measurements, simple texts or filling complex tables/structures. In total, there are 20 different types of answers the system can handle. Answer type is connected to a class in the system, meaning that all tasks in the class have an answer in similar style. Therefore, evaluation algorithm is also connected to the class.

Competences appear in tasks (classes) in different form and therefore there cannot be one single universal evaluation algorithm to grade their usage. The derived method for transitioning from topic-based learning to CBL recommends connecting every competence in a task with an evaluation method. In practice, it is more suitable to have complete evaluation algorithm that covers all the competences and their evaluations for a class.

A perfect algorithm for answer (that may consist of several items) analysis would be able to find mistakes, understand the reasons why the mistakes were made and find the associated competences that were used to find the answer. It would also be able to alter given correct answers according to the mistakes made. Therefore, one mistake would not make the rest of the answer fully wrong, even if it depends on earlier calculations. The system has adopted the approach that answer evaluation should simulate the process of student solving the task. By simulating the process, the weight of repeated mistakes can be lowered, reasons of mistakes identified and useful feedback generated, and only student view to data used [103].

To facilitate non-stop learning process that does not require teaching staffs' interference, the evaluation should be automatic. Automatic evaluation also ensures fairness and consistency– all similar answers are graded the same way. Two different approaches are now described that have been implemented in ISC system by the thesis author.

3.4.1 Previous Works

Problems with assessments in e-learning, especially in context of CBL, are widely covered in literature ([37, 54, 137, 138] for example). The main complaint has been that current methods are not adequate enough or due to the vaguely defined competences it is hardly defined how to even assess them. Also, to facilitate a large number of learners, the assessment has to be automated. It has been stated that automated assessment is limited, disappointing, not adequate and without any real feedback [139].

It has also been noted that there are two types of assessment: formative that actually helps learner to learn and summative that certificates the learner [140]. The formative assessment and grading is the one that should be desired as only this one helps the learner to improve. Formative assessment consists of three steps: observation, intervention and regulation [141]. The first step involves general learning to generate the baseline. At the second step, problematic points are identified. Third step consists of steps to offer guidance, control and adjustments.

Different proposed methods in literature for assessments in e-learning in general, have been covered by Wang [142]. He also proposed his own method using web-based two-tier model. In his model, students get a set of questions randomly. If they answer question incorrectly, short instruction appears and question re-appears later randomly. This is repeated three times.

Assessment in the context of CBL has been covered by Sluijsmans and others. They state that despite many efforts put into deriving better face-to-face evaluation, evaluation in e-learning has not been researched in depth [143]. They found that exams in classic topic based learning tended to differ a lot from the tasks done during the courses [137]. They criticize the lack of proper feedback and using one-dimensional summative grades. They state that in order for CBL to succeed, it is necessary to use different kinds of assessments in which learners are not tested solely on remembering knowledge, but more on their ability to interpret, analyse and evaluate problems and explain their arguments [137]. They offer Four Component Instructional Design model (4C/ID model) as a solution. The model divides tasks into classes based on complexity. Learners start with simple tasks and move on to more complex exercises that represent complex situations from real world. Then they offer new approach to store results using three criterions – mix of assessment tasks for quality, standard set of assessment criteria and "horizontal and vertical" assessment. They argue that model they offer enables flexible learning.

Also they offer guidelines for developing assessment for e-learning with CBL. It consists of four steps: defining the competences that students should learn when solving a task, designing a performance task around those competences, developing explicit performance criteria to measure the extent to which students have mastered the skills and knowledge and usage of one scoring system or performance rubric for each performance task [143].

Guenaga and others cover complexity of assessment in CBL [138]. They point out that the main problems of assessment are: lack of scalability, subjective nature of interpreting student's development, difficulty of finding latent skills used and lack of assessment activities. They developed a complex data analysis tool called SCALA for analysing learning data to support teachers in competency assessment.

Brahim and others have developed a method for assessment of competences in CBL following formative assessment principles [140]. At the first step they identify competency gap between goal and current state. At the interception step they analyse the motivations and knowledge of the learner. Then they define optimal trajectory through the course and on the fourth step they define learning activities based on a personal diagnosis [140].

Schaeper measures competences by using self-assessment instead of direct assessment [77]. He later agrees that self-assessment may not suitable for objective measurement of competences but cites multiple authors who have found correlation between self-assessment and results in tests. He also emphasises that self-concept has strong impact on students' future action.

A lot of focus on the assessment has also been put into e-portfolios and their management in CBL ([70, 74, 144, 145] for example).

Previous works in the literature do not cover automatic assessment that does not suffer from summative grading and offers feedback according to mistakes made.

3.4.2 Algorithm Mimicking Student Answering Process

Answer evaluation in competence learning differs from answer evaluation in topic-based learning. As every task is related to multiple competences (even if we tried to divide exercises into small units, in majority of cases, they require more than one elementary skill to solve them), answer evaluation will not give one result for the whole task but different results for every competence that is present in the completed task. All the competences and results are independent and therefore some competence levels can rise and others fall at the same time [146].

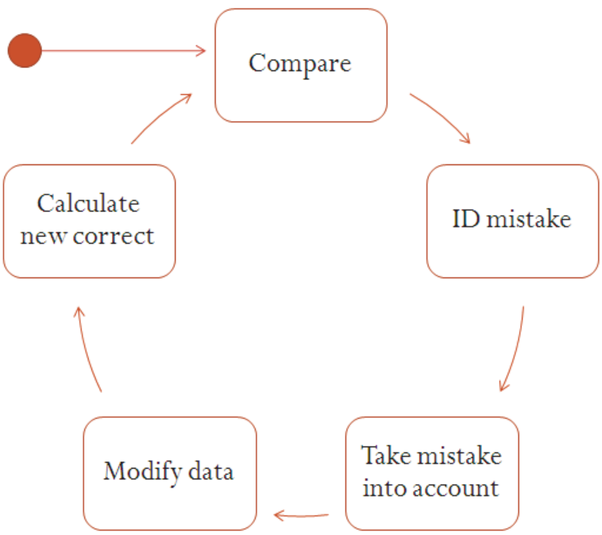


Figure 3.3: General structure of algorithm mimicking student's answering process step-by-step [103]

As tasks contain more than one competence, usually they also contain multiple middle-steps with their own results. If one of those results is wrong, the mistake is carried over and the final answer will also be wrong. As competences are graded separately, making a mistake with one competence should not be carried over to evaluation of other competences or affect other intermediate states. Therefore, the evaluation algorithm should mimic student's answering process step-by-step while solving the tasks. When the algorithm follows the steps student takes and uses only the data generated by the algorithm during the process, it enables the algorithm to detect only the actual mistakes. By simulating the process, the weight of repeated mistakes can be lowered and reasons behind the mistakes identified more accurately. Therefore, more adequate feedback can be generated and only student's view to the data can be used. The algorithm for that kind of evaluation is much more complex .

The evaluation algorithm starts by comparing the student's answer with the correct answer provided by teaching staff, as can be seen from Figure 3.3. If needed, some limits on how much of the answer is compared in a turn may be given. If a difference is found, the type of mistake is identified using list of pre-defined errors. Answer evaluation algorithm therefore not only contains a code to evaluate but also has a knowledge base connected to it with types of errors, how to identify them and what kind of competences that kind of mistake affects and how strongly.

If a mistake is identified, the source data of the task is modified so that from now on, the current state of student data is used. If the mistake was not identified (not in a knowledgebase), then the evaluation is stopped with "unidentified" mistake found. The submission is also flagged for a teaching staff for a review. In most of the cases, the unidentified mistakes are just random numbers students have submitted when they have no idea how to solve the task and they want to get a next one (due to the design of the learning control algorithm, even if they log out and come back, they always get the same task until the solve it – so student is stuck with a task until he/she solves it, otherwise learners would start to choose what they like to solve and what not).

After student's view of data has been achieved by modifying the original data, a new correct answer has to be calculated for the rest of the task. For this, part of the algorithm has to mimic correct solving process. After new correct answer using student's view is used, the comparison continues until the end of the answer has been reached or unidentifiable mistake is encountered.

The knowledge base with mistakes is generated from the previous wrong answers. Again, as with secondary competences, historical wrong answers are source of great information. Teaching staff may guess what kind of mistakes could be made but logs show actually what types of errors are most common by the learners.

Usually when new algorithm is created, the member of teaching staff uses his/her imagination on the types of mistakes students could possibly make and programs algorithm to detect and identify those. After a certain time period (usually 6 months), when enough information has been gathered, the log files for those exercises are scanned and the algorithm is updated according to the mistakes students make in real learning process. It has been found that this practice gives the best optimal algorithm efficiency [103].

These kinds of algorithms work very well if they are composed with care. They lower the impact of repeated mistakes and make sure mistakes are not carried over. They also offer much more detailed and adequate feedback than previous answer evaluation method. This kind of algorithm allows to determine competences used in a task and grade them independently.

The problem with this kind of approach is that the algorithms tend to be really complex as they have to contain multiple parts – error detection, data modification, knowledge base of errors and new correct answer generation. Creating this kind of complex algorithms requires good programming skills. Also, one algorithm is only suitable for a small portion of exercises. Analysis shows that currently there are over 1000 algorithms in use. Some of those algorithms are really similar with only minor differences. When analysing only their structure, 300 distinct types can be identified. This shows that a lot of work goes into writing those complex algorithms even if there is only need for a minor difference [146].

3.4.2.1 Example: Process scheduling in CPU in Operating System course

In the courses of "Operating Systems" there is a set of tasks where students have to mimic process scheduling in CPU, as can be seen from Figure 3.4. They are given description of three processes with time spent in CPU and resources declared. Correspondingly, to scheduling strategy given, student has to fill table and show in which moment of time where certain process is taking place [147]. There are four different methods a student may have to use for process scheduling. Before CBL, a set of those tasks were considered by students to be the most difficult one and they also had very low average result.

Originally the task had control algorithm what compared input string with teacher's provided correct answer string until the first difference. Then the percentage of the answer what matched, was used to calculate the result. With that kind of analysis, a mistake meant the end of answer evaluation. It did not matter how serious the mistake was. Even typos or carelessness mistakes like forgetting to assign an action to process in every moment of time were as crucial as misunderstanding the task or making mistakes in interpreting scheduling strategy [147].

That control worked efficiently if there were no mistakes or mistakes were made in the later part of the answer. Sadly, that kind of approach did not allow any proper feedback or identification of mistake made. In order to give students proper comments to their answer, as well as an understanding of their mistakes and suggestions on materials according to gaps in student's knowledge, a different approach was needed. The first phase of a new algorithm development contained an analysis of previous answers as the only source showing where and why mistakes were made. It was learned that majority of mistakes were done with 1) process priorities, 2) resource management (many students failed to understand that one file can be read by many processes but only one process can write a file and writing is preclusive for reading and vice versa), and surprisingly 3) many typos [147].

Process scheduling

Given
Proc 1: R1-2:P-3:R2-1:P-2:W1-2:P-2
Proc 2: P-3:R1-2:P-3:W1-1:R2-2:P-2
Proc 3: R1-2:W1-3:P-3:R1-1:P-2:R2-2
Strategy: SJN
Q=

Task
Fill the table by arranging process numbers 1,2 and 3 in appropriate place. In every row (moment of time) every process can appear only once until its work has not been finished.

Time	Ready	Processor	Resource1			Resource2		
			wait	R	W	wait	R	W
1		2		31				
2		2		31				
3	1	2			3			
4		1	2		3			
5		1	2		3			
6	3	1		2				
7		3		2			1	
8	21	3						

answer

Figure 3.4: Example of process scheduling task [147]

The first priority was to reduce the impact of typos. All processes can be in CPU, in waiting list or performing resource action in any given time. Many students had assigned process in one moment of time for more than one action. Therefore, a pre-submitting control to answer input table was added what raised an error message when a process was added more than once and removed lastly added process. That lowered typos and carelessness mistakes almost by half [147].

The main part of algorithm is analysis of mistakes as can be seen from Figure 3.5. First, the moment where student's answer and provided correct answer start to differ. Then the line, where the mistake was made, is thoroughly examined. Mistakes with resource are easily determined. Types of mistakes are numbered and number of occurrences is stored in array. Every time when a certain mistake is made, that number in array is increased. So, during the every moment of answer evaluation, the number of different mistakes and total mistakes are known. With every type of mistake, a proper comment is added to global comment variable [147].

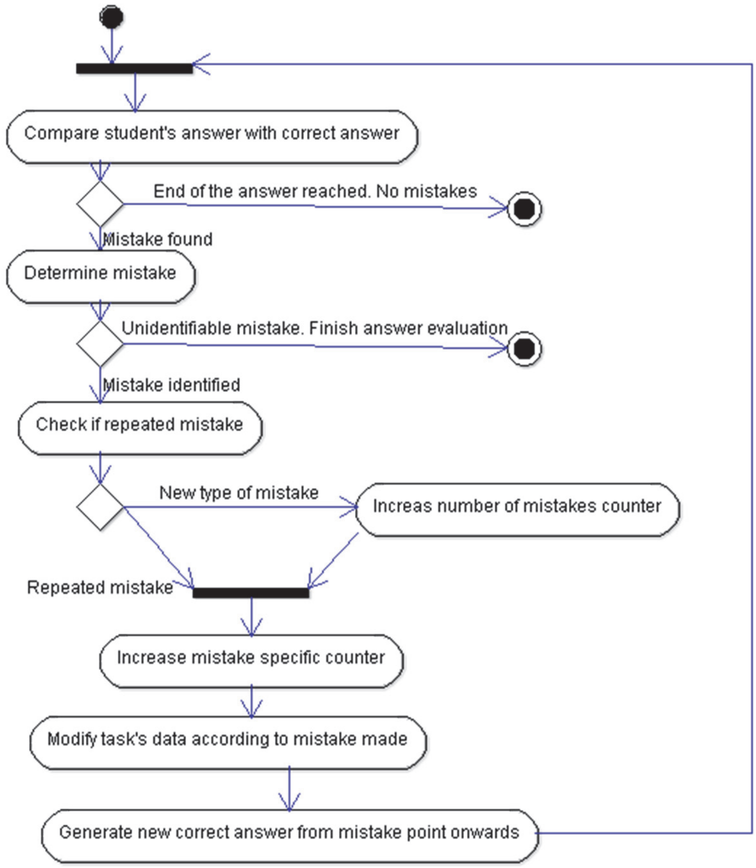


Figure 3.5: General algorithm structure for comparing student's answer

For example, if it is detected that at the time point 4, there are processes reading and writing the same resource, it is determined that competence "Process scheduling: Can only read or write" is not followed. Error is marked to have occurred and proper feedback is generated. In current case it will be: "Line 4: Currently processes are reading and writing to the same resource at the same time. Processes can only read simultaneously or write exclusively". The message will be added to global message what will be printed to the screen after the answer evaluation has ended [147].

When a mistake is determined and recorded for answer evaluation purpose, the current line is considered to be "correct". Therefore the "global" correct or teacher provided correct answer cannot be used anymore and a new correct answer has to be generated [147].

To obtain the rest of the new correct answer, another algorithm has to generate the new correct answer while mimicking student's thinking. With process scheduling that is easy as students have to follow certain rules and scheduling algorithm. Every time a mistake is made, the description of processes (task's data) has to be altered according to the mistakes made. With altered task's data, a set of rules on how to solve the task and point where the mistake was made, a new correct answer can be calculated [147].

With new correct answer, generated from modified data, the answer evaluation continues until the end of the answer is reached or an unidentifiable mistake is found. If new mistake is found, the process is repeated. New mistake comments are added to global pool of feedback and it will be only printed at the end.

Final result is not one number for the task as every task is related to many competences. In current example, up to 31 different types of mistakes can be determined and 12 different competences have been evaluated. Many different types of mistakes can affect one competence or one type of mistake can affect many competences. For example, reading and writing at same time, affects competence "Process scheduling: Can only read or write" but it also has smaller effect on competence "Process scheduling: Resources". Also, every mistake and total number of different mistakes is used to calculate an outcome for competence "Processes" and the competence corresponding to the algorithm used for process scheduling [147].

3.4.3 Automatic Synthesis of Answer Evaluation

The second approach to answer evaluation uses machine learning and concepts of expert systems.

As all exercises in the system use some set of rules that are applied to input data to derive the answer, the idea is to come up with common "world rules" that can be used with all of the exercises and then apply algorithm that is able to understand input data, which rules apply to. The result would be one system, so called expert system, that can be used to derive correct answer and also for mimic step-by-step student answering process. That system would also contain another set of rules for mistakes that are applied if inaccuracy is detected [146].

That kind of solution does not require having many different algorithms with only minimal differences. Only one copy of the system is maintained. Rete algorithm is used as a motor for pattern matching between world rules and current student input [146].

Rete algorithm is efficient algorithm for a large collection of patterns connected with a large collection of objects [148]. It has many possible applications and should also be suitable for storing a set of "world rules" for answer evaluation and matching given information with rules [149]. A basic production system checks each if-then statement to see which ones should be executed based on the facts in the current dataset, looping back to the first rule when it has finished. The computational complexity is of the order O(RFP), where R is the number of rules, P is the average number of patterns in the condition part of the rules, and F is the number of facts in the knowledge base [148, 150]. Algorithm tries to derive all possible new knowledge from rules, not only looking for specific knowledge.

The algorithm is efficient even when it processes large sets of patterns and objects, because it does not iterate over the whole set [149]. As a result, the computational complexity per iteration drops to O(RFP), or linear in the size of the fact base [149]. Patterns are compiled into a program to perform the matching process. The program does not have to iterate over all the patterns because it contains a tree-structured sorting network or index. It also does not have to iterate over the data because it maintains state information: the program computes the matches and partial matches for each object when it enters the data to the memory, and it stores the information as long as the object remains in the memory [150].

Rules given for Rete Algorithm are usually given in following form: if <condition> then <action> [148]. For example, resistors in parallel connection can be written down with following rule: If there are two elements, that are connected in parallel connection and both of those elements have resistance available/assigned, then the resistance of element created when combining those two in parallel connection, equals 1/(1/R1+1/R2) [146].

During the implementation, it was found that the complete knowledge base for all the tasks in the system would be very complex, it was decided to compile multiple knowledge bases. There is a knowledge base for simple unit conversions, one for basics in electrical engineering, another one for Operating System competences etc. During the implementation it was also found that applying the knowledge bases, if they contain many rules, is more time consuming than using those mimicking algorithms. Therefore, it was more beneficial to split the knowledge base between multiple domains. Currently, simple well known truths in electrical engineering field and unit conversions have been implemented as rules usable for Rete algorithm in Drools system [151] and prototype system is able to evaluate simple exercises correctly where those rules are used. Sadly, this kind of approach did not yield as many beneficial results as was first hoped and teaching staff did not find it easier to use than writing more "human readable" mimicking algorithms.

3.4.4 Confirmation

Important factor of answer evaluation is making sure the right person is doing the learning [152]– otherwise all the effort put into making sure the knowledge is acquired, goes to waste. It is important factor in e-learning [152]. As mentioned earlier, students tend to be ready to do anything to pass the course – including having other people doing the work for them.

Another reason for considering confirmation is learners' behaviour – some of them are afraid of doing practical experiments. At the start of the transition from topic-based to CBL, it became visible that as competences can both contain lab and theoretical tasks, notable portion of students avoid lab experiments every way they can. They never borrow a HomeLabKit, they never attend labs, and they try to complete the course without solving a single lab experiment. At the early stages, due to not anticipating this kind of behaviour, it was possible for students to get quite a good grade for a course without any experiments. That was found to be problematic.

To overcome the problem, the competences and their weights were modified so that there are some competences that only contain practical experiments. Usually these competences were given heavy weight towards the whole course but due to time constraints (solving lab experiments takes on average 5 minutes when solving theoretical tasks takes about a minute) it is not possible to make half a course consist of lab experiments. Therefore, changes limited the grade that student is able to get without solving lab experiments but it did not overcome the problem.

Thus, a term "confirmed" competence and class confirmation tests were introduced. After a student has reached acquisition level with a competence, he/she has to confirm the knowledge in a test on campus. Class test uses the same kinds of exercises that student has solved before. Only competences where student's ability level is higher than acquisition level are used. Confirmation means that learner solves the task and competences are graded. The level achieved in a class test (from 128-level scale) is considered "confirmed". Only confirmed levels and competences are considered for final grade. If learner achieves higher level with the competence in usual learning mode, the confirmation is lost and it has to be confirmed again in a class test.

Furthermore, class tests add blending learning level for the learning process [115].

3.4.5 Final Grade and Grade Suggestion Engine

As mentioned before, courses contain a list of competences with weights. As a part of the personalized learning, function that uses competences in a course, as well as weights and student's ability level, is used to give student the final grade.

As 128-level grading system is used, this has to be transformed into 6-grade (grades 1 to 5 and fail) system used officially. One option would be to assign levels to match 6-grade system but that would result in loss of details and grades will be sampled from high definition scale to lower detail scale.

In order to keep the granularity of the learning results, more complex aggregation function was developed, called grade-suggestion engine. It takes into account the official number of credit units assigned for the course. Current curriculum guidelines for grading were taken account when developing this engine. These regulations state that student should be able to pass a course with 51% of work done/knowledge acquired. Therefore, it is set that if student achieves highest levels for competences worth half the points, he or she should be able to pass. Also, if student confirms competences equalling 100% of required points to minimal level, he or she should also be able to pass. Those two points are used to derive a line. Below the line, a student cannot pass the course. Above the line, a grade-suggestion engine is able to suggest a grade. There are exactly 50 levels between acquisition level (77) and maximum (127). Therefore steps of 10 levels are used to define line-functions for other grades [63].

The formula that defines the line for pass on the course takes the following shape:

$$y = \frac{100 \times (x - A)}{-A} + 77 \tag{3.4}$$

where y is ability level needed to pass the course, x is credit units earned (sum of weights of confirmed competences above acquisition level) and A is credit units assigned for the course by curriculum. Formula 3.4 clearly shows that if x equals A (all credits done), y equals 77 – the acquisition level. And if x equals A/2, then (x-A) is receded to –A/2 and the first part of the formula (division) yields 50. 50+77 equals maximum level for competences that is 127.

Formula 3.4 can be evolved to take account all possible grade lines:

$$y = \frac{100 \times (x - A)}{-A} + 77 + 10 * (G - 1) \tag{3.5}$$

where G represent desired grade. It can be seen that if desired grade is 1, then last part of the formula disappears and same form as formula 3.4 is achieved [63].

Grade suggestion engine only uses competences and ability levels that have been confirmed in the class test. Therefore, every time student solves a task during a class-test, grade suggestion engine is called automatically by the system. Student's current grade is calculated. If the grade suggestion engine returns a grade for the first time, it is offered to the student. Student has the option to accept it and finish the course or he/she can continue working. If a grade has been suggested once, the engine will not pop-up a notification second time until a better grade becomes available, to minimize noise and extra actions required by a student. Also, learner has the possibility to take the grade anytime later, as long as confirmed competences' weights and their levels remain high enough for a grade-suggestion engine to be eligible for a grade.

For this kind of method having a maximum efficiency there are no deadlines except the official limitation of the end of semester. It is a last date when grades can be carried from ISC e-learning system to official registry and therefore students have to take a grade by that date or they achieve "fail to appear" [63].

3.5 Conclusions

In this chapter framework for CBL used in this thesis is formulate. It consists of detailed description of grading, application of personalization and memory model. Steps for transforming classical topic based e-learning environment to CBL are offered. The process of transformation from topic based methodology to CBL is covered by the author using example of Operating System course.

Also, answer evaluation in CBL is covered in detail as it is essential part for allowing implementing the approach to its fullest.

The main conclusions of this chapter are:

1. The aim of the learning should be acquiring new concrete knowledge and skills and being able to analyse and apply them. Classical topic based approach has not delivered that goal.
2. Usage of fine-detailed and granular grading with 128-level scale for each atomic competence allows to form a clear and detailed picture of students' current state. Definition of competences in CBL in atomic level is crucial for valid and profitable assessment.
3. Power law based memory model is more accurate to describe human forgetting than exponential.
4. Classical topic based approach is not viable to support personalized approach to learning process. With CBL different approaches can be taken to ensure personalization.
5. Using automatic answer evaluation with algorithms mimicking student's answering process, enables to lower the impact of repeated mistakes, generate adequate feedback according to specific mistakes made, grade only competences actually used to solve the exercise and combined with pre-submission check of common typos, enables an adequate evaluation of submission. Automatic algorithm synthesis does not yield good response times and therefore is unsuitable for proposed framework.

4 ANALYSIS

To verify the effectiveness of CBL and proposed method, analysis of changes in student behaviour is conducted. The analysis is fully conducted by the thesis author.

The analysis is conducted using educational data mining principles using both hypothesis testing and exploratory data analysis [153]. Data mining itself is a large field consisting of different disciplines from machine learning and statistics to database technologies and visualization. There is a specific sub-field called educational data mining that is concerned with developing, researching, and applying computerized methods to detect patterns in large collections of educational data – patterns that would be otherwise hard or impossible to analyse due to the enormous volume of data they exist within [154].

Educational Data Mining (EDM) is concerned with developing, researching, and applying computerized methods to detect patterns in large collections of educational data–patterns that would otherwise be hard or impossible to analyse due to the enormous volume of data they exist within [154]. EDM borrows from and extends from related fields like machine learning, text-mining and statistics [155]. EDM can be used to evaluate learning systems, evaluate scientific theories on technology enchanted learning. It can also be used to determine student parameters for modelling, for intelligent tutoring systems for example, informing domain models for developing student models, creating diagnostic models and alerting instructors, for example [155]. Romero and Ventura have done detailed literature reviews on usages on EDM [156, 157]. Data mining has been used in many different educational research projects (for example [158-161]).

For data mining, timeframe is limited from 2008 spring semester (sometimes 2008 fall). Data before that is also saved but a different database system was used back then and not all information, recorded now, was logged then. Therefore, to have adequate comparisons, in most cases, data from 2008 is used.

This chapter is based on publications P1, P2, P3, P4 and P5.

4.1 General Statistics

ISC system has 26 275 users from which 5 468 have used the system actively. 1 784 active users registered between the start of 2008 spring semester and before CBL was introduced. Since CBL has been used, 2 479 active users have registered in the system.

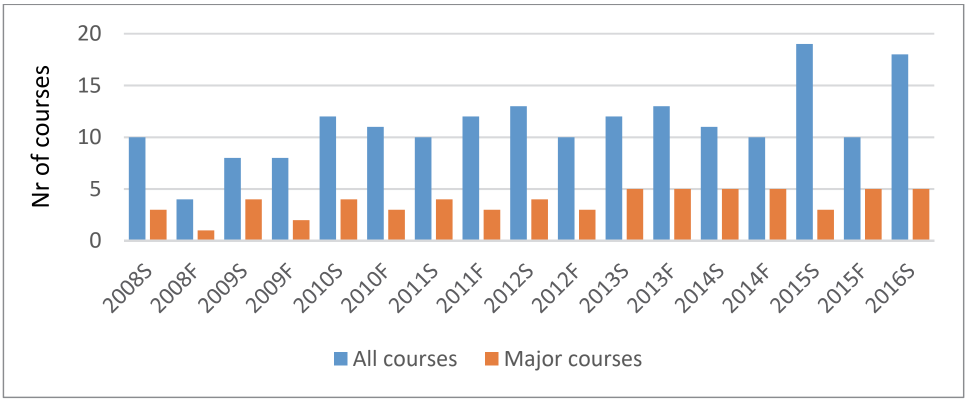


Figure 4.1: Number of coursers thought via the system per semester

Figure 4.1 shows distribution of courses throughout the semester. In a last few years the number of courses has risen as small custom tailored courses, offering only a small portion of bigger courses, have been started for many learners who were saying that they are only interested in a small part of the course and do not dare to declare a course with high ECUs. Many students have also been interested in small ECU courses to fill holes in their curriculum. Therefore, in Figure 4.1, major courses with usually around 80 participants or more are shown separately. On average both number of courses and number of major courses throughout semesters has risen.

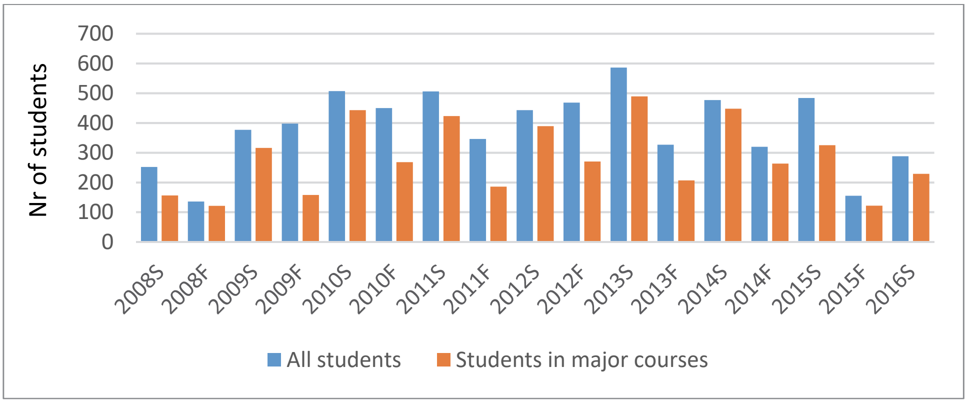


Figure 4.2: Number of students studying via the system per semester

Figure 4.2 shows distribution of learners by semester. As can be seen, during every semester, a majority of the students participate in major courses with around 25% of students enrolling in additional, smaller courses. The number of learners has been falling due to changes in demographic situation and the number of students enrolling to university.

4.2 First Impressions

When CBL was introduced in the system, it was first received negatively by both teaching staff and students. Teaching staff was so used to the topic-based learning and they were not willing to change their methods of conducting lecturers. The amount of work required to transform the courses was also scaring them. At first, there was lot of resistance.

Students had similar negative reactions. They were also used to studying in topic based learning style. They were quite skilled in knowing exactly how much effort they needed to put into the course to just pass it – CBL turned it all upside down. Unknown learning method scared them and many of them had trouble comprehending what exactly had changed and why. There was no real "introduction" material for students to CBL and they had to rely on teaching staff to explain them in person what is required of them in the new system. Some of the teaching staff also had hard time comprehending the changes. To confuse the matters even further, the switch to CBL was completed in the middle of the course so students started in topic-based learning environment and during the second month of the course, they were switched to CBL.

Therefore, for a more painless transition from topic-based to CBL, it is essential that at first the teaching staff is briefed adequately about the new method and what it changes for them and for the students. They should be encouraged to ask additional questions and address their fears. It would then be advised that teaching staff takes time for face-to-face meetings with students in order to explain the system before letting them try it themselves. The switch should happen between semesters to make things less complicated.

A positive lesson from the first impressions was that negative reactions from both students and teaching staff did not hinder the reform. First reactions did not stop the transition nor did it result with switch-back to topic-based learning. After the first alienation passed, students were actually the first who started to see the benefits of the system. By the end of the semester, students were commenting that they like the new grading were they see what skills have been graded and they get more feedback than before.

Furthermore, it was observed that students who were studying their first course in the system had more negative reaction to the change than those who had used the system in previous semester – it would have been thought to be the other way around as new students were not as familiar with the ISC system as returning students. Maybe more advanced students were more open to adapt different approaches and were able to benefit more than freshmen.

4.3 Competence Usage

There are currently a total of 539 competences with 459 of them being input competences in the system. On average every competence has 156.8 tasks connected to it where it is an input competence. The competence with highest number of tasks (2574) is "CPU, Round Robin" in Operating System course.

There are 3 competences that have less than recommended 40 tasks where they appear as an input competence, 5 for every scale range. There have been huge improvements of coverage compared to earlier years of CBL. In 2012, one third of competences were represented with 8 or less tasks. Half of the competences were covered up to 30 tasks. These were very low numbers and led to repetition of tasks [136].

On average, every theoretical task has 2.0 input competences connected to it with minimum 1 and maximum 11. They also have, on average, 1.6 secondary competences. Lab experiments have, on average, 2.3 input competences with minimum 1 and maximum 8. They have 2.2 secondary competences on average.

Those numbers show that despite splitting the tasks into smaller portions, every task and lab experiment contains more than 1 competence as it is hard to compose an exercise for just one skill at higher levels. The fact that the average is 2 competences per theoretical task is also a good result. It shows that despite not being able to compose tasks with single competence, most of the time, only one extra competence is required and learner can concentrate on a small portion of material. It also makes it easier to let learner repeat only tasks connected to skills he/she really needs to fortify. The tasks with high number of input competences are usually really complex tasks mimicking process in how operating system works in Operating System course.

Lab experiments also yield expectable results. The higher number of input and secondary competences follows the nature of lab experiments – they are usually more complex and time consuming than small theoretical tasks.

The usage of secondary competences is slightly lower than expected, especially considering that there are many possible mistakes a learner can make and their appearance in grading is not guaranteed.

Overall, it could be said that small tasks with small number of competences that are used together, has been achieved with CBL, making sure that the learner can concentrate on a small portion of theory and repeat specific competences required without too much overlap with skills already acquired.

On average a course contains 19 competences per credit unit. On average, the major courses with most of the students taking the courses have 4 or more European Credit Units (ECU). Courses have 1133 milli-credit units per official credit unit connected to it meaning there is slight overlap of competences per course to facilitate personalization.

When looking at connecting materials and competences, some separation has also been achieved. On average, every material has 5.5 competences connected to it. There are different types of help materials offered in the system – wiki links, pdfs, PowerPoint presentations etc. Pdfs are having a highest number of competences connected to one item – 8. Wiki links and videos have lowest average competences connected to them – 3. It would be preferred if materials are smaller and would give information about one competence at the time. Therefore, for the system improvement, pdfs could be split further to facilitate fewer competences at a time.

71% of competences have been covered with materials. On average, every competence has 5.3 materials connected to it. Therefore, it could be said that if there are materials connected to a competence, then there are usually enough different sources offered. The problem might be the coverage as less than three thirds of all competences available have materials connected to them. This is one area where system could be improved.

4.4 Students' Learning Time

One of the main goals for CBL was to get students to repeat the tasks over time to achieve stronger fortified knowledge. To achieve the repetition and spacing, tasks were divided into smaller units. As can be seen from Figure 4.3, the average number of tasks solved by declared students who finish the course per semester rose remarkably from 2010 fall semester when CBL was introduced (as there are different modules for fall and spring semesters, they are given in separately).

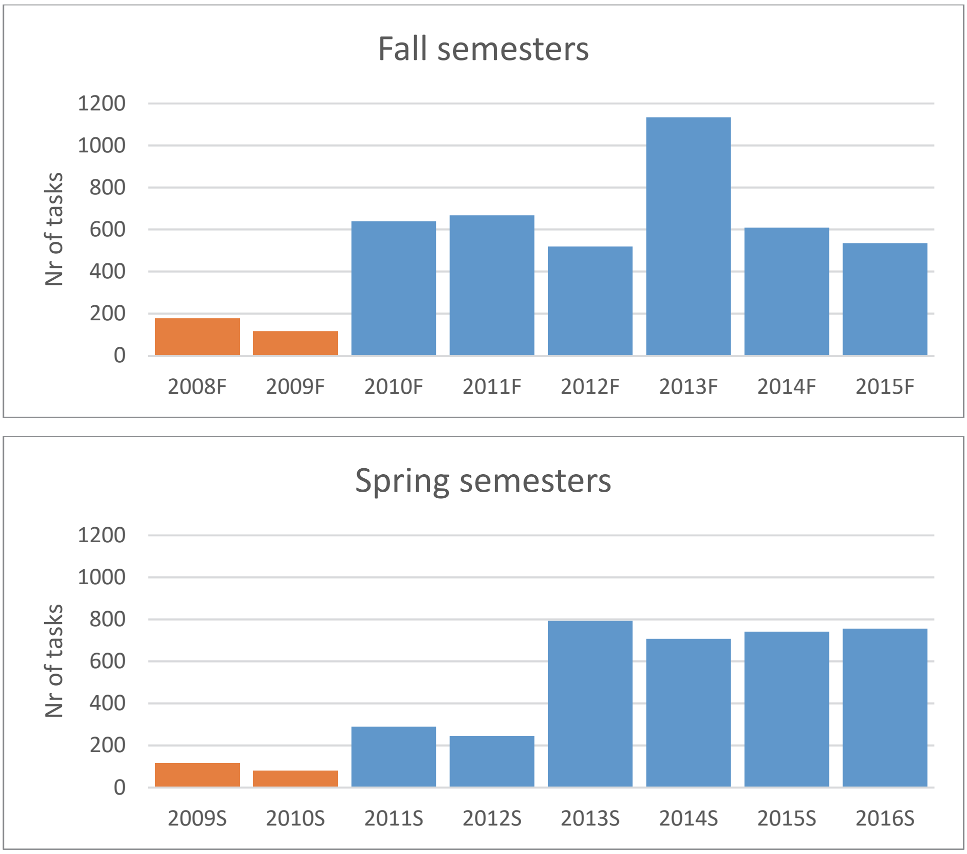


Figure 4.3: Average number of tasks solved in spring and fall semesters

Some of the differences in the Figure 4.3 are due to changes in curriculums. From 2013 spring onwards more courses were taught in spring. Before that there was just one major course in spring semester. In 2013, a new major course was added to the system and transformed into CBL. At first, when this course was taught at fall, students who declared it did enormous amounts of exercises. That kind of behaviour was not recorded again. Without 2013 fall semester, the average number of tasks per semester has been quite similar - around 600 tasks for fall semester and almost 800 for courses in spring.

It can be concluded from Figure 4.3 that repetition (and therefore higher number of submissions) has been achieved with CBL.

In topic-based learning, on average, solving a theoretical task took 4 to 5 minutes. In CBL it is down between 1 to 1.5 minutes that enables to solve more exercises without too much extra time [103]. 43% of students solve the tasks with 30 seconds [103]. Another 18% solves tasks with 10 seconds. Latest, sadly, contains students inserting random answers but that behaviour does not help them to make progress. Moreover, such behaviour can be detected and applied when evaluating competences.

When comparing the times spent solving exercises in the system (only part of the learning!), as can be seen from Table 4.1, it has increased. In 2010 fall semester, when switch was made, students spent more than twice the time solving tasks than their peers a year before. With years it has come down slightly but they are now spending around twice the time on solving tasks than before. It can be argued that the extra time is needed to make sure the competences are firmly acquired and the time is still within acceptable range. Also, time spent in the system reflects well on changes in average tasks per semester. If students, on average, solve fewer tasks, they also spend less time in the system, like in spring 2011 and 2012.

Table 4.1: Average time spent in the system solving excersises by semester

Semester	2008F	2009F	2010F	2011F	2012F	2013F	2014F	2015F
Time	7:12:16	7:31:21	20:28:57	17:44:59	14:09:58	24:31:11	14:23:39	13:51:11
Semester	2009S	2010S	2011S	2012S	2013S	2014S	2015S	2016S
Time	9:14:30	7:18:51	6:16:28	8:11:48	17:27:18	17:59:01	18:24:40	17:01:35

Overall, it can be concluded that tasks do not take too long to solve and therefore it is not a problem for students to repeat them. Slightly longer learning time is also beneficial as it was a consensus of the teaching staff that 7 to 9 hours in the system solving tasks is not enough for courses spanning whole semester (around half an hour per week).

4.5 Students' Results

The main aim of CBL was to fortify the knowledge and therefore students should show better average result with CBL than with topic based learning. CBL seems to show positive effect of repetition. For 2012, two years after CBL was introduced, the margin between totally wrong answers over all competences in the system fell from 17.69% to 8.34%. Slightly wrong answers also fell from 29.15% to 23.74%, meaning that totally correct answers rose significantly from 53.16% to 67.93% [136].

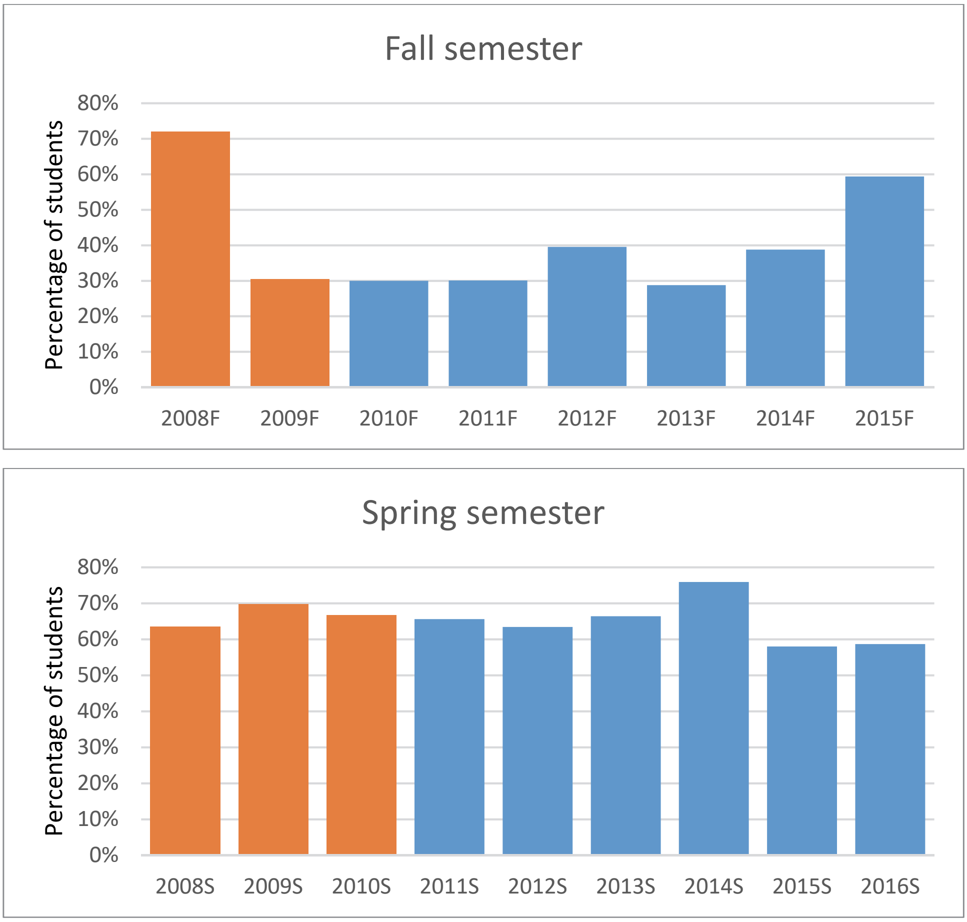


Figure 4.4:Percentage of students passing the course from all declarations

When looking at percentage of students who pass the course from all declarations, as can be seen in Figure 4.4, again clear patterns between fall and spring semester emerge that are same for both topic based and CBL approach. It used to be very varying for fall semester courses in topic-based approach. The drastic change in 2008 and 2009 fall semester is due to students' behaviour change – the major course that most of the students took during that semester remained almost the same for those two years. In 2009 fall semester the students finished later than a year before. Many of them needed additional semester or two to get the grade and more than half of them never finished it. Fall semesters lower passing percentage can also be explained by the fact that majority of the students who took fall courses were freshmen who have high dropout percentage.

Very low passing ratio for fall courses continues to CBL. 2015 fall, on the other hand, saw a remarkable rise and passing rate similar to spring semester courses.

Spring semesters have different kind of patterns. They tended to have around 60-70% pass rate before CBL and it has remained the same with highest in 2014 spring with 76% and falling with last two spring semesters slightly below 60%.

Therefore, it can be concluded that CBL does not put students into disadvantage and dropouts did not rise due to the switch in learning method. On the other hand, there was no significant drop in them either.

When looking at learners' final marks, seen in Figure 4.5, then it can be observed that learners have started to accept lower grades than before. Actually, it is not so simple and there has been a significant change during the years.

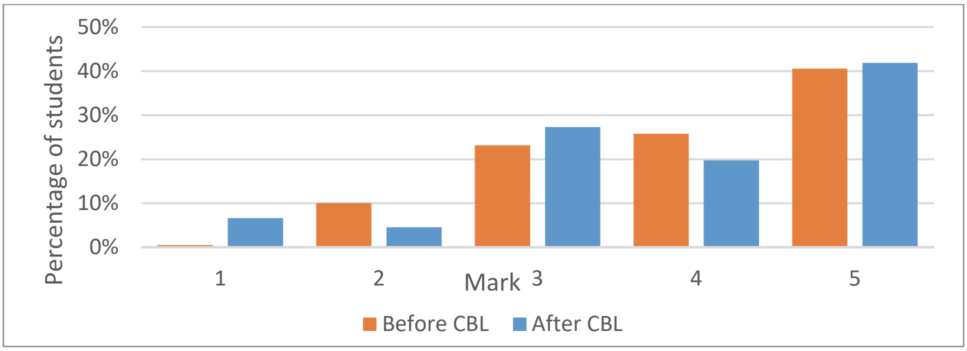


Figure 4.5: Marks taken by students who pass the course (2008S -2016S)

When comparing Figure 4.5 with Figure 11 in [103], it can be seen that on first five semesters when CBL was used, grades 1 and 2 were not taken at all, grade 3 was hardly ever accepted by the students and majority of them worked until they got 4 or 5. Spring semesters saw better average result than fall, similar to pattern before CBL semesters. First five semesters with CBL show better grades than before. Figure 1 in [63], with data up to 2014, is very similar to Figure 4.5.

This all changed in 2013 spring semester as can be seen from Figure 4.6. Three distinct pattern periods can be seen.

First of them is semesters before CBL where grade 5 was not so common and other grades, except 1, were used quite evenly. Zig-zag pattern of grade 5 can be followed in this period. The zig-zag pattern can be explained by different courses thought fall and spring semesters. A major course taught in spring, Operating Systems, has been found by many students to be easier than courses taught in fall semester.

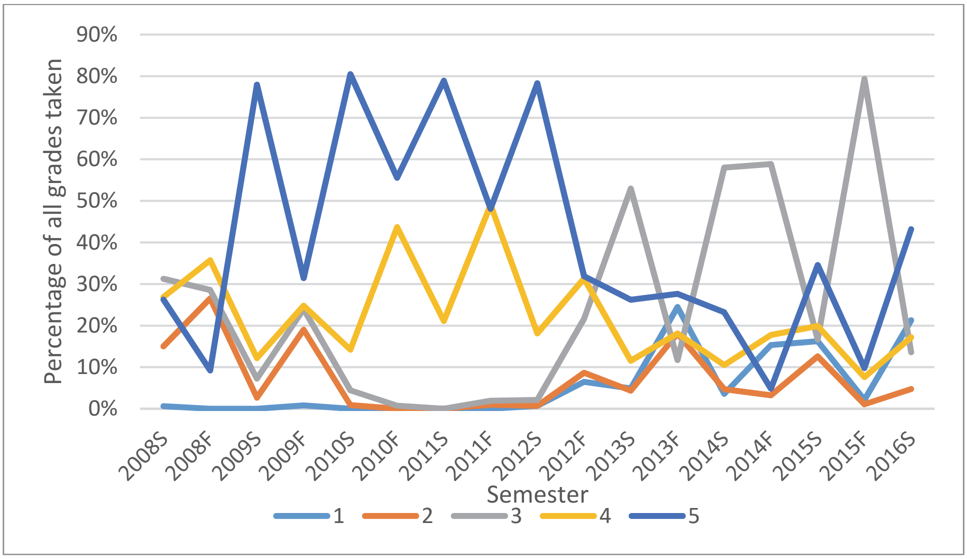


Figure 4.6: Change in grades taken by students by declaration semester

The second period is five semesters after CBL were clear signs of rise in average final mark can be observed. It continues the zig-zag pattern of fall and spring semesters, which was also observed during topic-based learning. On overall, grades 1-3 are hardly ever used with 5 and 4 achieved by majority of students in both fall and spring semesters. It shows that despite switch in learning methodology that made achieving good results harder, learners were very motivated to learn as long as they achieve a high grade. Therefore, it could be concluded that CBL achieved its goal to motivate learner to achieve higher knowledge and fortify it. 2012 fall starts to show early signs of pattern for third period but those students who take the grade in first semester (in 2012 fall), continue trying to achieve maximum marks. Those, who need more than one semester to complete the course, start to show behaviour where lower grades are also acceptable.

Third period starts from 2013 spring where students' attitude seems to change. Grade 3 becomes the most popular one and remains as one of the most often taken grades until now. Grade 5 becomes sparer, especially in fall semesters. Zig-zag pattern for grade 5 disappears for few semesters and re-emerges on 2014 fall but with much lower percentages.

Third distinct period can be affected by changes in university policies. From 2013 fall, all grades had to be achieved within one semester. Previously, students were able to have three tries for examination within three semesters. Also, in 2013, a new major course was added. The new course used pass or not pass as an official system but 5-grade scale in ISC system. Therefore, for many, grade 3 was acceptable as pass had been achieved. This one course cannot be the only reason for the change as similar behaviour of accepting lower grades is also recorded during spring semesters that have historically have much higher results than fall semesters.

Therefore, it was concluded, that it was students' attitude that changed, mostly affected by the policy change that grade has to be achieved within semester.

When looking at the time it took learners to complete the course, as can be seen in Figure 4.7, more students have started to finish the course quicker. Before CBL, even when deadlines were abolished, students were not very eager to complete the course before 4th month.

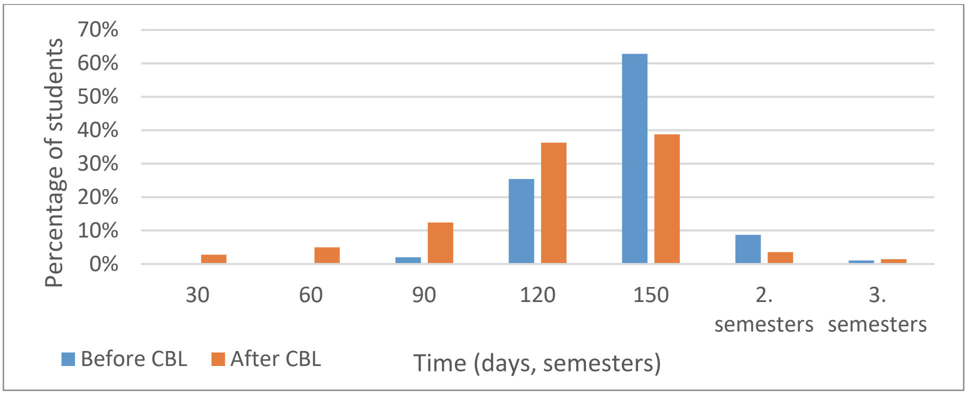


Figure 4.7: Time to complete the course [103]

As all the courses in the system are given via e-learning form, it means that there are only few lectures at the start of the semesters or no lectures at all. Therefore, a student does not have to wait until some material is introduced by the lecturer. Student can start working on the material on his/her discretion. That kind of approach has created interesting phenomena – some students complete course within first few weeks. Some of them even manage to finish before the official university date for course declarations (week 2). Feedback from those students has shown us that at the start of the course, the general workload is very light and therefore they have plenty of time to concentrate on the course. Students have stated that it makes their life easier even if they can get one course finished early and then deal with the rest. Automatic evaluation, the fact that all materials are accessible through the system and that there are no deadlines for in-semester submissions enables them to work intensively on the course for a short time period instead of spreading it over whole semester [63].

More than half of the students, with CBL, complete the course within 4 months of the semester. Exam session has always been busy time for "late-rushers", those who wait until the end to start working. Therefore, it is also expected that around 40% manage to finish within exam session and during the extra week after it.

Completing a course semester or two after the declaration has dropped, mainly due to change in university policy in 2013 fall. Before that, as can be followed in Figure 12 in [103], finishing within 2 semesters was more common.

When comparing time to finish the course and grades taken, another clear pattern emerges – early finishers have much better final marks than those who finish later. Early finishers usually get final mark 5. More variety appears in 4th month with high grades (4-5) the most often achieved. Students who finish after 4th month (exam session, extra week) usually get grades in lower scale (1-3) with few higher grades in the mix. Those who finish after original semester only achieve grades from lower scale.

Overall, a clear pattern can be seen in course finishing with CBL – students tend to finish earlier, if they can with good results. If not, they try to complete the course and accept the grade before the end of semesters. The late-rushers during the exam session achieve mostly low grades.

When looking at individual results, as can be seen from Figure 4.8, CBL has had positive effect. For comparison, both old and new evaluation system have been normalised to interval [0;1]. As with percentage of finishing the courses, spring semesters tend to have better average result.

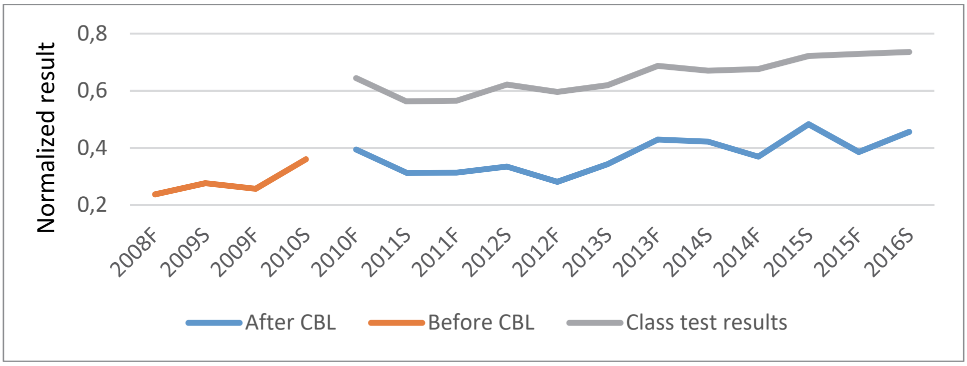


Figure 4.8: Averaged normalized results by the semester

When comparing class tests results with average results during class tests, it is clearly visible that they are much higher and have been rising by semester. This again proves that as students get comfortable with CBL, they can achieve high results that, hopefully, correlate to strong acquired competences.

When looking at the average result of all students by tasks' difficulty levels, Figure 4.9, it can be seen that there is no even trend. Some levels show slightly lower results than other. There is slight drop, as expected, with higher difficulty levels. With highest levels, on the other hand, there is a slight rise to reflect those students who want to achieve maximum result.

Interesting phenomena has been observed when comparing students' result by difficulty levels. Often, when a student gets a wrong answer in lower difficulty levels, it comes out that the learner actually had a high ability level for that specific competence. Last five tasks, solved by the students were high difficulty tasks and most of the time, they had very high average result for last five submissions with that competence. When they get simple basic exercises, they show inability to solve those simple problems despite being able to solve harder tasks adequately [136].

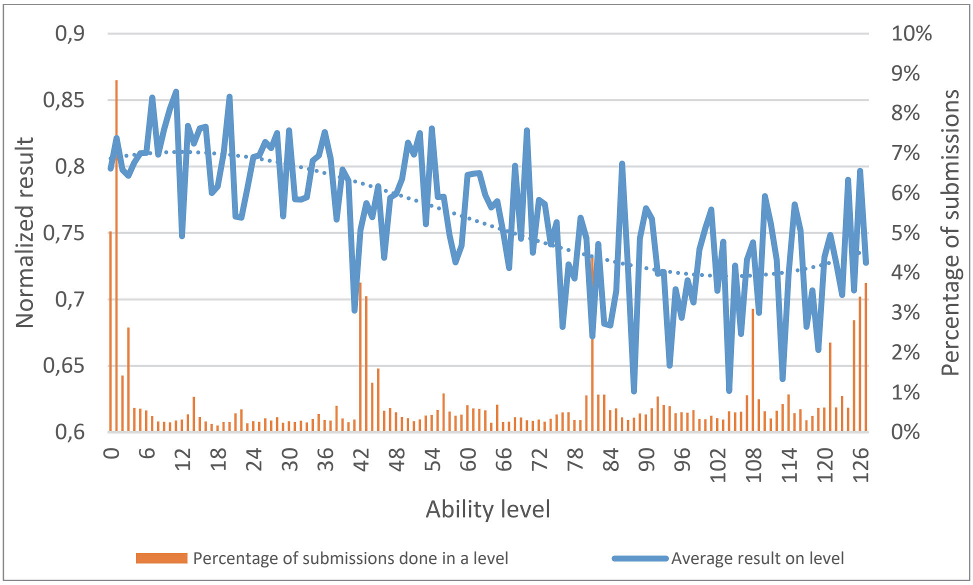


Figure 4.9: Average results by task's difficulty level and percentage of submissions by levels (2010F-2016S)

Also, the volume of submissions can be seen on Figure 4.9. Only the level of competence clicked is considered as tasks/ lab experiments are found using learning control algorithm by student's ability level for the clicked competence. There are some levels that show higher volume of submissions than others. As expected, low levels have a high percentage of submissions as many students never reach higher levels when they just try the competence or they drop out of the course. Therefore it is also expected that another high volume of submissions can be found at the other end with high levels reflecting those students who want to achieve best results. As the rise in ability levels on high difficulty levels is in singular steps, it is expected that highest levels have quite even number of submissions, as can be followed from the data. The peaks in the mid difficulty levels of learning control algorithm – from low levels, if answered correctly, student will move to difficulty levels around 42. Answering correctly will lead to level 81 and then to 108, all visible as peaks on Figure 4.9.

Figure 4.9 can be compared with Figure 5 in [123] that reflects data from 2010 January to 2012 March and uses non normalized interval -1 to 1 for student results. Similar three regions can be followed on both figures. The first region in Figure 4.9 has shortened and therefore the drop on average results starts with lower difficultly levels than in [123]. Also, the rise in higher levels starts later.

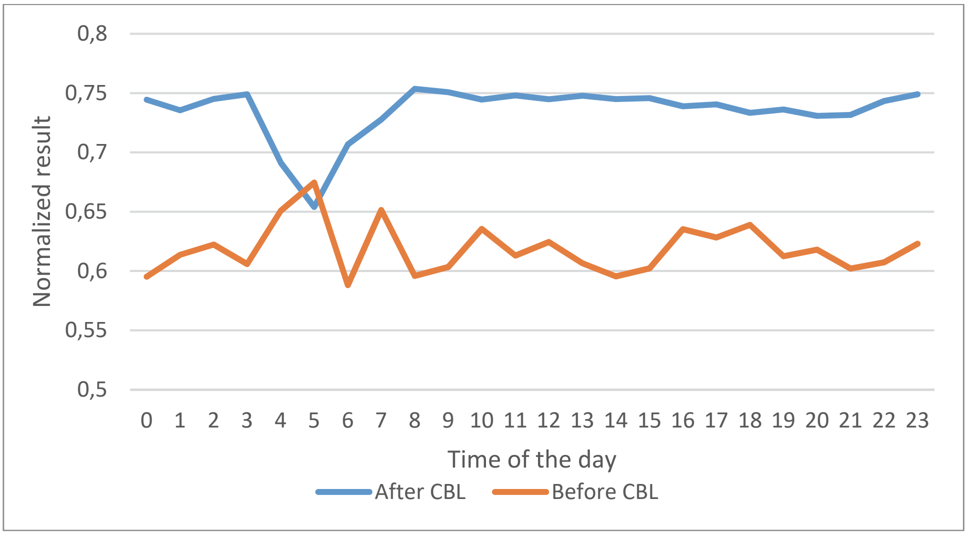


Figure 4.10: Average result by the time of the day before are after CBL 2008S-2016S

Change in average normalized result by the time of the day shows similar patterns – the results are now higher than before CBL as can be followed in Figure 4.10. This figure is very similar to one in [103], calculated in the end of 2012 when CBL had been used for five semesters. The patterns of the day are similar but, four years later, the line is higher. Noticeably, also the drop around 5 am is deeper than it used to be.

It was expected that the results during the night would be lower, as students are more tired than during on campus hours where onsite help is offered. However, the results show no clear signs of this. The results around midnight are actually better than in the evening. Highest average results are recorded at 3, 8 and 9 am. After 3 am the results reflect more of what would have been expected, as there is a drop in the average results, lowest at 5 am. Overall, the average result is more stable than before CBL.

One thing that differs remarkably before and after CBL was introduced, is marking behaviour. With topic based learning, small set of grades was mainly used to grade exercises. With CBL, there are still some grades given more often (-1 and +1), but values between them are used more widely now [103].

The acquisition level has been chosen to be 77 from historical reasons. Now with CBL, it was found that some competences may require higher or lower acquisition level to reflect their characteristics. The aim has been that if student has achieved acquisition level, he or she should be able to solve exercise correctly. Before customized acquisition levels, it was true in 63% of cases of theoretical tasks and 59% of cases of laboratory experiments [63]. Now, with competences having specific acquisition levels, the percentage of totally correct answers, after acquisition level has been achieved, has risen to 81% for theoretical tasks and 69% for experiments. The average acquisition level, over all competences, is 76. The aim of 3dB rule would be that out of three tasks, two of them are solved completely correctly to keep students both motivated and to provide them with challenge. Those new percentages are closed to 3dB rule and have been found more suitable to reflect specifics of every competence.

Forgetting is also very important for students' result. Comparison of students' results and time interval between submissions competence wise, can be seen from Figure 4.11. Data from 2014 spring to 2016 spring is used due to changes in logging in database. Previous data could be derived but for accuracy it is not done. Logarithmic scale is used to represent time using formula 4.1.

$$x = Round(10 * \ln T) \qquad (4.1)$$

where T is time interval in seconds between current submission and last submission done where the same competence was graded.

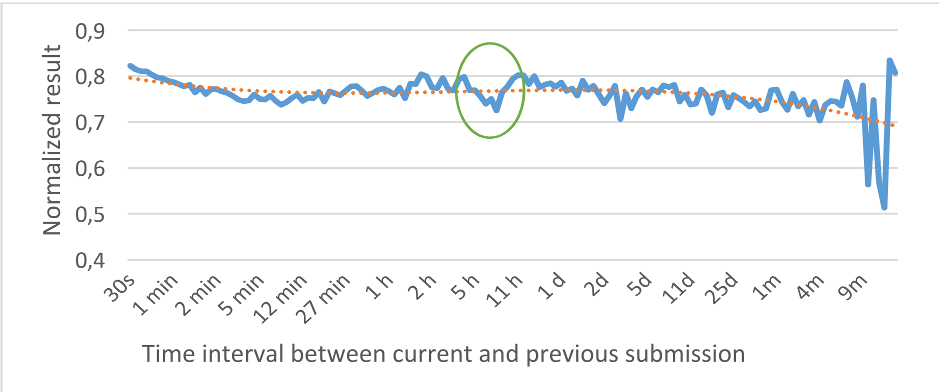


Figure 4.11: Change of average normalized result by competence vs time between submissions (2014S-2016S)

Three distinct areas can be seen in Figure 4.11. First is from 30 seconds to 2-3 minutes that has a very high average result at first, around 30 seconds when forgetting has not yet happened and knowledge is in short term memory [125], and high drop in results occurs. Second interval is from few minutes to few days and last one is up to a year. As can be seen, in the end (10 months to year and a half), the average results become jumpy due to lower number of data available. Second period is quite even and has a slow drop on average results. Third area sees higher drop as the interval between submissions becomes long.

An interesting interval can be observed around 4 to 12 hours. The average results are remarkably lower in that interval than in the others around it. Similar results have also been recorded previously in [123] on Figure 1. Around 4 hours between submissions, average results are high. In [123], the peak is recorded slightly later. Then a drop occurs in results, reaching lowest point at 7 hours. Again, in [123], the low point occurs later. And then the results rise to similar average with 12 hours between the submissions. Majority of those submissions are done in the evening before midnight, meaning that the previous submissions happened during 10 am to 2 pm during the day. Also, a large portion (31%) of those submissions is done on campus. Onsite help is offered on campus, therefore, at least in some cases, it can be explained that student was tutored on site and when he/she tried to repeat similar tasks later by him/herself, they were likely to have forgotten.

It is also interesting to compare forgetting and average results on Figure 4.11 and Figure 1 in [123]. The average result is much higher now when results on Figure 1 in [123] would be normalized. It should also be noted that Figure 1 in [123] does not use competence wise comparison but measures interval of actions. Therefore, Figure 4.11 is more accurate. Furthermore, the drop at high intervals is much lower, dropping at lowest point around 0.5 (0 in non-normalized scale on [-1 to 1]). On Figure 1 in [123] the drop below 0 occurs around 2 minutes. Again, results in Figure 4.11 can be used to verify the fact that with CBL the knowledge is more firm, even if there are months between using specific competence.

Overall, it can be clearly seen that with CBL the average results are higher than with topic-based learning. Therefore, it could be said that the aim of students achieving more concrete knowledge as a result of the learning has been achieved. Percentage of students who pass the course has risen slightly but is still low. Students are also more likely to finish the course earlier with CBL than before. On the other hand, undesired behaviour of accepting lower grades has been recorded.

4.6 Students' Study Behaviour

4.6.1 Time

When looking at the percentage of submissions over the day, the patterns have not changed much, as can be seen in Figure 4.12. The most active time of the day is in the middle of the day when students are able to attend on campus lab where supervision and class tests are offered. Before CBL that time of the day showed quite level number of submissions but with CBL, the afternoon has become more active than morning. The other more popular time for submissions is the evening when students are mostly working at home. Before CBL, working at home in the evening was more popular. Students' behaviour during the evening and early morning is similar with both learning methods.

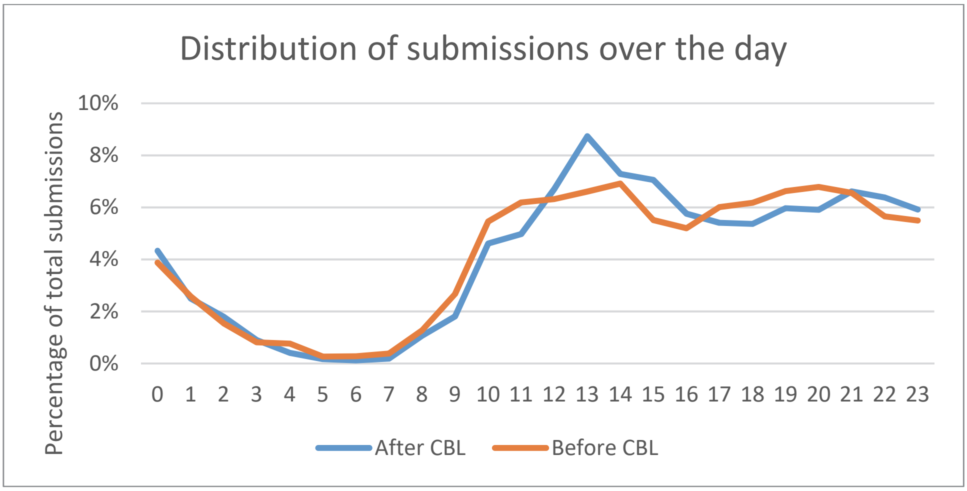


Figure 4.12: Distribution of submissions over the day

When students' learning pattern over the day has remained quite similar, there have been changes in their weekly study behaviour as can be seen from Figure 4.13.

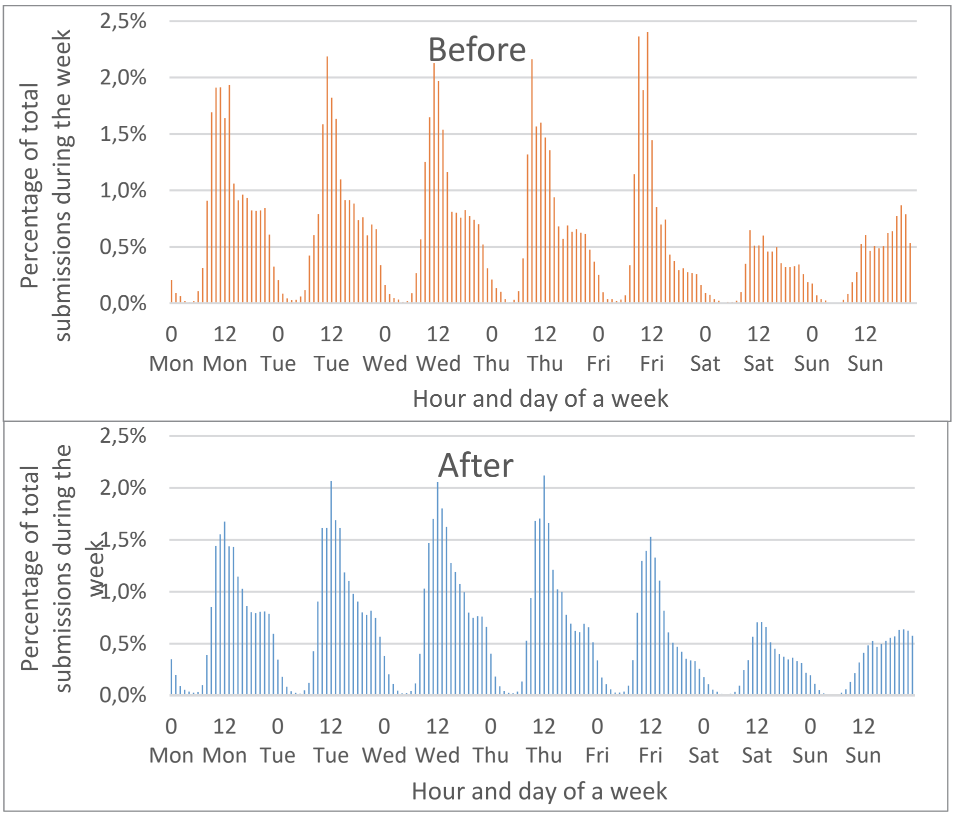


Figure 4.13: Students' weekly activity before and after CBL [103]

There used to be more fluctuations between minimum and maximum activity in a day with highs in the morning and lows during the night. The lows still exist but the peaks are not that high and have moved from the morning to the afternoon. Furthermore, the drop in the activity in the afternoon is more gradual. High student activity was connected to the offered on site laboratory times and tests and they still are – students now tend to request later times of the day and most on site labs and class tests take place in the afternoon. Before CBL, labs were hardly ever offered after 5 pm but now labs at 6 pm and later form around 5% of total on site activities.

Before CBL, Fridays were the most popular days where morning and noon times showed the highest activity of the week. Monday showed slightly less activity in the morning but rest of the workdays had very similar patterns. In the semesters after CBL has been introduced, that has changed. Wednesdays are more active and Thursdays and Fridays are slower days according to statistics. Mondays do not have a peak in the morning like rest of the workdays. Sundays do not show the peak at the evenings and working on weekends has lowered slightly. Students also tend to study more in the evenings than before.

The fact that number of submissions have risen with CBL is also visible in Figure 4.14. Semesters before CBL have quite an even slope for total submissions curve during the semesters. Some show slight rise in the end, others do not reflect late rushing.

In 2010 fall semester, it is clearly visible when students start to use CBL more frequently as the slope of the curve rises remarkably.

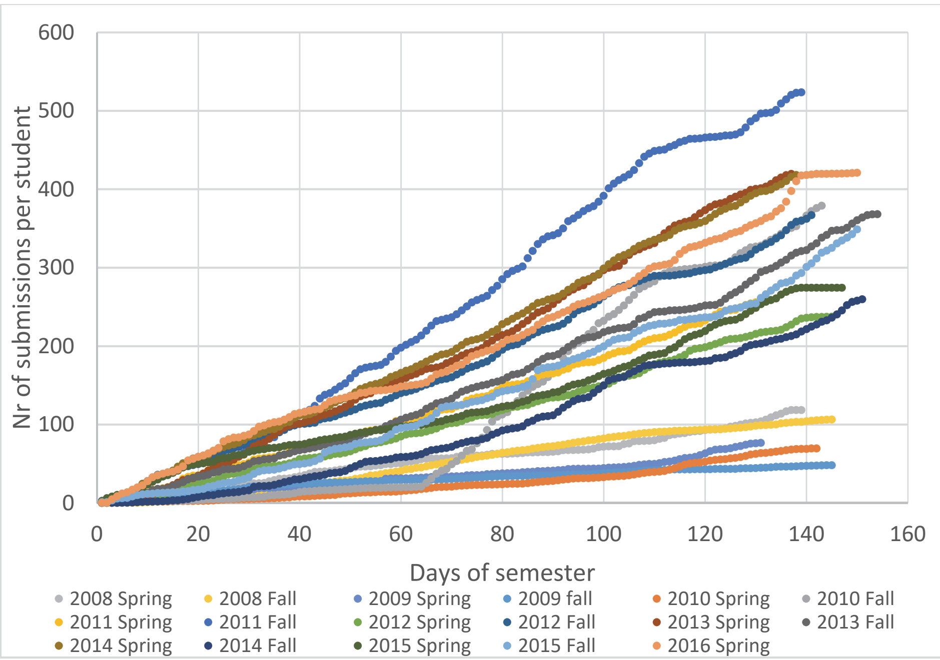


Figure 4.14: Number of submissions per declared student during the semester

Different semesters have different kind of curves but all of them have some periods were the slope gets very deep – usually at the end of the semester, showing the late rushing effect. Also, many CBL semesters show similar patterns to topic-based learning behaviour at the start of the semester, like 2014 fall and 2015 fall meaning that learners wait until they start to solve the tasks. Other curves, like 2015 spring and 2016 spring show quite an even middle part of the curve around 2 months mark. This reflects behaviour observed by teaching staff that in the middle of the semester, attending labs and doing class tests drops and students show less activity.

Not all of those behaviours cannot be assigned to CBL. A lot of those changes are due to changes in students' attitudes and possibilities. The move of peak time of a day to the evening can be a result of students' desire to attend university afternoon and work on a job in the morning. This pattern has been also adapted by the official timetable makers as nowadays master degree students rarely have any lectures before 2pm.

4.6.2 Competences

On average, out of 539 competences in the system, students have been graded on 102.39 competences. Compared to results of 2012 where system had around 300 competences, students were graded in 73.8 competences [136]. It shows that despite the number of competences having almost doubled; students have not caught up with it. In 2012, 70% of competences where students were graded, was above acquisition level, 77. Currently, on average, students have achieved acquisition level on 74% of the cases. The average result in 2012 for all the competences for all the students were 82 which is close to expected -3dB level 90. Currently it is 97, showing a remarkable rise with four years. Therefore, it could be said that if students have not caught up with the increased range of competences, they have achieved better acquisition and ability levels than they used to have four years ago.

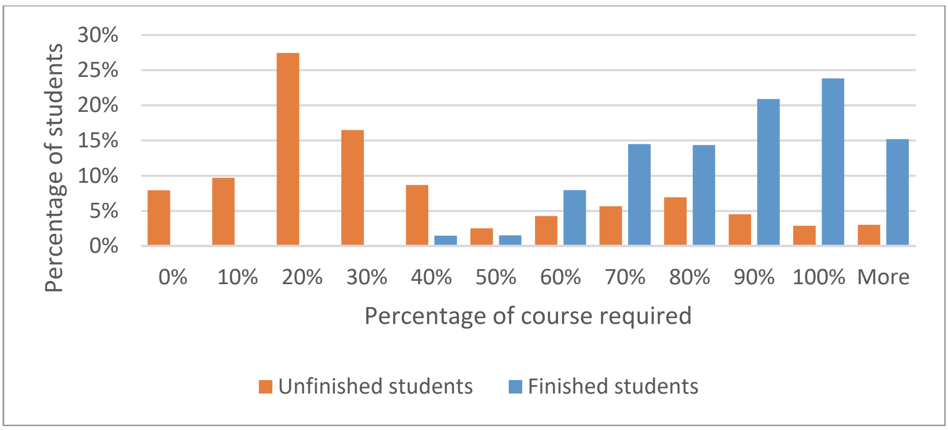


Figure 4.15: Percentage of milli-credit units earned by students who finished and who did not finish the course

When analysing students' behaviour in a course, it can be seen that a majority of students who finish the course, try to earn as many milli-credit units as possible, as can be seen in Figure 4.15. 15% of students who pass the course earn more than required amount of credit units to compensate ability levels to achieve better final grade. Those students who do not pass the course usually get stuck around 20-30% of the course.

When looking at a number of competences confirmed by the students who pass the course, as can be seen from Figure 4.16, majority of learners only complete 40-60% of all available competences. As mentioned before, there is an overlap around 10-20% for the courses so they do not need to complete all the available competences. The fact that many students (around 20%) confirm below 50% of course competences is not a desired behaviour. It shows that a student only learns essential competences with high weight to the course and they try to pass the course with minimal effort, without caring about the results or additional knowledge they might have acquired.

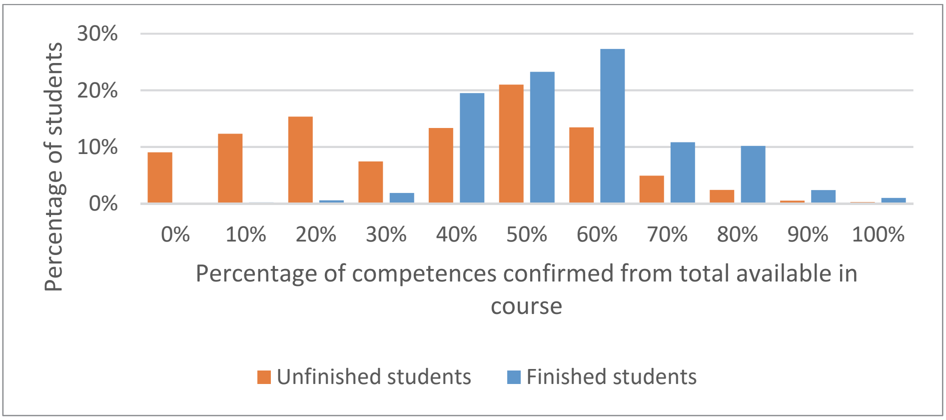


Figure 4.16: Percentage of competences available in the course confirmed by students who finished and who did not finish the course

Comparing Figure 4.16 with Figure 8 in [103], some changes in last 4 years can be noticed. When CBL was introduced, a lot of students tried to take maximum of the course and cover majority of the competences that were offered. This behaviour has disappeared with last 4 years. The peak has now moved more left, around 50-60% of competences. It could also imply learners prioritizing to basic, essential competences that have larger impact with higher weights and skipping optional competences. Also, the fact that some of the courses have been changed and additional higher level competences, not available when [103] was calculated, have had an effect on it. It is surprising that there are students who are in unfinished students' group with high number of confirmed competences but have not accepted the mark.

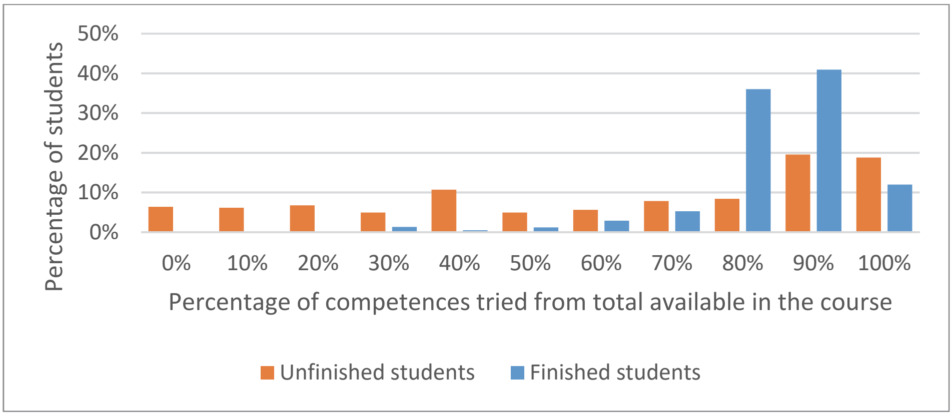


Figure 4.17: Percent of course connected competences students have tried

When looking at competences they have tried (both confirmed and unconfirmed) in the course and comparing results from Figure 4.17 with Figure 7 in [103], similar changes can be noticed. In 2012 when CBL had been used for two years, majority of the finished students at least tried all the competences. Now this peak has dissolved. Majority of learners who finish the course try up to 80%-90% of competences available. That means that around 10-20% of competences are never opened and therefore they never even try to acquire them. On the other hand, this 10-20% is within the usual overlap showing that learners have used the possibility to choose what they want to learn.

4.6.3 Confirmation of Competences

The most important factor of confirmation of competences is forcing students to come to university laboratory in person to perform class tests to show that it has been them who has been solving all the tasks.

We can distinguish by IP address if student works on campus labs or outside it. As shown in Figure 4.18, two distinct periods are visible – before and after CBL. Before CBL and changes applied with implementing this method, it was possible to complete all the work outside university as no compulsory confirmations were required. It was very popular option with almost 50% of students doing most of the work without ever coming to on campus lab. That showed that they felt confident enough to work on their own and did not required assistance or they used possibility of learn from distance to its fullest.

After CBL was introduced with required class tests, there are around 10% of students who do more than 90% of all work outside the campus meaning that they do some class tests and are able to complete the rest of the course outside campus.

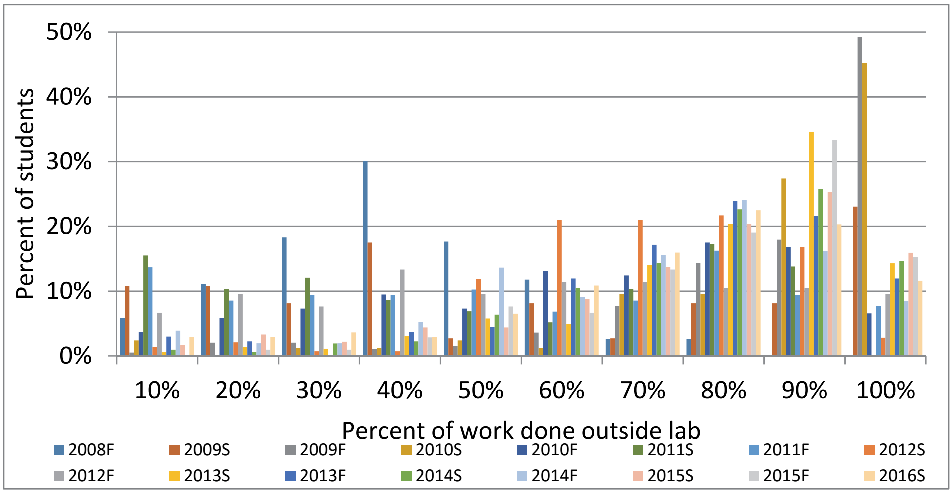


Figure 4.18: Percentage of learning done outside campus by semester 2008F-2016S

When before compulsory confirmations, doing 91-100% of work outside campus was most popular, then now it has shifted more left. Up to 80% of work done outside campus has had stable 20% of students from year to year. Up to 90% of work done outside the campus has more lows and peaks but it has been on the rise lately. Overall, it can be seen that a majority of the students still try to do as much outside campus as possible and attend class tests on site.

When comparing the number of times class tests and laboratories are offered, Figure 4.19, the number has remained quite constant since 2011 spring. It is visible that since CBL the number of lab and test slots offered has risen. It is also noticeable that the number of labs and tests offered are almost same, tests being offered slightly more often for majority of semesters.

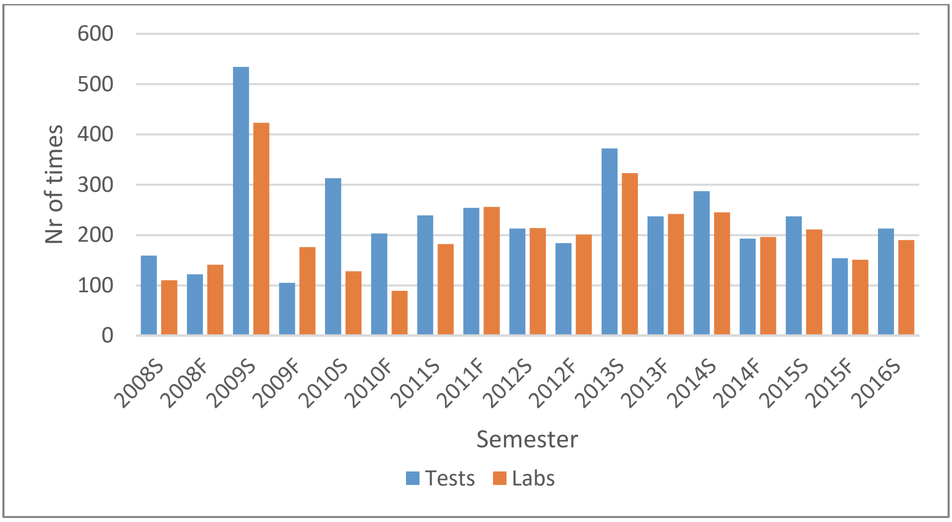


Figure 4.19: Number of times tests and labs have been offered on campus by semester

It can also be seen from Figure 4.19 that "tests" have been offered before CBL. Before CBL the tests were similar to current labs where students were able to solve tasks on site with teaching staff present to answer questions. Tests, then, contained only theoretical tasks without student needing the kit. Lab required students to do experiments with a kit. Labs now contain both theoretical tasks and experiments with onsite help. Tests from 2010 fall onwards are used to confirm competences.

When looking at the utility rate, a different picture emerges as can be seen from Figure 4.20. Until 2012 spring semester laboratories have had very high utility rate, especially for spring semester. As there are 15 slots in class, for some semesters the rate has been almost maximum or very close to it.

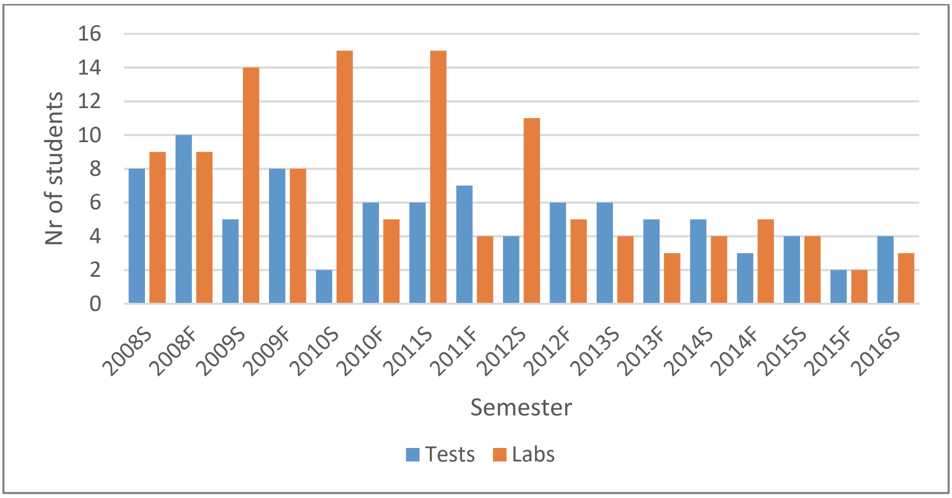


Figure 4.20: On site utility rate for tests and labs

From 2013, the utility rate of laboratories have dropped a lot. In some semesters, the average number of students per timeslot offered is only 2-4 students. That gives the teaching staff plenty of time for offering personal assistance to students but the most optimal utility rate, found by teaching staffs' experience, is 10 students.

Utility rate for tests have never been as high for labs but they have also dropped. Considering that all students should attend class tests, this trend is troublesome, especially considering that the number of times tests are offered has remained almost constant.

When looking how many times students who finish the course need to attend class tests and how many times they choose to come to the lab on average, the results are varying as can be seen from Figure 4.21.

The number of times tests are done on average has dropped in last few semesters compared to the semesters when CBL was introduced. It is positive that students can finish course with fewer class tests. Due to the fact that students have started to take lower grades, changes in number of tests done could also show a trend where the average result per test might actually have not risen. Therefore a line reflecting the number of class tests needed for single grade unit is added to Figure 4.22. It shows that since 2014 spring, the number of tests done per grade unit is smaller and therefore the results during class tests have to be better than before.

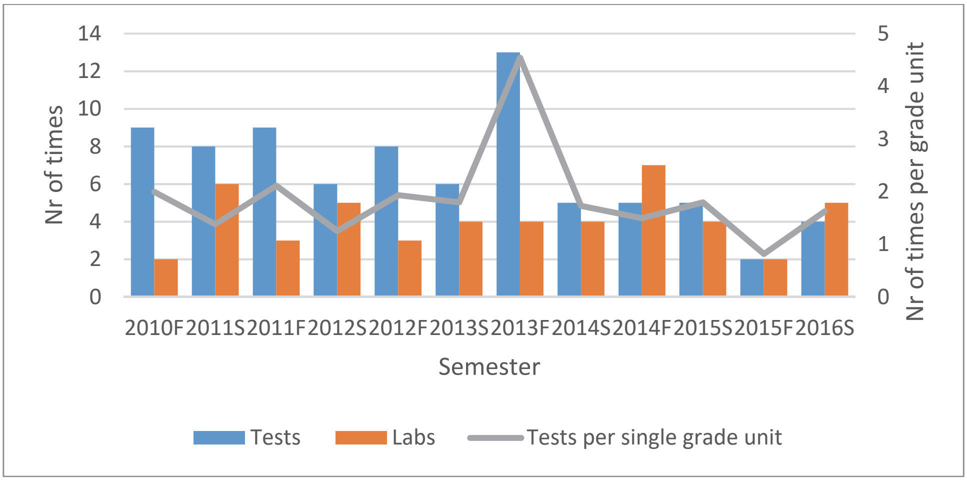


Figure 4.21: Number of times tests and labs are done by students on average and average by grade unit

As average results for experiments and theoretical tasks done during class tests have risen, as can be seen from Figure 4.8, it can be concluded that the drop in class tests is partly caused by the fact that students do not need to complete as many class tests to confirm their competences as they used to. Another factor may be that as the new method has been used for some years, they have gotten used to it and understand that they should prepare numerous competences to confirm before they attend class tests. This behaviour has also been enforced by changes in the system during the last year when students were only able to register for class tests if there were enough credit points available for confirmation.

4.6.4 Laboratory Experiments

Laboratory experiments are important factor in many courses but it has been the observation of the teaching staff that students try to avoid them. Figure 4.22 shows how CBL has changed the students' behaviour. Before CBL, around half of the exercises solved by the students were experiments. After CBL it has dropped a lot to around only 10% of the course.

It could be argued that with CBL more exercises are solved and therefore drop in lab percentage is expected. When looking at just an average number of lab experiments done, in topic-based learning it was around 300 experiments per semester. With CBL, in 2010 fall, the number actually rose to around 500. The percentage drops but students do small experiments more frequently. After that the undesired behaviour starts (with few exceptions) where number of experiments done by students during the semester drops around 200. On average it is just around 150 experiments per semesters. That is less than before when experiments where larger and took longer to carry out. Therefore, it could be said, that data reflects teaching staffs' observations.

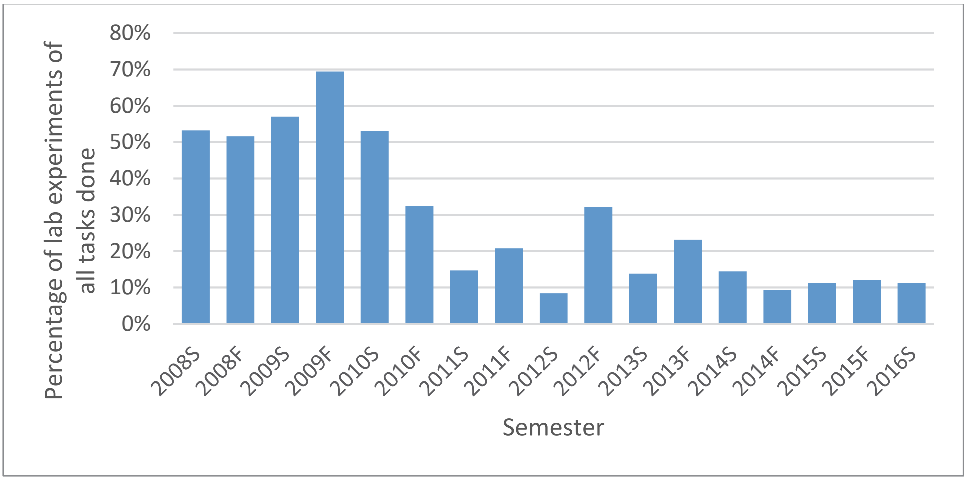


Figure 4.22: Percentage of lab experiments done from all tasks by semester

There are some possible explanations caused by the learning system design. When student borrows a kit, he/she can switch kit on and off for his/her current session so that he/she could work even if the kit is not next to them. If kit is not connected to the learning session, only theoretical tasks are offered. This should not be changed as students should be able to learn anytime and anyplace.

The second possibility is that lab experiments usually require higher level of competences and not all students reach to those levels, therefore the total number is also lower. This could be altered by adding very simple and basic experiments that are available even in low difficulty levels.

Third factor is students' willingness to attend laboratories on site or borrow HLKs. When looking at borrowing home lab kit, as can be seen in Figure 4.23, that is essential part of doing lab experiments outside of campus laboratory, the average lending time was very varying before CBL – from average 5 days on 2009 fall semester to 18 days on 2008 spring semester. What those periods have in common is the students' behaviour when it comes to returning the kit. Majority of them return it before the due date or day later. The delay is usually a day caused by not coming to campus that day or simply forgetting. Getting students to return the kits has not been a problem. In only a few cases have we been forced to track down the student after longer time period and ask them to return the kit [103].

With CBL, the average lending time becomes longer, averaging between 5 and 10 days depending on semester. The lending types also change – more students now borrow kit just for a day – probably so that they could work whole day with it on campus and not carry it back home. They do not register for a lab where they get kit automatically. Therefore, it shows that students want to work longer than 1-1.5 hour that we usually offer as a one lab cycle.

As can be seen from Figure 4.23, borrowing for 3 days has been the most popular option before CBL and has become even more popular with CBL. Borrowing for a week or longer than 3 days has dropped. On the other hand, lending for longer times (usually long-distance learners) has risen.

The main characteristic of learners with CBL is that they are not returning the kit on time. In some cases, they hold the kit for an additional week after the due date. With every new semester, the period students tend to go over the due date has risen. It is not their wish to use it more – they are just lazy to return it as system does not let you work with overdue kit. They could ask teaching staff to extend the deadline but they do not do that either.

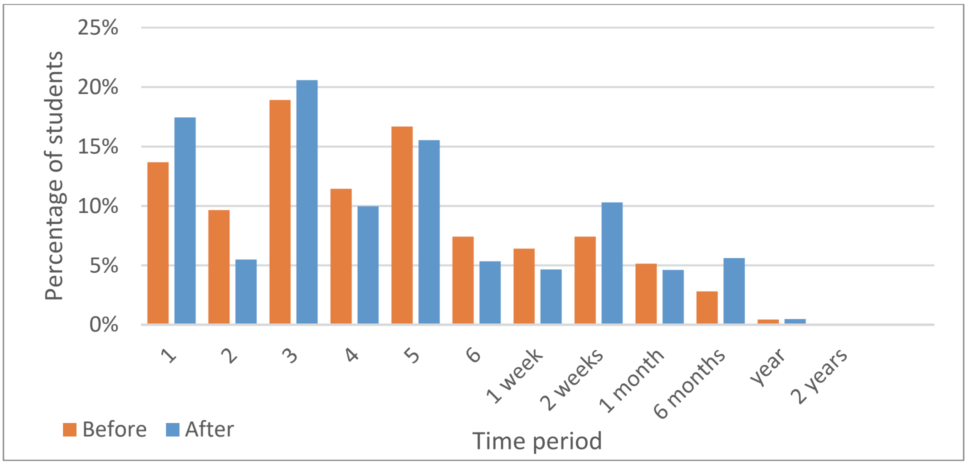


Figure 4.23: Intended HLK lending time indicated by the student when booking a kit before and after CBL 2008S-2016S

It is interesting to compare Figure 4.23 with Figure 13 in [103] calculated at the end of 2012. The patterns at the first five semesters for short term lending seem to be very different from the average now. Then 5 days was the most popular option by far and 4.11 was average for full-time students. Very few students borrowed the kit just for a day. The long term lending, on the other hand, looks very similar.

The number of times student borrows a kit has dropped over time as can be seen from Figure 4.24. More students lend the kit only once or twice during the semester. On the other hand, there are students who lend kits tens of times during a semester.

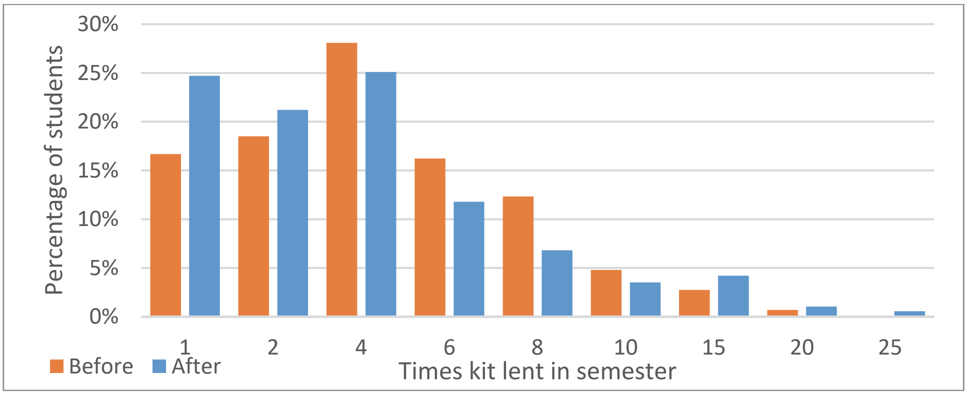


Figure 4.24: Times kits have been lent by the student 2008S-2016S

Before CBL, as can be seen in Figure 4.25, on average a kit was borrowed 4.30 times equalling to 45 days. During the second behaviour pattern in grade taking, as can be seen in Figure 4.25, the average time kit is lent is quite even from semester to semester averaging to 3.56 times equalling to 34 days. The third behaviour shows much unevenness but the average lending number drops even further – to 2.78 times. This corresponds to 35 days, almost same as during second grading behaviour.

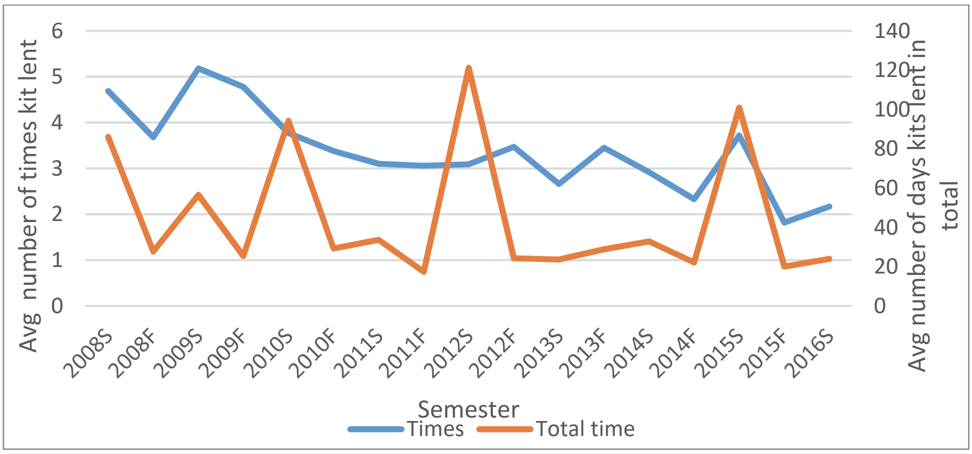


Figure 4.25: Average number of times kit lent and corresponding average number of days by semester 2008S-2016S

The total number of days kits are lent in total in a semester is quite varying. Fall semesters seem to have quite an equal average of days throughout the years, especially after CBL. Spring semesters, on the other hand, have irregular peaks. This can be explained that courses for distant learners are usually taught in spring and distant learners borrow kits for longer time periods.

When looking at students' behaviour when they have borrowed a kit, it is found that most of the time, they still do theoretical tasks as can be seen from Figure 4.26. Only 7% of students spent all the time solving the laboratory experiments when they borrow the kit. It would be expected that they do some theoretical tasks on the same learning session but the percentages of experiments are low. When comparing time during HLKs borrowing spent on experiments and on theoretical tasks, there is slight shift to right but nothing remarkable meaning that even when students have kits, they are not very willing to conduct practical experiments. On the other hand, it could be argued that some students just follow the natural learning process where they get experiments mixed with theoretical tasks and therefore the lab percentage remains low. That kind of attitude for learning would assume that students borrow kit often and have it around almost all the time when they learn with the system. As seen previously, it is not the case.

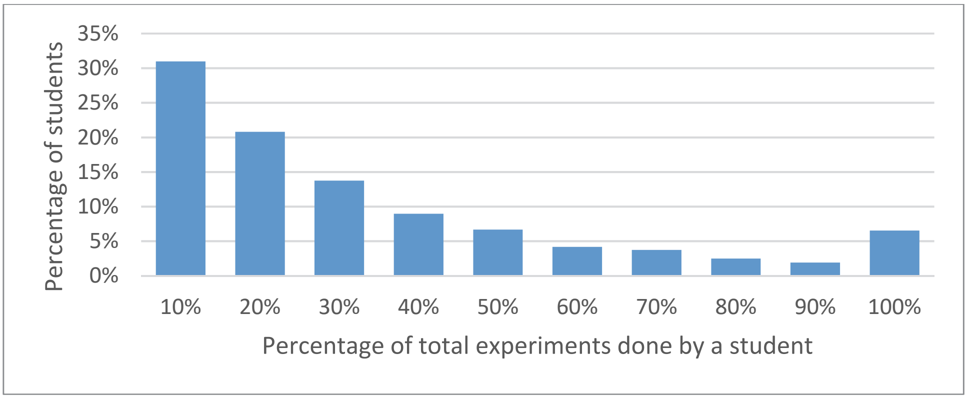


Figure 4.26: Percentage of lab experiments done while students borrows a HLK

4.7 Answerer Evaluation

Before improved answer evaluation mimicked students' answering process, a lot of wrong answers were due to typos. Typos were removed by adding some pre-submission controls (from simple verification, if number and making sure all fields are filled, to task specific requirements checking).

With typos being prevented and lowering repeated mistakes, 71% of students would have got better results than before. In 27% the result would have remained the same and in only 2% the new algorithm graded answer with lower result value [147]. This proved that generated correct answers and evaluating them based on competences is more efficient in reflecting student's real progress.

At first, the algorithms mimicking students' answering process were introduced in Operating System course, taught in spring semester. That course was chosen as the number of students taking it every year is large enough but it is not one of the biggest taught with the system. Also, that course contained many complex exercises where students felt that if they get the wrong answer, they do not understand what they did wrong [146].

Those exercises used in Operation System course that were transformed into algorithms mimicking students' answering process are called "test set" for current analysis. On the same time, when the first patch of algorithms was tested, there were other tasks in other courses that were used at the same time. Those tasks are called "others" to verify the effectiveness of changes and the effectiveness of proposed evaluation algorithms mimicking students' answering process.

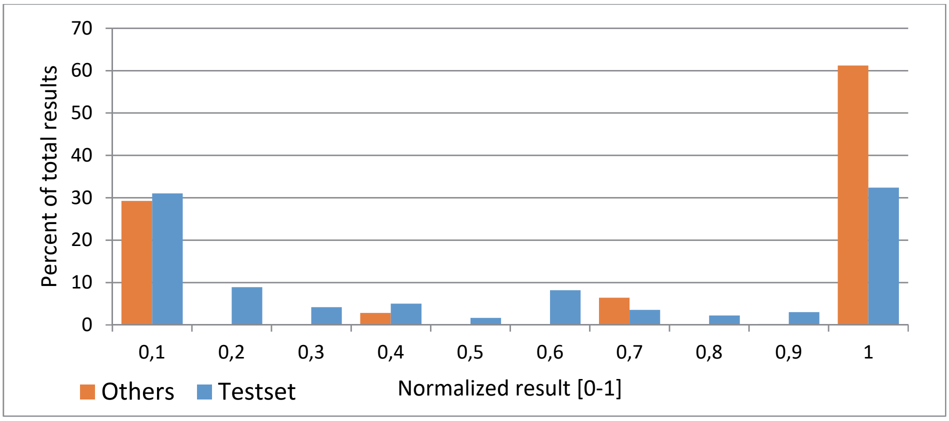


Figure 4.27: Distribution of results before CBL (2008S-2010S) [146]

At first, situation before CBL is considered. Figure 4.27 shows that before CBL, exercises in the test set were half less likely to be graded totally correct than other tasks. Noticeably, they have almost similar percentage of totally wrong answers. Therefore, the difference comes from evaluations between totally wrong and totally correct. Middle values were more often used in test set than in other tasks. Figure 4.27 also proves that students had a hard time with exercises in Operating system course.

When CBL was introduced, all the competences in the tasks were graded separately and this gave the students a better overview of what they got right and where the mistakes were made, even though no improvements for written feedback were implemented at first.

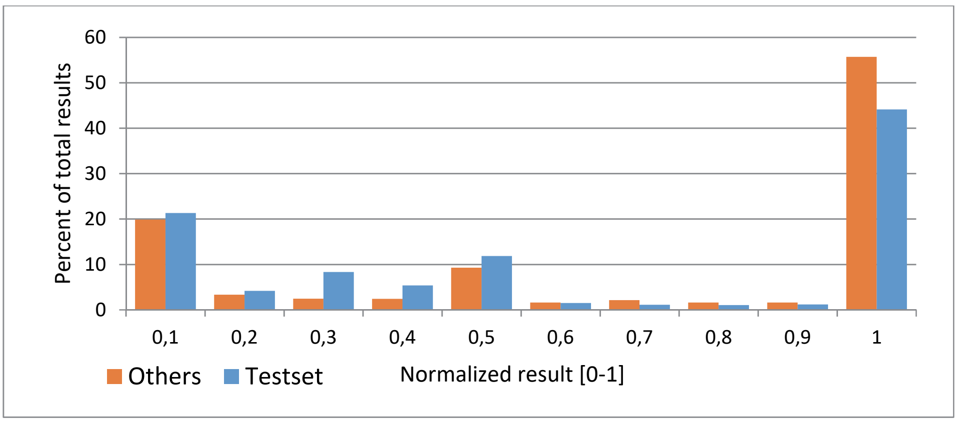


Figure 4.28: Distribution of results with CBL (2011S) [146]

Figure 4.28 shows changes in distribution of results with CBL but without algorithms mimicking students' answering process. It can be observed that totally wrong answers have dropped around 10%, again a positive sign. The percentage of totally correct answers has dropped slightly for other tasks but exercises in focus, test set, has seen remarkable rise. Also, much higher variety of results between wrong and right is now used.

Then, algorithms mimicking students' answering process were introduced for the tasks in test set. Those algorithms will follow the pattern of Figure 3.3. The other set continues using simple competence separated grading in the same time. Results can be seen in [146].

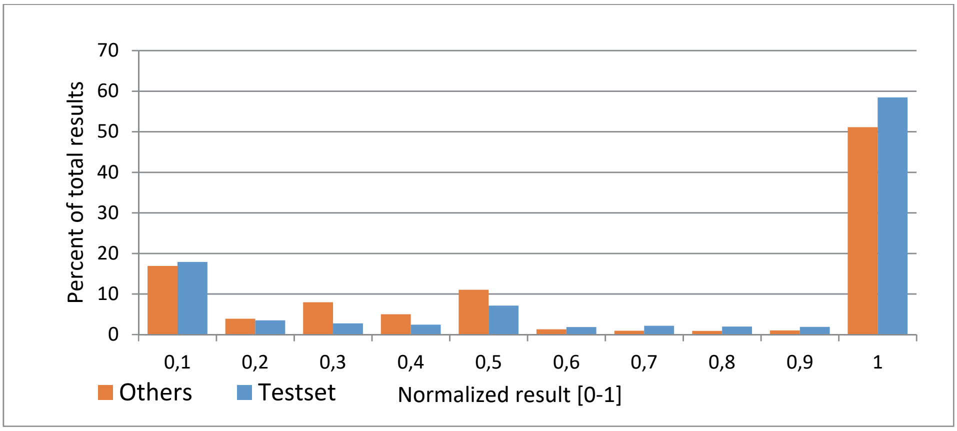


Figure 4.29: Distribution of results with algorithms mimicking students' answering process for testset (2012S,2013S) [146]

With evaluation algorithm mimicking answering process, as can be seen from Figure 4.29, first time during the analysis, the test set has higher percentage of totally correct answers than others. Others, for the timeframe used to analyse algorithm mimicking students' answering process has dropped about 5%, compared to Figure 4.28. The rise for test set is remarkable and therefore it would be deduced that granting student a detailed feedback about their mistakes helps them to learn from their errors and give them better probability to gain better result next time when solving exercise with similar competences.

Therefore, it can be seen how CBL with algorithms mimicking students' answering process have benefited a set of tasks in Operating System that were considered the hardest and had noticeably lower results before. With new algorithms they have managed to achieve a higher percentage of correct answers than other tasks solved in the system at the same time.

4.8 Conclusions

In this chapter usage of competences, students' study behaviour and other aspects of CBL were analysed in detail from e-learning system logs from 2008 spring to

2016 spring to verify the usefulness of proposed methodology using educational data mining principles.

The main conclusions of this chapter are:

1. Main goals on CBL has been achieved. Those were the repetition of tasks (higher number of submission), spacing (memory model) and better acquisition of knowledge (higher average result).
2. CBL is more beneficial for students in achieving high level concrete competences.
3. Due to splitting the tasks into smaller units it was possible to achieve the repetition without too drastic change in learning time. It was found that learning time is now around twice as before CBL but more in line with ECUs connected to the courses.
4. Adequate number of different tasks are required to avoid reappearance of same exercises. The bare minimum of competence coverage in 128-level scale that has been divided to 8 regions is 40 tasks (5 per region). Currently, adequate competence coverage has been achieved.
5. Three distinct student behaviour periods in covered time frame were identified. First is the period where topic-based learning was used. Second period is the start of CBL from 2010 fall to 2013 spring where students were trying to achieve high results and low marks were almost never taken. Third period starts from 2013 fall and is mainly caused by students' attitude due to changes in university policies where mark has to be achieved with one semester. Now students are working more intensively but do not continue with their courses to achieve high grades. Low grades and avoiding practical experiments are also common for the third period. Also, it has been found that students now tend to work later both during the day and during the evening than they used to. Also, the clear peaks during class times have lowered and students tend to visit labs and class tests less than before.
6. Problem of high-dropouts was identified. CBL with self-regulated learning suffers from unpredictable workload. The amount of work needed to be done by students differs a lot and the reason why many students fail the course is because at the end of semester they run out of time.

5 PROGRESS VISUALISATION

The analysis showed that better average results and repetition were achieved with CBL but also some problematic behaviours were discovered – noticeably the fact that large a portion of students did not pass the course or that they were willing to accept low grades instead of trying to achieve a better mark.

When looking into those problematic areas, it was found that many students do not pass the course or accept a lower grade than they wish due to time running out at the end of semester. Students have also complained to teaching staff that they are having a hard time understanding how much work they still have to do as the learning process throughout the semester is not linear – there is no specific number of tasks a student has to do. Using classical topic-based approach, it is relatively easy for a student to keep track on his/her progress - there were only a certain number of tests/tasks that had to be completed once in order to finish the course. Even with self-regulated learning, students were quite adequate at planning their own time and finishing before the deadline [63, 162].

That kind of straight forward approach is not possible in CBL. CBL requires repeating similar tasks over and over again to make sure the knowledge moves from short term memory to long term memory. Therefore, the time is not linearly predictable. Some students work very fast; others need much more time. It also depends on time between solving similar tasks, on forgetting model parameters, on how correctly student answers, on how often certain mistakes are made etc. The second problem is that when students are masters of their own time and the only deadline is at the end of the semester, they tend to leave everything to the last minute. As the system also embeds forgetting model, rushing at the end usually does not lead to pass [162].

It became obvious that when self-regulated learning worked sufficiently in classical topic-based approach, it became a source of many problems in the CBL. Therefore, different tools have been developed to help students to predict their progress, plan their time, motivate them and therefore help them to avoid running out of time at the end of the semester.

The learning in CBL is personal and grade is achieved using “grade suggestion engine”. It is therefore essential for learners to understand their current progress in the course which enables them to plan time accordingly. As the learning process is controlled by the student (by choosing which competence he/she wants to learn) it is vital that the learner has adequate data to be able to make the best decisions. Multiple new visualization tools were therefore developed for the learners by thesis author on 2014 fall semester.

This chapter is based on publication P5.

5.1 My Field

MyField, Figure 5.1, is the main hub of the learning in ISC system. It consists of small color-coded boxes. The colors indicate current (or future, depending on mode) competences' ability levels in 128 scale, as can be seen from Figure 5.2.

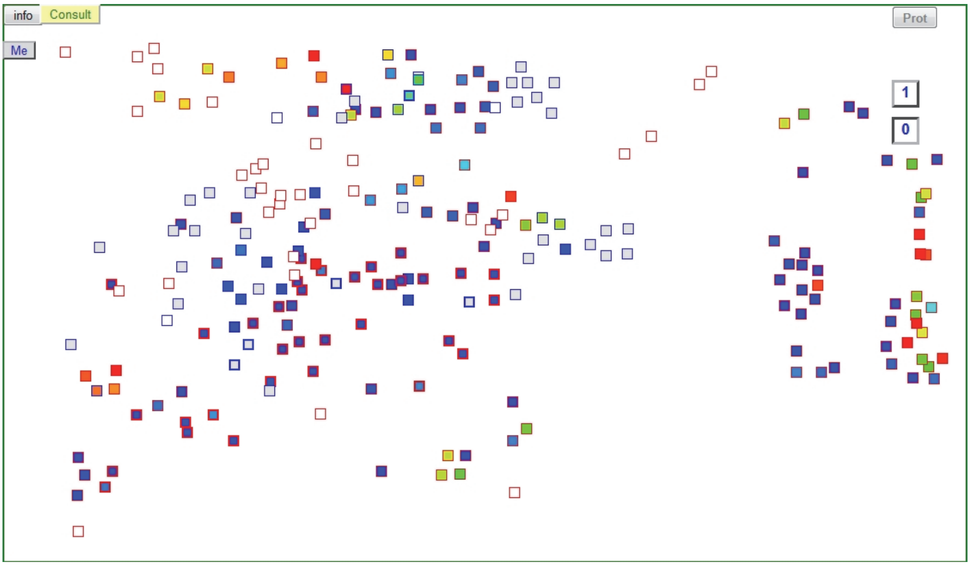


Figure 5.1: My Field view of students' current progress with competences [63]

During the learning process, students can choose a competence from the field they want to learn. Similar competences are gathered together with easier/main topics at the bottom and more advance topics on the top of the field. Every course has their own field with their own layout. If competence is chosen from the field, a proper task using learning control is found. After solving the task, student is redirected back to the MyField with changed colours for competences that were affected by the submission of the task.

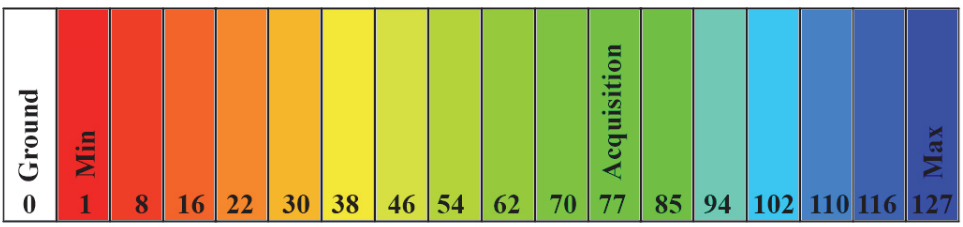


Figure 5.2: Colour codes used in learning process to indicate ability levels

Therefore, it could be said that MyField is not only the central hub of learning but it also doubles as the tool to follow the progress in the course as students can see how many of the competences in the course they have tried, how many of them are green or blue (high levels) and how many of the competences are still needed to be learned. Statistics show that students spend 93% of time when logged into the system on solving exercises, originating from MyField. It was the first progress following tool offered in the e-learning system when switching to competence-based approach, even though progress visualization problem was not acknowledged at that time [63].

From literature, similar central learning hubs can be found by Wang and others [93]. In their case, the competences are connected with a graph and the learner has a more pre-defined path. A learner cannot start learning other competences until the previous graph nodes have been completed. Therefore, our approach offers more freedom and more personalization.

5.2 Visual Grade-Suggestion Engine

MyField may use color-codes to indicate student progress with competences but it does not give a visual representation of progress in any given course (how much is done; how much work is still needed). Grade suggestion functionality and its visualization do that. Data from the function can be visualized so that student could follow precisely how much work needs to be done before the desired grade becomes available as seen on Figure 5.3. On x axis, credit units for the selected course are shown. On y axis competence states that are equal or higher than acquisition level are represented [63].

Possible grade areas on 2D plot are shown using formula 3.4 that takes into account the current university guideline that a student should be able to pass after acquiring 51% of course material. To construct the lines that are used to plot different grade areas, this guideline and the point where student completes 100% of course with minimal level are used using formula 3.5. 10 level steps are used to define lines for grades [63].

Yellow line on the graph shows levels of confirmed competences starting with the highest levels. The length of the line depends on the weight of the competences for the course. Black slim line is used to show progress with all the competences (including unconfirmed ones) [63]. The black line is showed visually to motivate students to enrol in class tests and to show what grade they could currently take if they would have all the competences confirmed.

When yellow line reaches any of the grade areas, grade-suggestion engine is able to suggest that grade for a student [63].

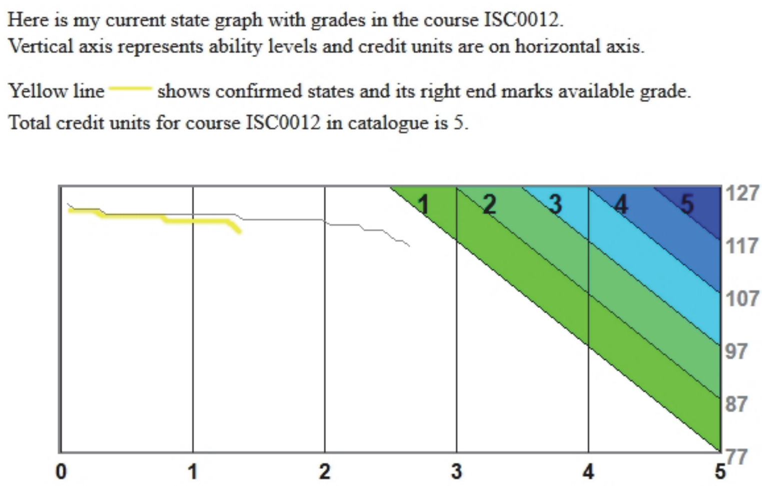


Figure 5.3: Grade-suggestion engine visualization tool [63]

5.3 Radar Chart

Lately it has also become popular to represent competence levels with radar style graphs (for example, [163]). Thus, a radar chart with all the competences required for the course with their current and acquired level has been offered for students so they could follow their progress as can be seen in Figure 5.4.

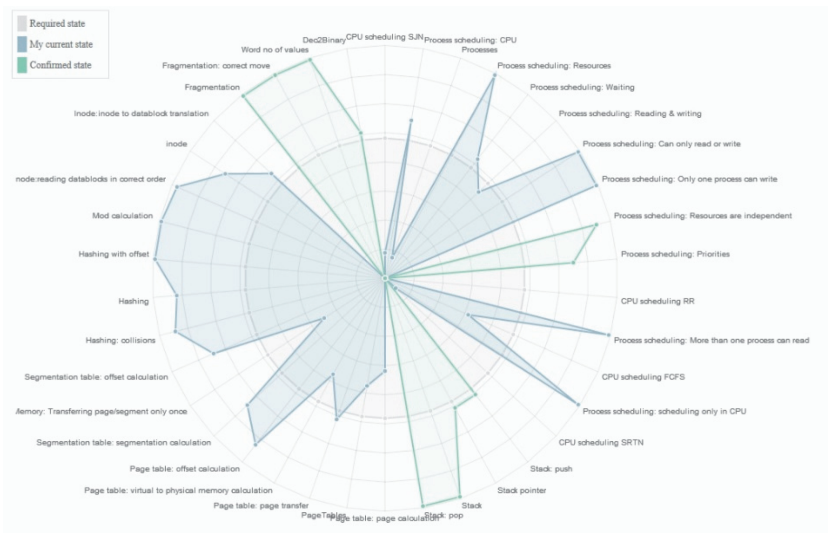


Figure 5.4: Radar chart of competences for Operating System course [63]

Radar chart shows student's current level of confirmed and unconfirmed competences. Acquisition level is shown as a grey circle. Similar competences appearing frequently together are positioned in adjacent sectors [63].

To enable students to have an adequate overview of their competences, the positioning of competences on the radar is very important. Similar competences have to be close by. There are multiple sources of information where "similar" could be deduced. The most natural source would be My Field where similar competences are positioned together on 2D plot area. As radar chart can be considered to be a line where both the start and the end of the line are connected, information from 2D plot area cannot be used as straight forwardly on 1D line on radar chart.

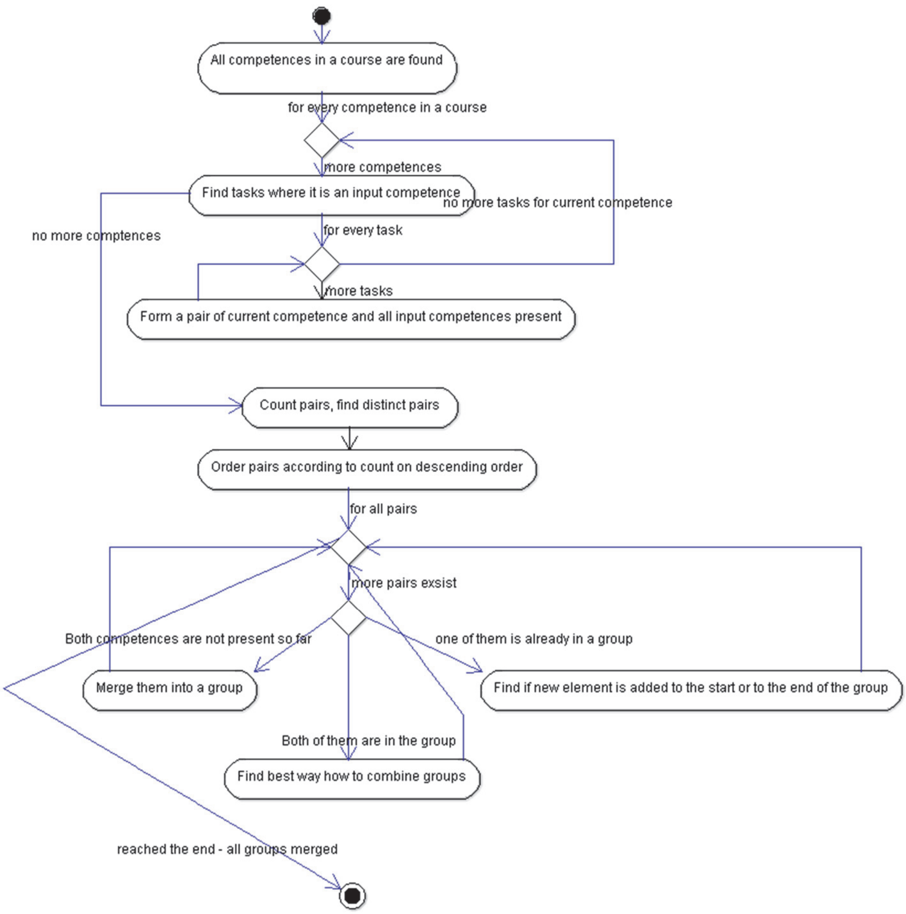


Figure 5.5: Process of finding line of competences for a course

Furthermore, the desired position for competences on the radar chart would be as follows: it is expected that competence is surrounded from both right and left by two of the competences it appears most often together in exercises. Therefore, instead of using the same data that is used for MyField, a new methodology to find closely connected competences was derived [63].

The algorithm first collects all the competences that have a weight for the course. Then, for all the competences, tasks where they are included as input competences are found. Pairs are then formed of current competence and all the other competences that are also input competences for those tasks. Some of those pairs may appear multiple times. Then the list is scaled down to distinct pairs and the count of how many times they appear together is added to the pair. That process is repeated for all the competences and for all the tasks they appear in. Then, by using ideas of maximal spanning tree and travelling salesman methods, a line according to similarities is formed of all the competences that will be used for the radar chart. This process can be seen in Figure 5.5. The code for forming the line can be found in Appendix 1.

5.4 Interactive Data Viewer

Another visual tool offered for students is dashboard style interactive data viewer as can be seen from Figure 5.6. Similar dashboard has been offered to teachers for years now to follow the progress of solo students. For student view of the dashboard, some of the graphs were replaced or modified to hide system sensitive information (task id-s, correct answers etc.). Some new graphs were also added according to the specific needs of the students [63].

The dashboard offers many different graphs for a student to interact with. All the graphs on the dashboard are interlinked (or cross-linked) meaning that a selection in one graph will force filtering on all the other graphs. For example, a student can see the time he/she spends on solving the task or which hours of the day he/she works as a bar chart. The number of exercises done every day is offered as a timeline. Competences that student has solved and, if learning actions have been part of class test or not, are shown as a pie chart. The data table with last 30 submissions is also offered [63].

Interactive data viewer has been designed using D3 JavaScript library [164] and data driven approach for documents. Different graphs were linked together using Crossfilter JavaScript library [165].

This tool enables students to follow their overall progress or use the interactive tools to limit the data in order to draw more specific conclusions. The interactivity of the data viewer gives students freedom to "play" with the data. At first, the students may not realize the benefits of using filters to limit the information they are looking at, but it has been observed that the more they use the tool, the more they start to comprehend it and find its benefits [63].

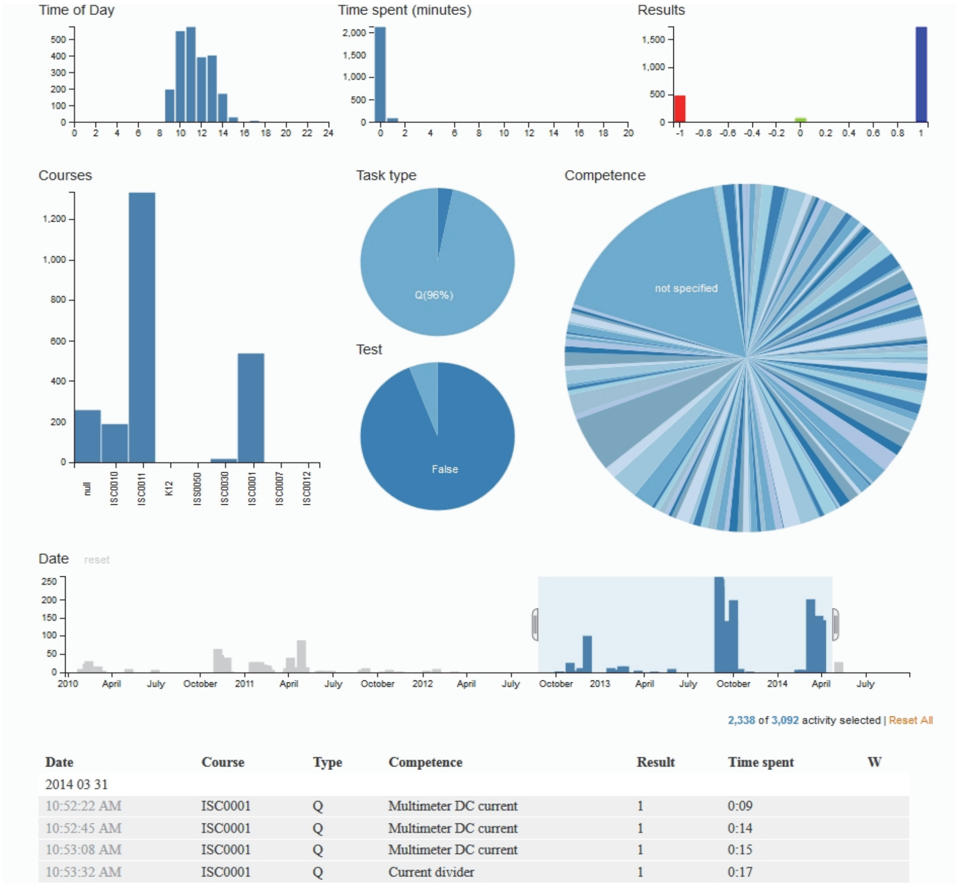


Figure 5.6: Interactive data-viewer with multiply interlinked graphs enabling students' selection to redraw all the other views [63]

5.5 History Map

Taking initial feedback into account from the launch of visualization tools, another view for data was designed. History map, Figure 5.7, has been designed to enable students to playback their competence state changes over the last four months [63].

Students can select any course they have declared and their original state from four months ago is shown. They are given information on how many competence changes have been recorded on that timeframe. In case they have not been active on the course for four months, the timeline is shown from their first action until the current date and therefore there is also information about the resolution of the timeline in the history map [63].

History map loads student state into the course at the start of the timeframe. Chosen course's MyField layout is used as a base for the view. Then student has an option to start and pause the playback. Playback consists of changing competence box colours according to the state changes. Student can hover over the colourful boxes representing competences to see what elementary skill is behind any specific box [63].

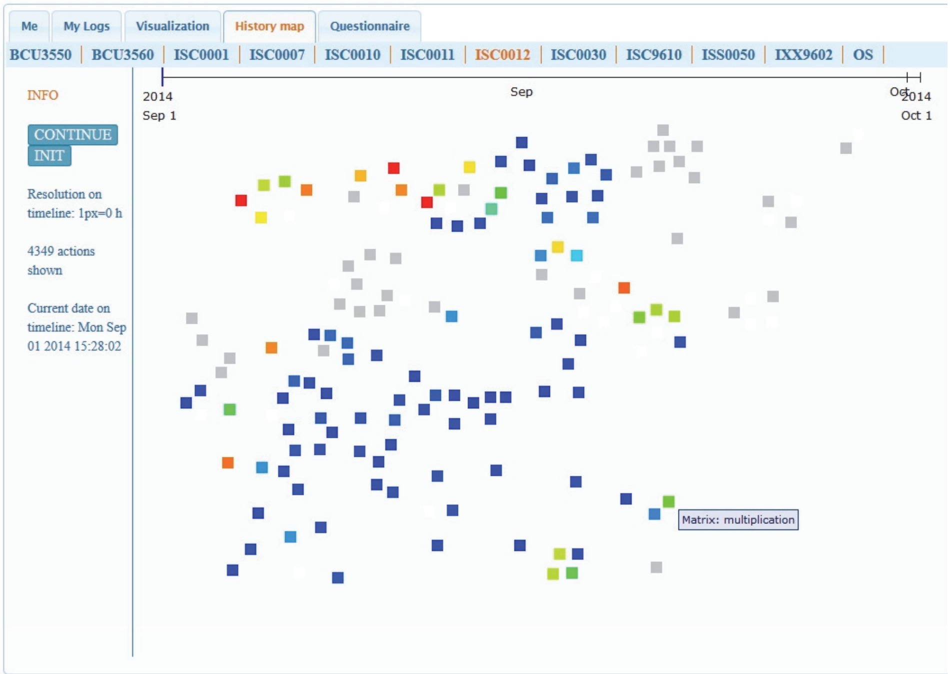


Figure 5.7: History map that enables students to replay their actions over last 4 months for selected course [63]

Four months was chosen for the default timeframe as it equals with one semester. If all the data would have shown, for some students it would have meant that tens of thousands of actions would have been displayed. It would make the playback too long, and it was thought that the earlier data might not be relevant anymore for learners [146].

5.6 Analysis

In order to validate if visual tools helped students to comprehend the data and abled them to predict their time in the course, questionnaires, both in Estonian and English, were added to the ISC e-learning environment when new features were introduced. The questionnaire used 1-5 (1 lowest, 5 highest) scale to grade how easy it is to use the tool, how much the tool helps to follow the progress and does the tool have all the desired information.

Students had time to try the features and they were encouraged to fill the form. On 2015 spring, when feature was introduced, around 12% of students filled the feedback form – 2/3 in Estonian and 1/3 in English. Feedback originating from the questionnaire and from verbal conversations with students were used to draw the following conclusions.

5.6.1 Tools

MyField is a tool that students have used for learning since switching to CBL. Students have said that they familiarize themselves quickly with MyField at the start of the course. They state that they can check their progress between exercises without having to click or go to some other part of the system. Instant access to information has been brought out as a very important factor for progression tools. Also, some have thought that current progress might not be so easily grasped with MyField [63].

Simplicity, familiarity and constant automatic access as a side product of choosing new tasks, seem to yield major benefits for students and therefore, even if more detailed and specific views are offered, some students prefer to stick with the old and familiar concepts [63].

Interactive dashboard catches students with its interactivity. It has been stated by students that they first start with just playing with different views to familiarize themselves with the environment. When they have understood how the views are offered and linked with each other, they start to look for more specific information. This view enables them to check their results during class tests so they can see what competences they got right and which ones resulted with mistakes. According to students it is a major benefit for them as they were not able to access class test results before. No results/feedback is generated for the students during class test as it might discourage and disturb them [63].

This view also offers them the access to other data they had no previous access to – for example, detailed personal logs of their activity. In addition, it enables them to get statistics on what competence they are most likely to answer wrongly. Also on what competence they have solved most often and which the least. It opens up many possibilities and gives students a chance to analyse and modify their learning behaviour accordingly. This is the new visualization tool that students have given most positive feedback to and stated that it is highly useful for them [63].

On the other hand, some students have stated that interactive dashboard is hard to understand. As this tool expects students to take active lead on what they want to know, some are modest to do so. Those students have stated that it is hard for them to understand what those different views offer, even if they have titles. The cross filter functionality also seems to confuse them [63].

Radar chart assembles all the competences in the course together on to the same chart to give simple, fast visual overview on how much work is done and which of the competences have been confirmed. Even though radar chart seems to accomplish this goal, the view does not take into account that different competences have different weights for the courses. Students have pointed out that in general the chart gives them an overview but it does not make the planning accurate because of the missing competence’s weights. It has been suggested that this factor should also be somehow represented on the chart [63].

Grade suggestion engine has been praised by the students for showing progress on the course as a line that makes it very easy to grasp and leaves no room for misunderstandings. Representing grades as areas on 2D plot has also been well received and this chart offers them an easy way to see where they are in the course and how far are their goals. They have also stated that the little grey line which is giving the result if all current competences were to be confirmed, has a positive motivating force. They see that they may not be as far as they thought and are more eager to enrol for on-site tests. On the other hand, the problem that competence weights are not easily visible on this type of chart has been brought out. The fact that there is nothing interactive, also seems to bother the students even though when asked what they would like to see, they are not able to suggest anything specific [63].

History map was launched in the system later than other visualization tools, therefore this tool has slightly less feedback from the students. Those students who have filled out the feedback about this tool, have stated that they enjoy replaying their progress – sometimes only to see the progress they had made with weeks of learning. They have brought out that this tool motivates them to work harder as they see, like in a movie, where they have come from and how far in the course they have made. On the other hand, students have stated that this tool has an entertaining value but they have also stated that they prefer not to follow their progress only with this tool. It does give them an overview but they state that this tool lacks details which they can see with other tools. Students have also said that it is easy to understand as it uses the same "MyField" layout that is very familiar to them [63].

5.6.2 Usage

81% of students, who filled the feedback, state that they have benefited from the progress visualization tools. International students who filled the feedback in English had more positive attitude towards the tool with only 10% stating that they have not benefited from the tools. The preferred visual tool from all the tools was MyField with Dashboard in second place as can be seen from Figure 5.8.

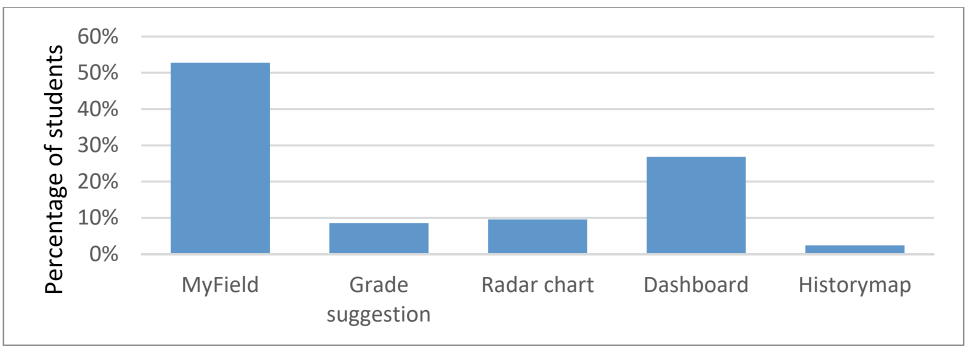


Figure 5.8: Preferred visualization tools by the students

Students who filled out the feedback, stated that radar chart was the tool easiest to grasp and understand as can be seen from Figure 5.9. It should also be noted that all the tools have quite similar averages with dashboard slowest, matching the conclusions drawn from the interviews with students.

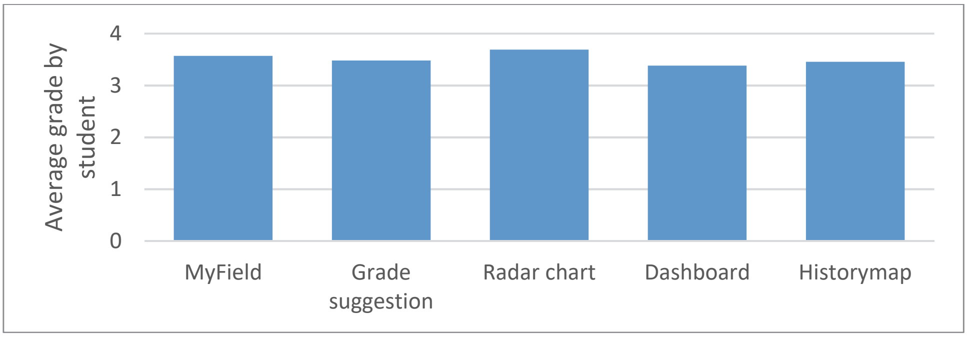


Figure 5.9: Average grade for how easy it is to use visual tools on scale 1-5

Grade suggestion engine was brought out as the tool that has most information that students desired as can be seen from Figure 5.10. Dashboard, surprisingly, is the tool with lowest grade despite having most varying set of information from all tools. That could be explained by the fact that many students just do not understand how to use the crossfilter functionality or they prefer simple pre-formed views instead of searching for answers by themselves.

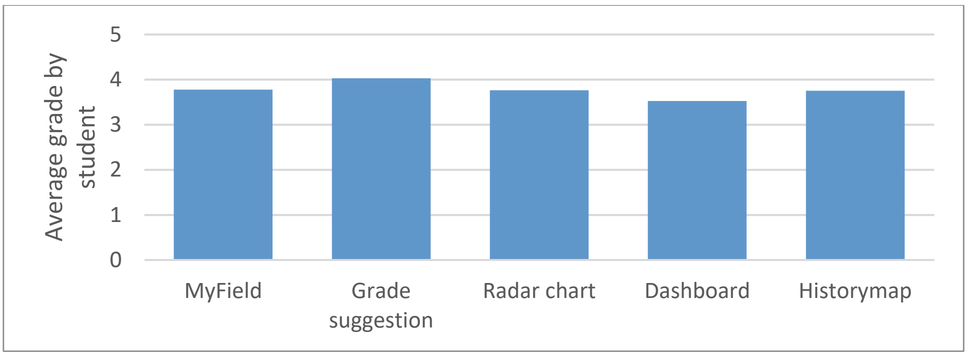


Figure 5.10: Average grade for how much of all the required information is included in a tool on scale 1-5

It was also asked how much the tools helped them to follow the progress. As can be seen from Figure 5.11, MyField has the highest average result with Grade suggestion engine in the second place. It shows that students prefer “old” and more familiar tools instead of new ones. They are used to work with already existing tools. Three new tools have similar averages without any one of them standing out too much.

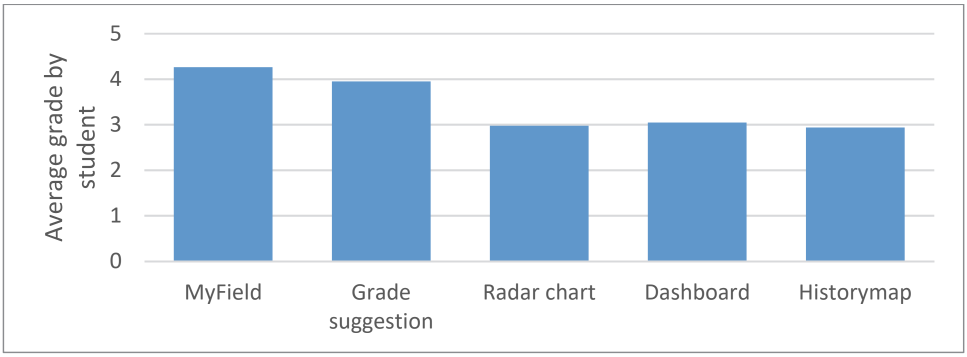


Figure 5.11: Average grade for how much a tool has helped students to follow their progress on scale 1-5

When looking at the frequency of students' usage of those visual tools, majority of students have used visual tools only once or twice or few times as can be seen from Figure 5.12. Around 20% of students use tools once a week or all the time. There has been a rise in frequency as [63] only 20% of students in total used tools once a week or all the time. This rise is expected as students have now had more time to get used to with the tools.

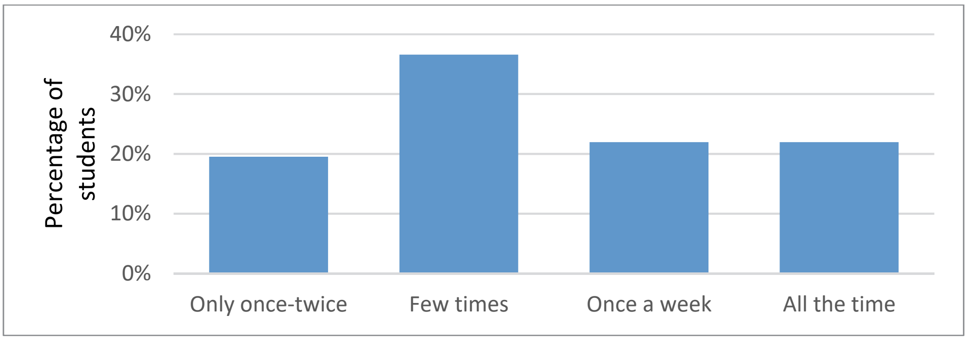


Figure 5.12: Frequency of using visual tools, stated by student

When looking at the logs from the system, the situation is different as can be seen from Figure 5.13. Students are overestimating their usage. Only three new visual tools are considered. Students who have declared the course during a semester are considered. Students who are categorized as once or twice have accessed visual tools 1 or 2 times during the semester. Those who are categorized as few have accessed tools on average less than 0.8 times. Those who are categorized more than once a week have weekly average at 1.3. Those in the middle are categorized to be using tools every week.

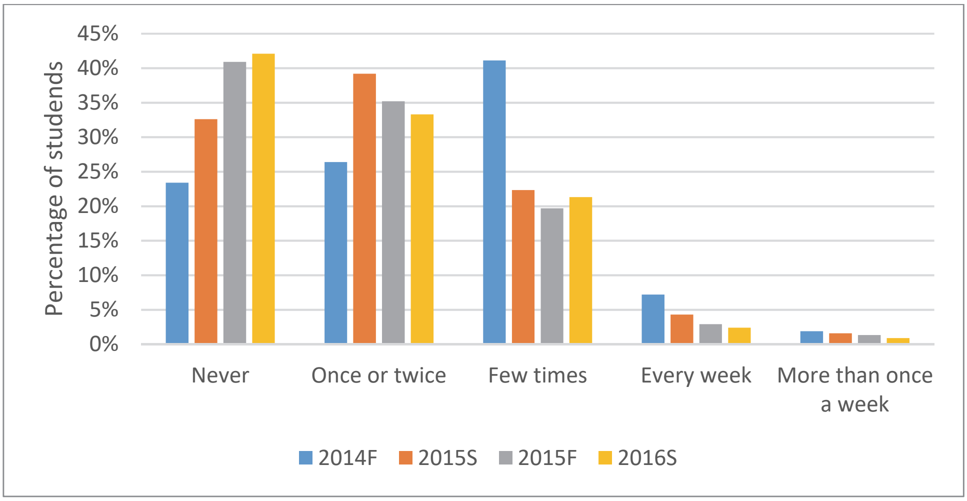


Figure 5.13: Students usage of visual tools by semesters according to logs

The percentage of students who use new visual tools is actually quite low and has been dropping in every consecutive semester. The percentages can be explained with the fact that at first two semesters when tools were added to the system, it was often reminded to students to try those tools by the teaching staff. The differences between qualitative and quantitative results could also be explained with those students who answered the feedback form, all had tried the tools and probably used them more frequently than others or had some opinions about the tools.

Apriori frequency set algorithm [166] was applied to questionnaire data. Those who said that they used visualization tools frequently were also the ones who stated it was very useful for them. On the other hand, the students who said visual tools had not helped them, had different favourite tools and also their frequency of using those tools varied to almost never to every day. Students who voted MyField to be the best visual tool, also gave it the best scores. On the other hand, students who liked interactive dashboard best did not score it higher than all the other tools. As expected, the opinions of students vary a lot as everyone as their own personal learning style [63].

Overall, the feedback has shown that those students who use visual tools also appreciate them. Also, as we have seen with many new features, new students who had not used the system before the new features, were more eager to adapt to new tools. Students who had worked hours in the system tended to prefer older features, for example MyField over interactive dashboard [63].

It can be followed from Figure 4.4 that the percentage of students passing the course from 2014 fall has risen. There is sadly no adequate way to measure if offered visual tools have had any effect on the final results and passing. For spring semesters that can kind of pattern cannot be followed and therefore it could be assumed that visual tools did not have the desired strong effect.

Also, the fact that the usage of the visual tools has dropped when teaching staff is not constantly reminding students to use them, shows that visualization tools did not fulfil their goal. The conclusions drawn are that students do not want to have data and look for patterns and make conclusions by themselves. They prefer familiar, simple and concrete information.

5.7 Conclusion

On this chapter solution for the problem of high dropouts due to non-linear workload identified in analysis was considered using progress visualisation. Two different visualizations existed in the system and three new tools, radar chart, interactive data views and history map, were developed by the author. The usage of tools was analysed by collecting feedback from the learners and using data logged in the system.

The analysis showed that students preferred to use old tools that they were familiar and despite initial positive feedback the usage of the tools has dropped after active reminding by teaching staff stopped.

The main conclusions of this chapter are:

1. Visual process following tools do not lower the students fail rates.
2. Students prefer clear statements, not data to draw conclusions by themselves.
3. Students do not want to make an extra action to access the information.

6 PREDICTING STUDENTS BEHAVIOUR

As progress visualization did not yield the desired results in avoiding students rushing at the end of the semester and failing the course, another solution was developed. It considered students wanting answers instead of being shown data. The proposed solution by thesis author for those two problems is to predict students' final result throughout the semester using all the data that has been collected by the system from previous students.

In this chapter the process of prediction is described in detail. The chapter starts with explaining what symbolic data is and why it was chosen to represent students' behavioural data. Then, for histogram valued symbolic data, new dissimilarity measure is developed as generic measure for different bin numbers and bin widths did not existed before. Also, compactness of clusters for the new dissimilarity measure is defined. Then, the variables chose using principle component analysis are given and achieved clusters from clustering covered. During 2015 fall semester, the prediction model was tested on one of the major courses. Results of that test show that proposed prediction model works well for predicting final grade but is not very suitable for predicting students' finishing time.

This chapter is based on publication P6.

6.1 Methodology

Educational data mining is used to develop previous students' behaviour models. Regression analysis is a statistical technique for investigating and modelling relationships between variables [167]. It is widely used in prediction and in forecasting [158, 168, 169]. Regression analysis allows us to estimate how a set of variables is connected to the unknown variable. In current study, we are interested in knowing how known student behaviour (set of known variables) will result in finishing time and final grade (un-known variables).

As e-learning system logs all actions that a learner does in the system, there will be a lot of information (variables). Instead of using all of the possible information collected by the system, a set of linearly uncorrelated variables, so called principle components, can be found to describe the relationships without any significant loss of details. Therefore, instead of using all possible variables from the system, a set of most important variables could be found.

Principal component analysis (PCA) uses a vector space transformation to reduce the dimensionality of large data sets [170]. It is a statistical procedure using an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables [171]. This transformation is defined in such a way that the first principal component has the largest possible variance and each succeeding component has the highest variance possible under the constraint that it is orthogonal to the preceding components [171].

Furthermore, variables are usually given a single value (for example: number “5”, modal variable “woman”). In learning, it is not so important how student solved a specific task in a specific moment rather than how he or she is progressing through the course. So a single value for learning or action in the e-learning system may not be the best way how to present the data.

Symbolic data can be used to compress big data into analysable size or present data that is more natural in symbolic form. Symbolic data values are defined as hypercube in p-dimensional space opposite to classical data values that are defined as points in p-dimensional space [172]. Symbolic data comes in more complex forms than classical data – it does not only contain values or categories but also has internal structure and variation [49].

Good example of data that is more natural in symbolic form is weight as it is constantly changing. Therefore, instead of using single value to give persons’ weight, it would be more natural to give it as an interval. Interval valued data [49] can be derived even more precisely when probabilities are added to the intervals – that kind of symbolic data is called histogram data. For current model histogram, valued data is used.

For grouping similar students’ behaviours, a hierarchical clustering algorithm was used based on Cartesian system model [173]. The generalized Murkowski metrics was modified to work with histogram data. Cluster compactness was used as criteria in merging clusters. Cluster compactness was defined as the arithmetic means of distances between cluster average observation and every other observation used to form the cluster. Defined cluster compactness is also a metric. This kind of approach guarantees most compact clusters at the end of the clustering. The drawback is that, instead of common hierarchical clustering, this method is algorithmically more complex.

Clustering algorithm is used to find groups among previous students. Current students’ histograms are compared with those groups using similar metric as in clustering. A group with whom the current histogram has lowest dissimilarity is taken as a base for describing students’ predicted process.

6.2 Symbolic Data

The symbolic data analysis (SDA) [49, 174] is an approach to data analysis that allows describing and analysing complex data. If classical data is described by giving a single value to each variable, then the symbolic data appears in many different forms. That kind of data allows considering more complex situations and relations.

The symbolic data can be extracted from many distinct sources and in many different ways. The common feature is that bigger traditional data sets will be aggregated into more compact forms of data sets that hide entries’ specific information and give a more general and summative picture over the source data. Therefore, SDA methods enable to analyse big data sets (big data) that are too large to be analysed by usual methods. Furthermore, the fact that aggregated data hides entry specific information makes the symbolic data also suitable for fields where privacy concerns are vital.

The symbolic data has many forms – it can be described as an interval, list of modal variables or histograms, for example. There are many works concerning measuring dis/similarity between intervals or modal variables [173, 175, 176] but fewer publications concerning histogram data and how to measure dis/similarity between two histograms.

The problem of finding dis/similarity between histograms has been researched in the area of digital image processing and pattern recognition of images [177-180]. Those approaches are suitable for their domain as they consider specific characteristics of their base problem.

Kim and Billard [181] have proposed a methodology for general symbolic data, in order to find dissimilarities between histograms that have an equal number of bins with the same width. This may not be the case in real-life situation when data comes from different sources. Kim and Billard offer additional transformation to modify histograms so that their method could be used.

Ichino and Brito [182] have given a method that uses a preselected number of quantile vectors to represent histograms in order to measure their dissimilarity. Their method requires the transformation of the given symbolic data table to a larger sized classical data table.

Therefore, more general method that accepts histogram data with different number of bins and bin widths is proposed.

6.3 Representation of Objects by Histograms

Let ω_i, i = 1, 2,..., s, be the given objects, and let features F_j , j = 1, 2,..., p, describe each object.

Let D_j be the domain of feature F_j, j =1, 2,…, p. Then, the *feature space* is defined by

$$\boldsymbol{D}^{(p)} = D_1 \times D_2 \times \cdots \times D_p . \tag{6.1}$$

Since we permit the simultaneous use of various feature types, we use the notation $\boldsymbol{D}^{(p)}$ for the feature space in order to distinguish it from usual p-dimensional Euclidean space $\boldsymbol{D}^p$. Each element of $\boldsymbol{D}^{(p)}$ is represented by

$$\boldsymbol{E} = E_1 \times E_2 \times \cdots \times E_{p,} \tag{6.2}$$

where E_j,j =1, 2,… ,p, is the feature value taken by the feature F_j .

6.3.1 Histogram Feature

For each object ω_i, let each feature F_j be represented by histogram value:

$$E_{ij} = \{[a_{ijk}, b_{ijk}), p_{ijk}; k = 1,2 \dots n_{ij}\} \tag{6.3}$$

where $\sum_{k=1}^{n_{ij}} p_{ijk} = 1$, $b_{ijk} = a_{ij(k+1)}$, and n_{ij} is the number of bins that compose the histogram E_{ij}.

Therefore, the Cartesian product of p histogram values represents an object ω_i:

$$\boldsymbol{E}_i = E_{i1} \times E_{i2} \times \cdots \times E_{ip} \tag{6.4}$$

Since, interval-valued feature is special case of histogram feature with $n_{ij} = 1$ and $p_{ij1} = 1$, the representation of (9) is reduced to:

$$E_{ij} = [a_{ij}, b_{ij}]. \tag{6.5}$$

6.3.2 Histogram Representation of Other Feature Types

6.3.2.1 Categorical Multi-Valued Feature

Let F_j be a categorical multi-valued feature, and let E_{ij} be a value of F_j for an object ω_i. The value E_{ij} contains one or more categorical values taken from the domain D_j that is composed of finite possible categorical values. For example, E_{ij} = {"white", "green"}is a value taken from the domain D_j = {“white”, “red”, “blue”, “green”, “black” }. For this kind feature value, we can use again a histogram. For each value in domain D_j, we assign an interval with equal width. Then, assuming uniform probability for values in multi-valued feature, we assign probabilities to each interval associated with specific value in D_j according its presence in E_{ij}. Therefore, the feature value E_{ij} ={“white”, “green”}, for example, is now represented by the histogram E_{ij} = {[0, 1)0.5, [1, 2)0, [2, 3)0, [3, 4)0.5, [4, 5) 0}.

6.3.2.2 Modal Multi-Valued Feature

Let $D_j = \{v_{j1}, v_{j2}, \dots, v_{jn}\}$ be a finite list of possible outcomes, and be the domain of a modal multi-valued feature F_j. A feature value E_{ij} for object ω_i is a subset of D_j with nonnegative measure attached to each of the values in that subset, and the sum of those nonnegative measures is *one*:

$$E_{ij} = \left\{v_{ij1}, p_{ij1}; v_{ij2}, p_{ij2}; \dots; v_{ijn_{ij}}, p_{ijn_{ij}}\right\}, \tag{6.6}$$

where $\left\{v_{ij1}, v_{ij2}, \dots, v_{ijn_{ij}}\right\} \subset D_j$, v_{ijk} occurs with the nonnegative weight p_{ijk}, k =1, 2,…, n_{ij}, and with $\sum_{k=1}^{n_{ij}} p_{ijk} = 1$.

For example, E_{ij} = {“white”, 0.8; “green”, 0.2 } is a value of the modal multi-valued feature defined on the domain D_j = {“white”, “red”, “blue”, “green”, “black”}. By the same manner for the categorical multi-valued feature, we assign a same sized interval to each possible feature value from the domain D_j. The probabilities assigned to a specific feature value of the modal multi-valued feature are used as the bin probabilities of the corresponding histogram with the same bin width. Therefore, in the above example, we have a histogram representation: E_{ij}={[0, 1)0.8, [1, 2)0, [2, 3)0, [3, 4)0.2, [4, 5)0}.

6.4 The Cartesian System Model and a New Dissimilarity Measure

6.4.1 Definitions of the Cartesian Join and Meet Based on the Histogram Representations

Let ω_i and ω_l be the given two objects, and let E_{ij} and E_{lj} in (13) be their histogram values for the j-th feature:

$E_{ij} = \{[a_{ijk}, b_{ijk}), p_{ijk}; k = 1,2 \ldots n_{ij}\}$ and $E_{lj} = \{[a_{ljk}, b_{ljk}), p_{ljk}; k = 1,2 \ldots n_{lj}\}$
(6.7)

Then, we define the Cartesian join and meet for these two histograms as follows.

Definition 1: Let E_{ij} and E_{lj} be two histogram values with respect to the j-th feature. We represent the Cartesian join of E_{ij} and E_{lj} by $E_{ij} \boxplus E_{lj}$, and define it by the following histogram.

$$E_{ij} \boxplus E_{lj} = \{[a_{(i\boxplus l)jk}, b_{(i\boxplus l)jk}), p_{(i\boxplus l)jk}\ ; k = 1,2 \ldots n_{(i\boxplus l)j}\} \qquad (6.8)$$

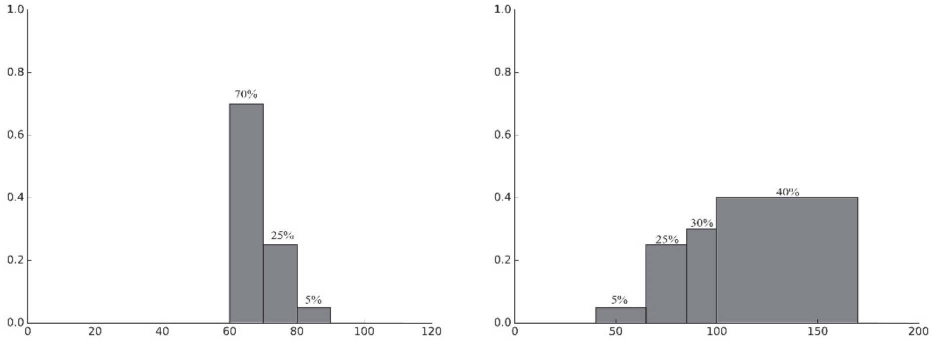
where:

1) $[a_{(i\boxplus l)jk}, b_{(i\boxplus l)jk}) \in [a_{ijm}, b_{ijm})$ or $[a_{(i\boxplus l)jk}, b_{(i\boxplus l)jk}) \in [a_{ljr}, b_{ljr})$ for every m =1, 2,..., n_{ij} and r = 1, 2,..., n_{lj} where k =1, 2,…, $n_{(i\boxplus l)j}$.
2) $a_{(i\boxplus l)jk} = a_{ijm}$ or $a_{(i\boxplus l)jk} = a_{ljr}$ and $b_{(i\boxplus l)jk} = b_{ijm}$ or $b_{(i\boxplus l)jk} = b_{ljr}$
3) $p_{(i\boxplus l)jk} = \left(\frac{p_{ijm}*(b_{(i\boxplus l)jk} - a_{(i\boxplus l)jk})}{(b_{ijm} - a_{ijm})} + \frac{p_{ljr}*(b_{(i\boxplus l)jk} - a_{(i\boxplus l)jk})}{(b_{ljr} - a_{ljr})}\right)/2$

The Cartesian join $E_{ij} \boxplus E_{lj}$ has the following properties:

1) If $a_{(i\boxplus l)jk} = a_{ijm} = a_{ljr}$ and $b_{(i\boxplus l)jk} = b_{ijm} = b_{ljr}$ for all $k = m = r$, then both histograms E_{ij} and E_{lj} have the same number of bins with the same width, and therefore histogram $E_{ij} \boxplus E_{lj}$ will also have the same number of bins ($n_{ij} = n_{lj} = n_{(i\boxplus l)j}$).
2) If histograms E_{ij} and E_{lj} had different number of bins with different width, additional bins are generated for $E_{ij} \boxplus E_{lj}$.
3) The sum of probabilities $p_{(i\boxplus l)jk}$, k = 1, 2,..., $n_{(i\boxplus l)}$ equals 1.
4) The number of bins $n_{(i\boxplus l)j}$ satisfies the inequality:

$$n_{ij}, n_{lj} \leq n_{(i\boxplus l)j} \leq n_{ij} + n_{lj}. \qquad (6.9)$$



(a) E_{ij} (b) E_{lj}

Figure 6.1: Graphical representation of histograms E_{ij} *and* E_{lj}.

Example 1

Let E_{ij} and E_{lj} be two histograms given by:

E_{ij} = {[60, 70)0.7; [70, 80)0.25; [80, 90)0.05}, and

E_{lj} = {[40, 65)0.05; [65, 85)0.25; [85, 100)0.3; [100, 170)0.4}

Figure 6.1 is graphical representation of these histograms. Then, by the Definition 1, we have the histogram for the Cartesian join of E_{ij} and E_{lj} as:

$$E_{ij} \boxplus E_{lj} = \left\{ \begin{array}{l} [40, 60)0.02; [60, 65)0.18; [65, 70)0.20625; [70, 80)0.1875; \\ [80, 85)0.04375; [85, 90)0.0625; [90, 100)0.1; [100, 170)0.2 \end{array} \right\}$$

Figure 6.2 illustrates the obtained histogram.

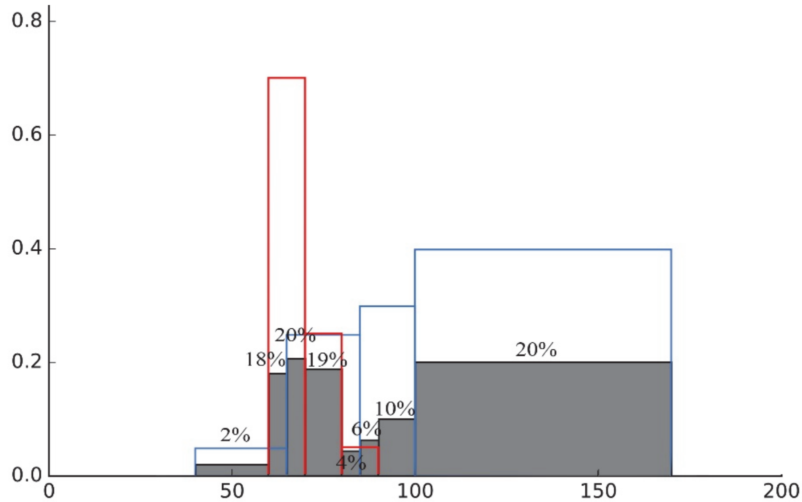


Figure 6.2: The histogram of the Cartesian join $E_{ij} \boxplus E_{lj}$.

Definition 2: Let E_{ij} and E_{lj} be two histogram values with respect to the *j*-th feature. We represent the Cartesian meet of E_{ij} and E_{lj} by $E_{ij} \boxtimes E_{lj}$, and define it by the following histogram. [173]

$$E_{ij} \boxtimes E_{lj} = \{[a_{(i\boxtimes \mathrm{l})jk}, b_{(i\boxtimes \mathrm{l})jk}), p_{(i\boxtimes \mathrm{l})jk}\ ; k = 1,2 \ldots n_{(i\boxtimes \mathrm{l})j}\} \quad (6.10)$$

where:

1) $[a_{(i\boxtimes l)jk}, b_{(i\boxtimes l)jk}) \in [a_{ijm}, b_{ijm})$ or $[a_{(i\boxtimes l)jk}, b_{(i\boxtimes l)jk}) \in [a_{ljr}, b_{ljr})$ for every m =1, 2,..., n_{ij} and r = 1, 2,..., n_{lj} where k =1,2,… $n_{(i\boxtimes l)j}$.
2) $a_{(i\boxtimes l)jk} = a_{ijm}$ or $a_{(i\boxtimes l)jk} = a_{ljr}$ and $b_{(i\boxtimes l)jk} = b_{ijm}$ or $b_{(i\boxtimes l)jk} = b_{ljr}$
3) $p_{(i\boxtimes l)jr} = min(\frac{p_{ijm}\times(b_{(i\boxplus l)jk}-a_{(i\boxplus l)jk})}{(b_{ijm}-a_{ijm})}, \frac{p_{ljr}\times(b_{(i\boxplus l)jk}-a_{(i\boxplus l)jk})}{(b_{ljr}-a_{ljr})})$

The Cartesian meet $E_{ij} \boxtimes E_{lj}$ has the following properties:

1) If $a_{(i\boxtimes l)jk} = a_{ijm} = a_{ljr}$ and $b_{(i\boxtimes l)jk} = b_{ijm} = b_{ljr}$ for all $k = m = r$, then both histograms E_{ij} and E_{lj} have the same number of bins with the same width, and therefore histogram $E_{ij} \boxtimes E_{lj}$ will also have the same number of bins ($n_{ij} = n_{lj} = n_{(i\boxplus l)j}$).
2) If histograms E_{ij} and E_{lj} have different number of bins with different widths, additional bins are generated for $E_{ij} \boxtimes E_{lj}$.
3) The sum of probabilities $p_{(i\boxtimes l)jk}$, k =1, 2,…, $n_{(i\boxtimes l)j}$, satisfies $0 \leq p_{(i\boxtimes l)jk} \leq 1$. We obtain $p_{(i\boxtimes l)jk}$=1 when histograms have the same number of bins with same width and probabilities, and $p_{(i\boxtimes l)jk}$=0 when histograms have no overlapping bins.
4) The number of bins $n_{(i\boxtimes l)j}$ satisfies again the inequality:

$$n_{ij}, n_{lj} \leq n_{(i\boxtimes l)j} \leq n_{ij} + n_{lj}.$$

Example 2

Let E_{ij} and E_{lj} be two histograms given in Example 1. Then, by the Definition 2, we have the histogram for the Cartesian meet of E_{ij} and E_{lj} as:

$E_{ij} \boxtimes E_{lj}$ =
{[60, 65)0.01; [65, 70)0.0625; [70, 80)0.125; [80, 85)0.025; [85, 90)0.025 }

Figure 6.3 illustrates the histogram of the obtained Cartesian meet of E_{ij} and E_{lj}.

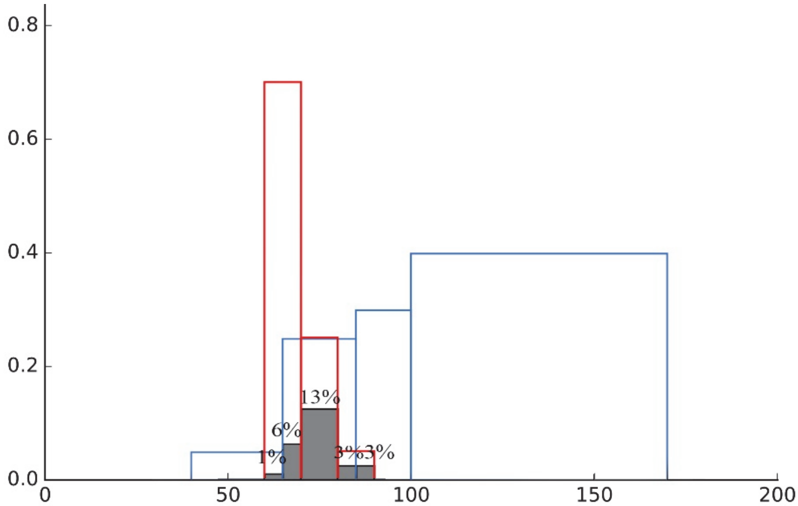


Figure 6.3: Illustration of the Cartesian meet of $E_{ij} \boxtimes E_{lj}$.

It should be noted that Definitions 1 and 2 are valid for histograms with different number of bins and different bin widths. The algorithms to obtain the Cartesian join and meet are represented in Appendix 2.

We are able to obtain the Cartesian join and meet for other feature types following the same principle as described for histograms. The triplet ($D^{(p)}$, $\boxplus$, $\boxtimes$) is initially called as the Cartesian space model [173], and is now called as the Cartesian system model (*CSM*) [49].

6.4.2 New Dissimilarity Measure

We introduce the notion of the *size* of histograms in order to define new dissimilarity measure for histogram-valued objects.

Definition 3: Let $E_{ij} = \{[a_{ijk}, b_{ijk}), p_{ijk}; k = 1,2 \ldots n_{ij}\}$ be a histogram value of object ω_i with respect to j-th feature. We define the *histogram size* of E_{ij} by

$$|E_{ij}| = \sum_{k=1}^{n_{ij}}((b_{ijk} - a_{ijk}) \times \sum_{m=1}^{k} p_{ijm}). \tag{6.11}$$

From the definition, the size of the histogram is equivalent to the sum of the area covered by the cumulative probability bins.

We should note the following properties for the histogram size.

Proposition 1

1) The histogram size $|E_{ij}|$ is reduced to a single bin width w multiplied by the sum of cumulative probabilities, if the given n_{ij} bins have the same width w.
2) The histogram size $|E_{ij}|$ is reduced to $(\sum_{k=1}^{n_{ij}}((b_{ijk} - a_{ijk}) \times k)/n_{ij}$, if all bin

probabilities are the same, i.e., $p_{ijk} = (1/n_{ij})$, $k = 1, 2, ..., n_{ij}$.

Proof: Properties 1) - 2) are clear from the definition in (6.11).

Example 4

The sizes for the histograms in Example 1 become as follows.

$|E_{ij}| = |\{[60,70)0.7; [70, 80)0.25; [80, 90)0.05\}| = (70 - 60)\times 0.7 + (80 - 70)\times 0.95 + (90 - 80)\times 1 = 26.5$

$|E_{lj}| = |\{[40, 65)0.05; [65, 85)0.25; [85, 100)0.3; [100, 170)0.4\}| = 86.25$

Definition 4: Let $E_{ij} \boxplus E_{lj} = \{[a_{(i\boxplus l)jk}, b_{(i\boxplus l)jk}), p_{(i\boxplus l)jk}\ ; k = 1,2 \dots n_{(i\boxplus l)j}\}$ be the Cartesian join of histograms E_{ij} and E_{lj}. We define the size of the Cartesian join of histograms E_{ij} and E_{lj} by

$$|E_{ij} \boxplus E_{lj}| = \sum_{k=1}^{n_{(i\boxplus l)}} (b_{(i\boxplus l)jk} - a_{(i\boxplus l)jk}) \times \sum_{m=1}^{k} p_{(i\boxplus l)jm}. \quad (6.12)$$

Definition 5: Let $E_{ij} \boxtimes E_{lj} = \{[a_{(i\boxtimes l)jk}, b_{(i\boxtimes l)jk}), p_{(i\boxtimes l)jk}\ ; k = 1,2 \dots n_{(i\boxtimes l)j}\}$ be the Cartesian meet of histograms E_{ij} and E_{lj}. We define the size of the Cartesian meet of histograms E_{ij} and E_{lj} by

$$|E_{ij} \boxtimes E_{lj}| = \sum_{k=1}^{n_{(i\boxtimes l)}} (b_{(i\boxtimes l)jk} - a_{(i\boxtimes l)jk}) \times \sum_{m=1}^{k} p_{(i\boxtimes l)jm}. \quad (6.13)$$

The sizes of the Cartesian join and meet satisfy the following properties.

Proposition 2

1) $min(|E_{ij}|, |E_{lj}|) \leq |E_{ij} \boxplus E_{lj}| \leq max(b_{ijnr_i}, b_{ljnr_l}) - min(a_{ij0}, a_{lj0})$.
2) $|E_{ij} \boxplus E_{lj}| = 0.5 \times |E_{ij}| + 0.5 \times |E_{lj}| + 0.5 \times |b_{ijnr_i} - b_{ljnr_l}|$ if E_{ij} and E_{lj} have no overlap.
3) $|E_{ij} \boxplus E_{lj}| = |E_{ij}| = |E_{lj}|$ if $E_{ij} = E_{lj}$.
4) The size of meet satisfies: $0 \leq |E_{ij} \boxtimes E_{lj}| \leq \min(|E_{ij}|, |E_{lj}|)$.
5) $|E_{ij} \boxtimes E_{lj}| = min(|E_{ij}|, |E_{lj}|)$ if $E_{ij} = E_{lj}$ and $|E_{ij} \boxtimes E_{lj}| = 0$ if E_{ij} and E_{lj} have no overlap.

Proof:

1) Size of join of histograms cannot be larger than the interval covered by the two histograms. Minimum case comes from the definition of histogram size. As histograms with different bin width's and bin numbers can be joined, it can happen that bins may be split and their probabilities uniformly divided between new bins. To due to formula (6.11), if bin is divided into smaller bins, its probabilities will get smaller and its size will also shrink as can be seen in Example 6.
2) When histograms have no overlapping area, half of the size of both histograms contribute to the size of the join due to formula (6.8). In addition to that, when

first histogram is finished, cumulative probability 0.5 is achieved. That probability 0.5 continues to the end of second histogram, its size being cumulatively added to the 0.5, reaching 1 for the end of second histogram, as can be seen from Figure 6.4.

The properties 3-5 are clear from the definitions in formulas (6.8) ~ (6.13).

Example 5

We obtain the sizes of the Cartesian join and meet in Examples 1 and 2 as:

$$|E_{ij} \boxplus E_{lj}| = \left|\begin{Bmatrix} [40,60)0.02; \ [60,65)0.18; \ [65,70)0.206; \ [70,80)0.188; \\ [80,85)0.044; \ [85,90)0.0625; \ [90,100)0.1; \ [100,170)0.2 \end{Bmatrix}\right|$$

= (60 - 40)×0.02 + (65 - 60)×0.2 +(70 - 65)×0.406 +(80 - 70)×0.594 + (85 - 80)×0.638 + (90 - 85)×0.7 + (100 - 90)×0.8 + (170 - 100)×1 = 94.15

$|E_{ij} \boxtimes E_{lj}|$ = |{[60, 65)0.01; [65, 70)0.0625; [70, 80)0.125; [80, 85)0.025; [85,90)0.025}| = (65 -60)×0.01 + (70 - 65)×0.073 + (80 - 70)×0.198 + (85 - 80)×0.223 + [(90 - 85)×0.248 = 4.738

Since $|E_{ij}|$ = 26.5 and $|E_{lj}|$ = 86.25 from Example 4, we can easily check the properties in Proposition 2.

Example 6

Assume we have two overlapping histograms E_{ij}={[10, 30]1 } and E_{lj}={[10, 20]0.5; [20, 30]0.5}. They are almost similar, except E_{lj} has been split into two bins. Their sizes are $|E_{ij}| = 20$ and $|E_{lj}| = (20-10) \times 0.5 + (30-20) \times \ (0.5+0.5) = 15$. Their join is $E_{ij} \boxplus E_{lj}$= {[10, 20]0.5; [20, 30]0.5} as E_{ij} is split into two bins using (16) and its probabilities are divided between bins assuming uniform distribution. The size of the join is: $|E_{ij} \boxplus E_{lj}|$=|{[10, 20]0.5; [20, 30]0.5}|=15 that equals the size of $|E_{lj}|$. In this case, size of the join is equal to the minimum of sizes of the original histograms.

Example 7

Assume we have two non-overlapping histograms E_{ij} = {[10, 20]1 }and E_{lj} = {[50, 60]1 } with both size 10. Their Cartesian join, as can be seen in Figure 6.4, is: $E_{ij} \boxplus E_{lj}$ ={[10, 20]0.5; [50, 60]0.5}. The size of the join is: $|E_{ij} \boxplus E_{lj}|$ = |{[10, 20]0.5; [20, 50]0; [50, 60]0.5}| = (20 - 10)×0.5 + (50 -20)×0.5 + (60 - 50)×1 = 5 + 15 + 10 = 30. It can also be seen, that when histograms have no overlap, size of join can be larger than the sum of the sizes for two histograms.

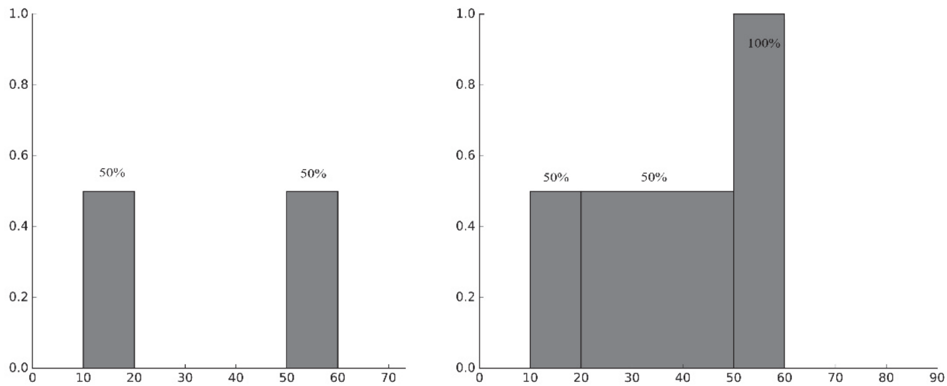


Figure 6.4: (a) Join of histograms {[10, 20] 1} and {[50, 60] 1} and (b) their cumulative probabilities

Now, we define a new dissimilarity measure using the same framework in [173] as follows.

Definition 6: Let E_{ij} and E_{lj} be the given two histograms. We define the dissimilarity between two histograms E_{ij} and E_{lj} with respect to the *j*-th feature by:

$$\Phi(E_{ij}, E_{lj}) = |E_{ij} \boxplus E_{lj}| - |E_{ij} \boxtimes E_{lj}|. \tag{6.14}$$

Definition 7: Let ω_i and ω_l be two objects presented in the *p*-dimensional feature space $D^{(p)}$. Then, we define the dissimilarity between objects ω_i and ω_l by:

$$\Phi\,(\omega_i, \omega_l) = \sum_{j=1}^{p} \Phi(E_{ij}, E_{lj}) = \sum_{j=1}^{p} (|E_{ij} \boxplus E_{lj}| - |E_{ij} \boxtimes E_{lj}|), \tag{6.15}$$

where we set the parameter γ in [173] to be *zero*.

For each feature F_j, *j*=1, 2,…, *p*, let $|F_j|$ be the length of the support interval the feature F_j. Then, we define the normalized dissimilarity measure as follows:

$$\Phi_N(\omega_i, \omega_l) = \tfrac{1}{p}\sum_{j=1}^{p} \Phi(E_{ij}, E_{lj})/|F_j| = \tfrac{1}{p}\sum_{j=1}^{p}(|E_{ij} \boxplus E_{lj}| - |E_{ij} \boxtimes E_{lj}|)\,/|F_j|. \tag{6.16}$$

From the property 5) of Proposition 2, we have the inequality:

$$0 \leq \Phi_N(\omega_i, \omega_l) \leq 1. \tag{6.17}$$

The definition (6.16) may be useful, when *p* features have different units.

We have the following proposition to assert the dissimilarity measures in (6.15) and (6.16) are metric.

Proposition 3

We have the following properties for any set of objects ω_u, ω_v, and ω_r:

1) $\Phi\,(\omega_u, \omega_v) \geq 0$

2) $\Phi\,(\omega_u, \omega_v) = 0\ iff\ \omega_u = \omega_v$
3) $\Phi\,(\omega_u, \omega_v) = \Phi\,(\omega_v, \omega_u)$
4) $\Phi\,(\omega_u, \omega_v) \leq \Phi\,(\omega_u, \omega_r) + \Phi\,(\omega_r, \omega_v)$

Proof: Properties 1), 2), and 3) are clear from Definitions 4 - 6, and Proposition 2. The proof of the triangle law is described in Appendix 3.

A special case of histogram when the histogram is composed of single bin with probability one is an interval. Then, the proposed dissimilarity reverts to Ichino-Yaguchi dissimilarity measure in which the parameter γ selected to be *zero* [173].

6.4.3 Application to Clustering

The hierarchical clustering procedure repeats the two steps: finding the most similar two clusters of objects among whole sets of objects based on a selected (dis)-similarity measure, and then combining those selected two clusters based on a linkage method [183].

6.4.3.1 Linkage Using Weighted Averages

Several linkage methods have been used for histogram-valued data. For example, Irpino and Verde [178] use a Ward criterion and inertia to combine histograms. Cha uses correlation coefficient [184]. Kim and Billard state that they use "average linkage method" [181].

We define a new method to link histograms based on the weighted average. For each cluster A, the number of objects, nr_A, in the cluster has to be known.

Definition 8: For each cluster A containing objects ω_i, i = [1, 2,..., nr_A], there exists object ω_{A_c} called descriptive object that is the average of all objects in A characterising that cluster:

$$\omega_{A_c} = E_{A1} \times E_{A2} \times \cdots \times E_{Ap}. \tag{6.18}$$

If cluster A contains only object ω_i, then the descriptive object ω_{A_c} is equivalent to the object.

When cluster A (containing nr_A elements) and B (containing nr_B elements) are combined, new descriptive object $\omega_{(A\cup B)_c}$ containing p histograms (one for each feature) has to be calculated for the new cluster $A\cup B$ (containing $nr_{AB} = nr_A + nr_B$ elements) that then characterises all elements in the new combined cluster. For each feature F_j (j =1, 2,…, p), cluster A contains histogram E_{A_cj} and cluster B contains histogram E_{B_cj}. Therefore, for the new cluster $A\cup B$ the weighted average histogram $E_{(A\cup B)_cj}$ has to be generated. It is found using the procedure described in Appendix 2. If E_{A_cj} and E_{B_cj} have different number of bins with different width additional bins are generated by the process. Uniform distributions of probabilities inside the bins are assumed. Therefore, when one histogram's bins are "cut" into two new bins, their probabilities are divided equally according to the new sizes of the bins.

New descriptive histogram for feature j will consist of bins with interval $[a_{(A\cup B)_c jk}, b_{(A\cup B)_c jk})$ that intersect with at least one cluster A's histogram $E_{A_c j}$'s bin $[a_{A_c jk}, b_{A_c jk})$ with probability $p_{A_c jk}$ or with cluster B's histogram $E_{B_c j}$'s bin $[a_{B_c jk}, b_{B_c jk})$ with probability $p_{B_c jk}$. For every bin-covering interval $[a_{(A\cup B)_c jk}, b_{(A\cup B)_c jk})$ in $E_{(A\cup B)_c j}$, probability associated with that interval is found using weighted average:

$$p_{(A\cup B)_c jk} = \left(\frac{p_{A_c jk} \times nr_A \times (b_{(A\cup B)_c jk} - a_{(A\cup B)_c jk})}{(b_{A_c jk} - a_{A_c jk})} + \frac{p_{B_c jk} \times nr_B \times (b_{(A\cup B)_c jk} - a_{(A\cup B)_c jk})}{(b_{B_c jk} - a_{B_c jk})}\right) / (nr_{a+} nr_b) \tag{6.19}$$

If histogram $E_{A_c j}$ and histogram $E_{B_c j}$ have same number of bins with same width formula 6.19 simplifies to following form:

$$p_{(A\cup B)_c jk} = \frac{p_{A_c jk} \times nr_A + p_{B_c jk} \times nr_B}{nr_A + nr_B} \tag{6.20}$$

If there is no intersecting bin for one histogram's bin, then it is assumed that other histogram's matching bin has probability 0.

Example 8

Assume we have two histograms: $E_{A_c j}$={[0, 1)0.4; [1, 3)0; [3, 4)0.6; [4, 5)0} containing four elements and $E_{B_c j}$={[0, 1)0.6; [1, 3)0.2; [3, 4)0.2; [4, 5)0} containing two elements. Both histograms have the same number of bins with the same width. Therefore, the result of linkage will also contain only those bins with the same width.

For the first bin with interval [0, 1) the probability is found: $p_{(A\cup B)_c j1} = \frac{0.4\times 4\ +\ 0.6\times 2}{4+2} = 0.47$.

For the second bin with interval [1, 3) the probability is found: $p_{(A\cup B)_c j2} = \frac{0\times 4\ +\ 0.2\times 2}{4+2} = 0.06$.

This is also done for the next two bins. Overall result of the linkage leads to the histogram $E_{(A\cup B)_c j}$ ={[0, 1)0.47; [1, 3)0.06; [3, 4)0.47; [4, 5)0} describing 6 elements.

Example 9

Assume we have two histograms: $E_{A_c j}$={[0, 2)0.2; [2, 5)0.8} containing 1 element and $E_{B_c j}$={[0, 1)0.1; [1, 3)0.2; [3, 4)0.7} containing 2 elements. They have different number of bins with different widths. Additional bins will be generated during the process.

For the first bin with interval [0, 1) the probability is found: $p_{(A\cup B)_c j1} = \frac{\left(\frac{0.2\times 1\times(1-0)}{(2-0)} + \frac{0.1\times 2\times(1-0)}{(1-0)}\right)}{1+2} = 0.1$.

For the second bin with interval [1, 2) the probability is found: $p_{(A\cup B)_c j2} = \frac{\left(\frac{0.2\times1\times(2-1)}{(2-0)}+\frac{0.2\times2\times(2-1)}{(3-1)}\right)}{1+2} = 0.1$

Overall result of the linkage leads to the histogram $E_{(A\cup B)_c j} = \{[0,1)0.1; [1,2)0.1; [2,3)0.156; [3,4)0.556; [4,5)0.089\}$ with 5 bins describing 3 elements.

The following is the proposed algorithm for hierarchical clustering for objects containing histogram-valued descriptions.

Algorithm

1. Add all objects ω_i (i =1, 2,…, s) as the initial clusters containing single observation into the set U.
2. For each pair of clusters ω_i and ω_j in U, calculate distance $\Phi_N(\omega_{i_C}, \omega_{j_C})$ between descriptive objects by using formula (6.19) and the procedure in appendix 1. Then, find the pair ω_p and ω_q that has the smallest distance $\Phi_N(\omega_{p_C}, \omega_{q_C})$.
3. Merge the pair ω_p and ω_q by using the procedure in Appendix 1 for finding bins and formula (6.19) for finding probabilities. New cluster ω_{pq} is generated as a result. It is added to U, then ω_p and ω_q are removed from U.
4. Repeat Steps 2 and 3 until all objects have been merged (i.e., the size of U is 1).
5. As the result, we obtain the dendrogram that describes the proposed hierarchical clustering.

6.4.3.2 Cluster Compactness

To evaluate the effectiveness of clustering, a measure for cluster compactness or "goodness" should also be derived.

Definition 9: For a cluster A with descriptive observation ω_{A_c}, the arithmetic mean of distances between that descriptive observation and every other observation $\omega_i \in A$ in the cluster is called the *cluster compactness*:

$$C(A) = \frac{\sum_{\omega_i \in A} \Phi_N(\omega_{A_c}, \omega_i)}{nr_A}. \qquad (6.21)$$

Since all the normalized distances take values between 0 and 1, and since the sum of the normalized distances are divided by the number of observations in A (the count of sums added together), the cluster compactness $C(A)$ also satisfies the inequality:

$$0 \leq C(A) \leq 1. \qquad (6.22)$$

We should note that the smaller the cluster compactness, more common the elements are inside the cluster. It should also be noted that when all the objects are linked together, neither $\Phi_N(\omega_i, \omega_l)$ nor $C(A)$ may equal to one. In dissimilarity case, it shows how much of the histograms match. In compactness case the value shows, on average, how close by are objects inside the cluster to the centre of the cluster.

The cluster compactness is also applicable to the clustering as the measure of similarity. In step 2 of the algorithm, two clusters are combined and new cluster's compactness is found using formula (6.21). In step 3, the pair with the smallest compactness is merged.

This kind of approach guarantees most compact clusters at the end of the clustering. The drawback is that instead of finding just one operation distance between descriptive objects, now the merging is done first and then distance between all elements and new descriptive object is calculated resulting with algorithmically more expensive procedure.

It should be noted that the proposed compactness (6.21) becomes also a metric satisfying Proposition 3 due to its definition. The only special circumstance is with condition 2. The compactness of two clusters is zero only when those two clusters are the same and they contain only one element or the two clusters are the same and they contain multiple elements that are all equal to its descriptive object.

To verify the correctness of the proposed measure it was validated in two commonly used datasets – Hardwood [185] and US State Temperatures [186] data. The verification can be seen in Appendix D.

6.5 Implementation of Prediction Model

Proposed new method for finding dissimilarity between histogram valued symbolic data allows us to cluster and compare data in form that is found to be most suitable for learning data. Therefore the prediction model can be built. The first step is to determine the set of variables that are principle components. To transform classical data from database to symbolic form, a week was used as a period for one symbolic object.

The e-learning system contains a lot of different data about students. At first, results for every competence during a time period of a week were considered. The presumption was that competences are the most important part of the course and they should reflect both student process and the course they are studying. PCA was applied to competences using R software [187]. The regression model was developed based on those principle competences but it became evident that using competences as variables did not lead to a good model. Competences only reflected one side of student activity in the system – solving tasks.

Therefore, a wider range of information was considered and different kind of variables were deducted from the data. In the end, 35 variables were considered. After PCA, 9 most important indicators were found.

Those were:

- number of tasks done;
- number of active days;
- average result of submissions;
- average difficulty level of submissions;
- number of times help material was accessed;
- time of the day of submissions;
- current progress on the course;
- number of times of HomeLabKit were borrowed;
- number of times of class test were done [162].

The results of PCA are plausible as they cover all important factors of the system, not only the task submissions.

After principle components were found, all previous study records were analysed and histograms for all the students calculated. Every semester consists of 16 weeks of teaching and additional 3 weeks for exam session. Then, another week serves as an extra time for those who did not manage to finish on time. Therefore, for every student, 20 weeks were analysed. Christmas holiday weeks during fall semester were ignored so that the results would be comparable despite courses being taught in different semesters (and some courses switching semesters during the years) [162].

The e-learning system contains information from more than 15 years. Students' study behaviours and course context have changed a lot during those years. Therefore, at first implementation, only records since competence based approach was introduced in 2010 fall semester, were considered. The analysis, of those 10 semesters considered, showed that throughout the years learning patterns have changed [162].

Despite one course, "Circuits, Systems, Signals", has been taught during those 5 years in similar manner, students' behaviour during the course has changed a lot. For example, at 2010 they worked more and frequently. Getting a good grade was the main goal. Lab experiments were done in regular intervals and learners borrowed HomeLabKits from the start of the semester once in two weeks. In 2014 fall semester, at the same course, students waited around half the semester before majority of them started to show any progress in the course. Students accept low grades as soon as they become available. Lab experiments were avoided as often as possible, being the last things done in the course. Borrowing kits has also become very irregular and less frequent [162].

Similar changes were found in other courses. Therefore, it was concluded that not as much as a course that defines the difference in behaviour, it is the year that affects patterns the most. Therefore, for building the analysis, only last 2 years (4 semesters) were considered. It would make sure that significant pattern changes in students' attitudes do not affect the model too much. It would also help overcome the problem of courses changing during the years as it could be expected that at last two years had similar or almost similar content. The downside of that constraint is that the models have to be re-evaluated every year, meaning extra work. On the other hand, it would also keep the model up-to-date and allows it to adapt to changes in students general attitudes towards studying [162].

Therefore, for implementing the model, 5 courses with largest number of participants in the last 2 years containing 1100 students in total were considered. Those courses were Circuit Theory, Basic Measurements, Microprocessors and two courses on Operating Systems. For all the students in those courses, histograms over the 20 week periods of their studies were found for all 9 most important variables. Then, for all the students, clusters were found using hierarchical clustering algorithm described in 6.4.3. Students were clustered based on the course and all together [162].

Clusters were then analysed. It was done using dendrograms. Matrix containing distances between objects was acquired as a result of clustering process using R [162].

To extract meaningful groups or clusters from dendrogram, it had to be cut. Conditions for cutting were the shape of the dendrogram and the properties of the would-be-clusters. If the properties were too specific, merging it with another cluster was considered. If the properties were too wide, cluster was cut from the furthest connection point and both of the cut clusters were considered further. This continued until groups with well-defined properties had been achieved and cutting any further would just fragmentize the result [162].

Clustering students based on the course worked very well – clear clusters with different characteristics could be detected. Those clusters also tended to have specific features for finishing time and final mark – the two parameters that were wanted to predict using the model. Clustering all the students together did not yield satisfactory results despite high hopes that there could be a possibility of developing a general model for all students independent of the courses. Clustering showed that it is not possible. Apparently courses have distinct characteristics that affect the patterns. For example, Operating System courses do not use HomeLabKit. Therefore, a pattern for borrowing kits in that course is constantly zero. On the other hand, throughout the years, as students consider Operating Systems to be an "easy" course, there are more early finishers than in any other course. The courses have also different grade units that in turn mean different learning volumes [162].

Currently, models were developed for 5 largest courses in the system. Courses had 7 to 14 clusters. As an overall rule, the clusters contained at least 20 students. There were a few exceptions - clusters with clear (very different) characteristics had less members. Despite courses having different learning volumes and credit points, there were groups with specific characteristics which were common for all courses [162].

Those common clusters were:

- “Early rushers” – students who started working at the beginning of the semester and finished within the first month. They got very good marks (4-5, in scale 0 to 5 where 0 is fail and 5 is the best mark).
- “Late rushers” – do not work at all during the semester. Start studying during the 16th week or during the exam session. They do a lot of submissions in a small timeframe. Their submission times are well below the average (they hardly have time to read the task). Some of them manage to finish with lowest grade, some of them not.
- “Constant workers” – they work regularly with low number of submissions during the whole semester
- “Drop outs” – do not work at all or very little. Do not pass.
- “Mid-semester sleepers” – Work little at the start of the semester. Then activity slows down and they start working again on last few weeks of the semester and during the exam session. They pass the course with varying grades.
- “Second half-ers” – They start working after the mid-terms. They do not rush and have average number of submissions. Usually finish on time and with grade 3 [162].

All courses also had a common characteristic that in last two years, around 20-30% of students did not finish the course with a positive grade [162].

For every cluster in every course, a descriptive observation was deducted from the clustering step, meaning that every cluster had “average” or “common” histogram for all 9 variables for 20 weeks. Therefore, for every course a model was achieved containing descriptive observations about each cluster in the course [162].

6.6 Prediction

To verify the correctness of the achieved model, during a fall semester of 2015, the course with 110 officially declared students studying Basic Measurements was used as a test group. Basic Measurements group was the biggest one, containing 14 clusters, as can be seen from Figure 6.5, and had around 450 students who had studied the course in previous two years [162].

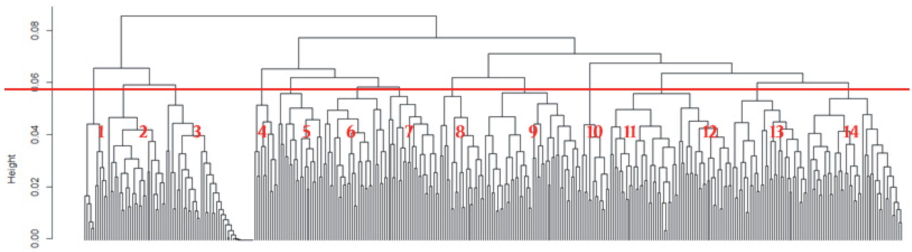


Figure 6.5: Dendrogram for students of previous two years with 14 clusters in course Basic Measurements. Red line shows cutting point of clusters [162]

During previous two years, some organizational changes have taken place. For example, the course was taught mostly for second and third year bachelor students in Informatics department (containing computer science, electronics and communication curriculums). From fall 2014, computer science students are not required to take this course. Also, from that time onwards, the course was moved to first semester [162].

Cluster 1 contained students who had completed most of the work in previous semester, before officially declaring the course, or those students who do not work at all. This group is characterized by almost non-existing activity and very bad results for submissions. Students in this group usually fail or, if they have done work in previous semester, pass with grade 3 [162].

Cluster 2 contains "late-rushers" who usually start working at the end of the semester, from week 14 onwards. They have very fast submission times and do a lot of submissions in a short time period. They tend to have bad results and in the end, during the exam session, they start using help materials as they realize they need some kind of theory. Majority of them pass with lowest grade [162].

Cluster 3, so called "drop outs", contains students who work very little, mostly on week 14 and 15. They try the course, get bad results for their submission and they give up. Students in this group do not pass the course [162].

Cluster 4, has students who try the course at the beginning of the semester, do not work at all during the semester and then try to complete it during the exam session. Around half of them manage to finish with low grades, another half fail. They stand out by having good results for submissions – so they know what they are doing but they just leave everything to the last minute. They are also called "lazy but smart" [162].

Cluster 5, "late-wakers" tend to work from week 14 onwards and usually continue up to the additional extra exam week. They have fast submission times. It means that they are not spending enough time with tasks. Therefore, they also have bad results. Students in this group usually do not pass on the right semester as they just run out of time [162].

Cluster 6, is quite similar to cluster 5. Those students also start working from week 13-14 and continue to exam session. They differ from previous cluster in that they are able to finish within three weeks during the exams. They also have a lot of submissions and mostly bad results but they have more reasonable submission times, meaning that they actually have time to read and think about what they have been asked. Surprisingly, they mostly finish with grade 3 or even with 4 [162].

Students in cluster 7 are so called “early-rushers”. They start working at the beginning of the course and usually finish by week 10. They have a high number of submissions and they get very good results. They tend to use lot of help materials, participate in class tests and borrow home kits. Despite having large number of submissions, their solving times are quite slow meaning that they spend a lot of time thinking and calculating while solving the exercises which probably contributes to their high results. Grades in this group are varying from 3 up to 5 [162].

Cluster 8 works mostly from 12 weeks forward. They have good results but their solving times are from average to slow. They do average number of submissions and usually finish with grade 3. They also have quite a high help material usage and class tests due to their working pattern – work is only done at the second half of the semester and during the exam session [162].

Cluster 9, “constant workers”, start working from the start of the course and they usually finish from week 12 onwards. Students in this group use help materials more than other students. They have average to high submission numbers, average solving times and average results. They tend to start doing class tests at the second half of the semester that does not allow them to finish earlier than mid-semester. They finish with 3s [162].

Cluster 10 is similar to cluster 7. They also have very good results and they know what they are doing. They manage to complete the course with lowest number of submissions that are submitted very fast. They are so called “geniuses”. They almost never use help materials. Despite having high knowledge, they do not care to work for highest grades. Usually they finish with 4 [162].

Cluster 11 contains “second half-ers”. They have almost no activity at the first 8-9 weeks of the semester. Then, when the mid-terms are over, they tend to remember they have registered to the course and start working. They have high number of submissions with very good results. Their results for laboratory experiments are much worse than theoretical tasks and they do not like using HomeLabKits. So they avoid lab tasks and finish with 3 or 4 [162].

Cluster 12 is so called “mid-semester sleepers”. They start working at the first half of the course, then they stop and start again from week 14 up to the end of the exam session. They do a lot of submissions on those peak working times but they tend to have bad results. They are especially slow when solving laboratory experiments. Usually they pass with grade 3. Few in this group finish with grade 2 [162].

Cluster 13 is similar to cluster 11. They also start working in the middle of the semester, usually earlier than students in cluster 11. They have bad results and they work very slowly with laboratory experiments. They are uneven learners with some weeks having more submissions and other weeks having significantly less submissions. The unevenness is not week-based but active weeks differ from student to student [162].

Students in cluster 14 are similar to students in cluster 11 but they work more slowly during the whole course. They tend to have average to low submission times. They do not rush and have almost constant number of tasks per week during most of the semester. It is surprising that they usually get good results for theoretical tasks but lab tasks are hard for them and they get bad results for those. Usually pass with 3 or 4 [162].

Table 6.1: Number of students who finished on specific week. The course is 16 weeks long and students should finish by the end of the teaching period. They do have an extra 3 weeks to get a grade during the exam session. Additionally, there is an extra week between semesters where grade acquisition is also possible (so called extra exam week).

Week	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Nr of Students	1	0	1	0	2	7	2	0	9
Week	13	14	15	16	17	18	19	20	
Nr of Students	2	8	4	4	8	12	12	0	

Using descriptive histograms from those 14 clusters, students' progress during the course was predicted. The first prediction based on the model was done after week 4. It was assumed that before week 4, there is very little activity and very few histograms can be generated for the students. For predicting student progress, at first 9 variables for four weeks were calculated. Then, using the same kind of dissimilarity measure as in hierarchical clustering, a cluster, from those 14 described, with smallest dissimilarity was found. That cluster was considered to be student's predicted pattern and student's finishing time and grade was predicted based on that [162].

Afterwards, this process was then repeated at the end of every week to keep the prediction up to date. The prediction is done up to the point when a student accepts the grade, which is varying [162].

The week when students finished, can be seen in Table 6.1. The students who get the grade can be divided into two groups – first those who try to complete the course as fast as possible and those who leave it to the end of the semester. On the semester analysed, no student managed to finish the course on extra exam week which is usually popular with "late-wakers" [162].

6.7 Analysis

By the end of the semester, 72 students (65.45%) had passed the course. This was slightly less than has been an average for the last two years. It could be explained that during this year, no student managed to complete the course in week 20 that has always been quite popular among "late-rushers". The other possible explanation could be that the course is taught on the first semester to freshmen who historically have high number of drop-outs [162].

As a result, in 89 cases (80.91%) the final grade was predicted correctly using developed prediction model as can be seen in the confusion matrix in Table 6.2. In 17 cases (15.45%), the actual final grade was higher than the predicted one. In all of the cases, it meant that student passed the course as was predicted but instead of predicted grade, the actually got a one step higher grade. This occurrence was further analysed and it was concluded that during the semester in question, it was the change in students' study behaviour compared to previous students; they continued to work further to achieve better grades. In some of those cases the grade prediction was spot on until week 17. All those students got a grade during the exam session [162].

In 4 cases (3.63%), a student was predicted to pass with 3 but in reality, the student failed. All those 4 students belonged to cluster 11. This case was also further analysed and it looks like again the prediction goes off at week 17 (start of the exam session). Before that (week 16 and before), students were clustered into cluster 3 that has characteristics of having almost no activity at all. Cluster 11, on the other hand, has the characteristics of having very high results and they avoid lab experiments. Both of those characteristics are common for those 4 students. On the other hand, those 4 students had almost no activity before week 19 or 20 and are very similar to the students in cluster 3. To be precise, with students who have zero activity during the semester. The overall descriptive histogram of cluster 3 is a combination of all those students who did very little during the semester and therefore the resulting histogram has little activity in common for almost all the weeks. Therefore, it was tried if adding another cluster with empty histograms would yield better clustering for those 4 students. As a result, for majority of the semester, those students got clustered into the empty 15^{th} cluster but, at the end of the semester, they get clustered in group 11 again [162].

Table 6.2: Confusion matrix for final grade prediction

n=110	Predicted: 0	Predicted: 1	Predicted: 2	Predicted: 3	Predicted: 4	Predicted: 5
Acutal:0	34	0	0	4	0	0
Actual:1	0	0	0	0	0	0
Actual:2	0	0	0	0	0	0
Acutal:3	0	0	0	55	0	0
Actual:4	0	0	0	17	0	0
Actual:5	0	0	0	0	0	0

The changes in clusters during the course are shown in Figure 6.6 using parallel coordinates [188]. The changes in groups are quite frequent at the beginning of the prediction process. Further into the semester, especially at the end, the changes become less frequent, almost stopping. It was expected that the changes stop earlier during the semester. On the other hand, most of the changes occurring are with adjacent clusters and during majority of the time, the final grade prediction remains the same despite cluster change [162].

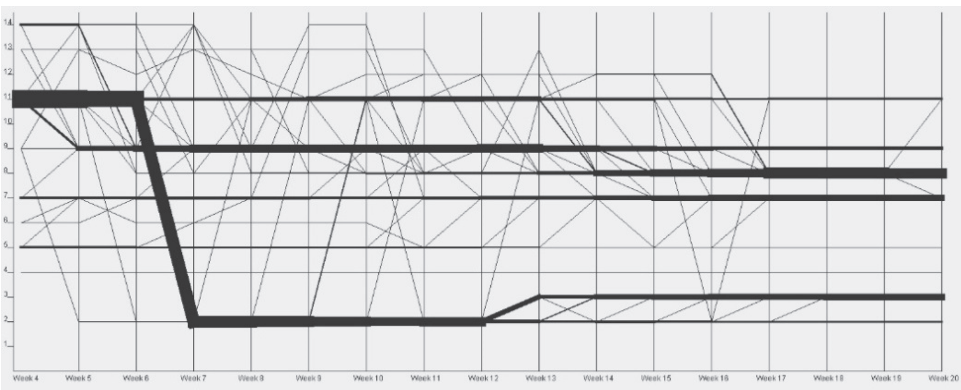


Figure 6.6: Cluster prediction and its changes (shown using parallel coordinates) for students in Basic Measurement course from week 4 till the extra exam week. The bolder the line, the more students are in the cluster that week [162]

Major changes are happening after week 7. Students who were predicted to be “mid-semester sleepers”, those who try little-bit at the first half of the semester and then start again after midterms, are expected to have shown some kind of activity but they remain inactive and are therefore clustered to cluster 2 – “late rushers". Their final grade prediction actually remains the same – what changes is the pattern that is expected of them [162].

From week 7 onwards, it can be followed, that as students start to work, they are moved away from cluster 2, mainly to cluster 11. This trend actually continued to the exam session [162].

Another major change happens after week 12 when "late-rushers" who do a lot of submissions in a short time are expected to start working. Those who remain inactive are moved to cluster 3, "drop-outs". It is problematic that "drop-outs" cannot be predicted earlier during the semester but that is due to the characteristics of students' study behaviour. Some "late-wakers" manage to finish as they work a lot in the end. Others do not work enough. Therefore the difference between those two groups become visible only at the end of the semester making it impossible to distinguish them before [162].

Changes between week 16 and week 17 draw attention. Despite changes between clusters slowing down before week 16 and grade prediction remaining almost constant, at the end of week 17, there are lot of changes between clusters that are also affecting the final grade. Week 17 is also the point where for 4 cases the prediction becomes "incorrect" – prediction for students who have been evaluated to fail are updated to cluster with pass. This gap between the end of studies and the start of exam session requires further analysis [162].

It was also observed that at the start, the predictions are usually more positive (when considering the final grade) than the predictions at the end of the semester [162].

Furthermore, for the semester at hand, the percentage of students in clusters is very different from other semesters from last two years. Clusters 3, 7 and 8 are much more present in current semester than in any of the last four semesters during the last two years. On the other hand, clusters 6, 13 and 14 are not present in current semester but they made 14%, 9% and 12% of all predictions in previous two years. There are also other differences between predictions in current analysed semester and between semesters used for developing the model but those are less significant [162].

When analysing clusters that were more likely predicted during the current semester, then those clusters are the most common "fail" cluster with students with little activity and clusters that actually have good results. As said before, during the analysed semester, students tended to work more to get better grades than during the last two years. Cluster 7, "early-rushers", is actually the only cluster where the highest grade prediction is possible. Therefore it is expected that more students got clustered into group 7 during this semester. Cluster 8 also has students with good results. Furthermore, as the course allows students to complete all the work before the end of the semester, it could explain why more students are being clustered into group 7. That option is becoming more and more popular among some students who wish to finish the course early and concentrate on other courses, which tend to leave all the deadlines to the end, later during the semester [162].

Popularity of fail cluster, cluster 3, can be explained by changes in curriculum. The course is now taught for the freshmen on the first semester and they have a high percentage of drop-outs from university during the first year, especially during the first semester [162].

When looking further into the clusters that were more common in model than during the actual test semester, changes are harder to explain. Cluster 14 contains constant workers during the whole semester who are doing well in theoretical tasks but not in lab experiments. Another cluster, 9, that also has constant workers, was also less frequent during fall 2015. Therefore, it could be argued that students are becoming less inclined to work throughout the semester. If they are good at planning, they try to complete the course quickly and move on to other courses. If they are lazier, they do not start working before second half of the course. That can be verified by other statistics from the course. During the years, number of students attending laboratory sessions or class tests at the first half of the semester, has shrank a lot. Also, during 2014-2015, the numbers of attendance (that is voluntarily – student can choose if she or he wants to do labs in university or borrow the kit and number of class tests done depends on personal study pattern) has also went down for second half of the semester with huge volumes of students registering on-site activities during exam session [162].

Cluster 6, students who work at the second half of the semester, who do a lot of submissions and usually get bad results but manage to finish with grades 3 or 4, has always been a little “unusual”. Cluster 13 contains students with very uneven study patterns. Seems that none of those extreme cases was present with students of 2015 fall semester [162].

When looking at the finishing time prediction, the results are less favourable as can be seen from Table 6.3. Only in 64 cases (58.18%) the actual finishing is within three weeks of predicted finishing week. Three weeks were chosen for prediction quality measure due to the fact that for some groups the finishing time is predicted to be “within exam session” and exam session is 3 weeks long. There were some extreme cases when student actually passed on week 9 but the group he/she was clustered into group expected the student to finish within exam session. The four cases that were problematic for the final grade prediction were also among “extremely-off” cases [162].

Table 6.3: Confusion matrix for predicting finishing time. Horizontal axis has predicted finishing time. Vertical axis has actual finishing time

n=110	w10	w11	w12	w13	w14	w15	w16	Exam session	Extra	Later
w4										1
w5										
w6										1
w7										
w8	2									
w9	4		1						2	1
w10	1						1			
w11										
w12	2		3				1			3
w13			1							1
w14	5						1			2
w15	1		1				1			1
w16	2						1			1
Exam session	3		3				1		5	20
Extra week										
Later	2		4				3		1	27

6.8 Discussion

Build model seems to be working satisfactorily for final grade prediction, with more than 80% students' being predicted with the right grade. Also, in majority of other cases the fact that their final grade was better than predicted can be seen as a good result because trying harder was a characteristic for the semester considered. As mentioned previously, the model should be re-evaluated regularly so that the newest students' trend would be taken into account. Therefore, before next semester, the model should be re-calculated with those students now added to the pool of previous students and "oldest" records removed. It will be interesting to follow how the clusters, their descriptions and predicted final grades/finishing times are changing from year to year when model is going to be re-evaluated [162].

The changes in clusters are more rapid than expected – the changes continued throughout the teaching period, slowing down by the end of it. Even during the exam session, there were slight changes. It is also interesting to follow that some of the changes have a pattern that continues throughout the weeks. For example, from week 12 onwards many students from group 2 are moved to group 3 when they are still inactive. Also a change from cluster 5 to 7 or from 9 to 8 happens often during the second half of the semester. It would be preferred if the changes in clusters could be lowered down even further by fine-tuning the model. Maybe some of the features considered before PCA should be reintroduced into clustering if they have the desired effect on the cluster prediction [162].

Using the model for predicting the students' finishing time does not yield such good results. The finishing time, which can be anything from week 1 to week 20, was averaged over the cluster with extreme cases ignored. For example, if 90% of students in the cluster finished during the exams but one student managed to finish in week 4, then the student with week 4 was not considered for averaged finishing time. Also, as there are more possible values than for final grade, it could be expected that averaged result will have some variety. The fact that there were quite extreme cases shows that there is room for improvement for the model in final grade prediction. Maybe a separate model that only concentrates on finishing time should be developed. The other option could be that when a cluster is found, student's histogram is compared with all the previous students within the cluster and that student's finishing time could be used as a prediction value [162].

The first prediction was tested only with the largest course and students weren't shown the prediction – it was due to the fact that we had no guarantee how well the model works. Now, when the models effectiveness has been proven, it could be also shown to students. The key concerns are how students react to the prediction and how can we measure if changes in students' behaviour are due to the characteristics of that semester or because of seeing the prediction model [162].

Showing predictions to students will be incorporated into the main page when students have logged into the system so that they do not have to specifically look for that information. That was decided so because all previous tools that were developed required an extra click in the system and statistics showed that students were not eager to do it. Incorporating prediction on the main page does not allow us to measure how often a student looks for the information. This problem will be solved by using "read more" link next to the prediction summary. Thus, a student sees a small prediction (grade and estimated time of finishing the course) but to read the full description of the cluster and recommendations in order what to do to get a better result, student has to click, which can be logged [162].

Another unknown factor is how students are going to react to the predictions. In psychological point of view, it has been advised that phrasing of the descriptions for the clusters and recommendations should be very carefully considered as opposite effect to the goal could be achieved – students get scared off as they are predicted to fail. On the other hand, predicting that student is going to fail usually happens at the latter half of the semester when no activity has been shown. Therefore, the prediction's description for students' in danger of later being clustered into the "drop-out" cluster, could remind them to start working earlier instead of stating that they are probably going to fail [162].

Many of the courses taught on the system have enough students for reasonable model. There are also some courses that may have only 10 or less students per semester. For those kinds of courses, developing a prediction model based on current methodology is not possible. Therefore, probably, the prediction model is going to be offered only for the largest courses in the system with many students and adequate volumes of data [162].

6.9 Conclusion

In this chapter the process of predicting students' behaviour during the course was introduced. For prediction, general dissimilarity method was developed for histogram valued symbolic data with different bins and bin widths. Compactness based on that definition was derived for better clustering results.

5 models for largest courses in e-learning system were developed for prediction student final grade and finishing time. PCA was used to find most important features. Using largest course in the e-learning system, students' behaviour was predicted during 2015 fall semester to verify the correctness of the developed model.

The main conclusions of this chapter are:

1. To lower the number of students failing the course due to non-linear workload students can be provided with prediction of their assumed behaviour based on the prediction model built on previous students' behaviour.
2. Predicting students final grade using prediction model built with histogram valued symbolic data and nine features worked well. Predicting their finishing time during the course did not yield good results.
3. It is possible to define general dissimilarity measure for histogram valued data with different bin numbers and bin widths. It is also possible to define compactness based on that general definition that yields better clustering results.
4. Prediction model can only be developed for larger courses with more than 100 students on average per semester.
5. Developing common model for all courses did not yield good clustering results as students have different behaviours depending of the course. Courses may also have specific characteristics that make the general model unfeasible.

6. Students' study behaviour is in a constant change. Suitable timeframe for building prediction model based on students' behaviour is 2 years where courses have similar content and students' study behaviour is also similar.
7. Using only competences as features for building the prediction model does not yield good and compact clusters. As learning depends on many aspects, not only competences studied, other, wider features yield better results.
8. Despite courses having different students' behaviours and characteristics there are some clusters that are common for all courses meaning that despite variation in courses, some behaviour patterns are constant.

7 CONCLUSIONS

In this chapter the overview of defining CBL framework for e-learning environment and tackling the problem of high drop-outs using visualization tools and students' study behaviour model is represented. Also the contributions of this thesis are summarised.

7.1 Overview

The e-learning environment in Department of Computer Control in Tallinn University of Technology that used classical topic-based learning was transformed into CBL. The development of e-learning environment continued to improve the CBL, help students achieve firmer competences and tackle the problem of high drop-outs.

Finally, the following conclusions can be formulated:

1. CBL is more beneficial for students in achieving high level concrete competences.
2. Usage of fine-detailed and granular grading with 128-level scale for each atomic competence allows to form a clear and detailed picture of students' current state. Definition of competences in CBL in atomic level is crucial for valid and profitable assessment.
3. Tasks have to be small for CBL to facilitate repetition. It also requires having an adequate number of different tasks to avoid reappearance of same exercises. The bare minimum of competence coverage in 128-level scale that has been divided to 8 regions is 40 tasks (5 per region).
4. Using automatic answer evaluation with algorithms mimicking student's answering process, enables to lower the impact of repeated mistakes, generate adequate feedback according to specific mistakes made, grade only competences actually used to solve the exercise and combined with pre-submission check of common typos, enables an adequate evaluation of submission. Automatic algorithm synthesis does not yield good response times and therefore is unsuitable for proposed framework.
5. CBL with self-regulated learning suffers from unpredictable workload. The amount of work needed to be done by students differs a lot and the reason why many students fail the course is because at the end of semester they run out of time. Therefore, there is a need for a method which predicts the amount of work still required to do.
6. Visual process following tools do not lower the students fail rates. Students prefer clear statements, not data to draw conclusions by themselves. Furthermore, they do not want to make an extra action to access the information.
7. Predicting students final grade using prediction model built with histogram valued symbolic data and nine features worked well. Predicting their finishing time during the course did not yield good results.

8. It is possible to define general dissimilarity measure for histogram valued data with different bin numbers and bin widths. It is also possible to define compactness based on that general definition that yields better clustering results.
9. Prediction model can only be developed for larger courses with more than 100 students on average per semester. Model has to be developed for every course separately as students have different behaviours depending of the course. Courses may also have specific characteristics that make the general model unfeasible. Furthermore, students' study behaviour is in a constant change. Suitable timeframe for building prediction model based on students' behaviour is 2 years where courses have similar content and students' study behaviour is also similar.
10. Using only competences as features for building the prediction model does not yield good and compact clusters. As learning depends on many aspects, not only competences studied, other, wider features yield better results.
11. Despite courses having different students' behaviours and characteristics there are some clusters that are common for all courses meaning that despite variation in courses, some behaviour patterns are constant.

7.2 Contributions

The contributions of this thesis can be summarised as following:

1. Framework for CBL in e-learning with personalization, memory model and granular grading was formed.
2. Steps for converting e-learning environment from classical topic-based learning to CBL was formed. Operating System course was converted from topic-based learning to CBL.
3. Methodology for developing automatic answer evaluation that mimics students answering process to lower the weight of repeated mistakes and provide detailed, adequate feedback was formed.
4. Usage of competences, students' study behaviour and other aspects of CBL were analysed in detail from e-learning system logs from 2008 spring to 2016 spring to verify the usefulness of proposed methodology.
5. Three different visualization tools were developed to tackle the problem of high drop-outs and fails. The usage of tools was analysed by collecting feedback from the learners and using data logged in the system.
6. General dissimilarity method was developed for histogram valued symbolic data with different bins and bin widths. Compactness based on that definition was derived for better clustering results. Both definitions were verified on two commonly used datasets.
7. 5 models for largest courses in e-learning system were developed for student final grade and finishing time prediction. PCA was used to find most important features. Model for all courses was also developed but it did not yield good clustering results.
8. Using largest course in the e-learning system, students' behaviour was predicted during 2015 fall semester to verify the correctness of the developed model.

REFERENCES

[1] B. Everitt, "Dictionary of statistics," ed: Cambridge University Press, Cambridge, 1998.

[2] R. S. Nickerson and P. P. Zodhiates, *Technology in education: Looking toward 2020*: Routledge, 2013.

[3] K. McKnight, K. O'Malley, R. Ruzic, M. K. Horsley, J. J. Franey, and K. Bassett, "Teaching in a digital age: how educators use technology to improve student learning," *Journal of Research on Technology in Education,* pp. 1-18, 2016.

[4] M. Warschauer and D. Healey, "Computers and language learning: An overview," *Language teaching,* vol. 31, pp. 57-71, 1998.

[5] R. E. Mayer, "Learning with technology," *The nature of learning: Using research to inspire practice,* pp. 179-198, 2010.

[6] M. D. Bush and J. D. Mott, "The transformation of learning with technology," *Educational Technology,* vol. 49, pp. 3-20, 2009.

[7] E. A. Ashburn and R. E. Floden, *Meaningful learning using technology: What educators need to know and do*: Teachers College Press, 2006.

[8] C. Buabeng-Andoh, "Factors influencing teachers' adoption and integration of information and communication technology into teaching: A review of the literature," *International Journal of Education and Development using Information and Communication Technology,* vol. 8, p. 136, 2012.

[9] J. Tondeur, J. van Braak, G. Sang, J. Voogt, P. Fisser, and A. Ottenbreit-Leftwich, "Preparing pre-service teachers to integrate technology in education: A synthesis of qualitative evidence," *Computers & Education,* vol. 59, pp. 134-144, 2012.

[10] R.-J. Chen, "Investigating models for preservice teachers' use of technology to support student-centered learning," *Computers & Education,* vol. 55, pp. 32-42, 2010.

[11] T. Georgiev, E. Georgieva, and A. Smrikarov, "M-learning-a New Stage of E-Learning," in *International Conference on Computer Systems and Technologies-CompSysTech*, 2004, pp. 1-4.

[12] A. Roehl, S. L. Reddy, and G. J. Shannon, "The flipped classroom: An opportunity to engage millennial students through active learning," *Journal of Family and Consumer Sciences,* vol. 105, p. 44, 2013.

[13] O. U. Press. (2016, 11.07). *Distance learning*. Available: http://www.oxforddictionaries.com/definition/english/distance-learning

[14] L. Sherry, "Issues in distance learning," *International journal of educational telecommunications,* vol. 1, pp. 337-365, 1996.

[15] D. Zhang and J. F. Nunamaker, "Powering e-learning in the new millennium: an overview of e-learning and enabling technology," *Information systems frontiers,* vol. 5, pp. 207-218, 2003.

[16] O. U. Press. (2016, 11.07). *E-learning*. Available: http://www.oxforddictionaries.com/definition/english/e-learning

[17] J. L. Badge, A. J. Cann, and J. Scott, "e-Learning versus e-Teaching: Seeing the pedagogic wood for the technological trees," *Bioscience Education,* vol. 5, pp. 1-11, 2005.
[18] K. C. O'Rourke, P. Rooney, and F. Boylan, "What's the Use of a VLE?," 2015.
[19] T. Rajala, M.-J. Laakso, E. Kaila, and T. Salakoski, "VILLE: a language-independent program visualization tool," in *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research-Volume 88*, 2007, pp. 151-159.
[20] D. C. Shakarian, "Beyond lecture: Active learning strategies that work," *Journal of Physical Education, Recreation & Dance,* vol. 66, pp. 21-24, 1995.
[21] N. I. Hugenholtz, E. M. De Croon, P. B. Smits, F. J. Van Dijk, and K. Nieuwenhuijsen, "Effectiveness of e-learning in continuing medical education for occupational physicians," *Occupational Medicine,* vol. 58, pp. 370-372, 2008.
[22] G. S. Mason, T. R. Shuman, and K. E. Cook, "Comparing the effectiveness of an inverted classroom to a traditional classroom in an upper-division engineering course," *IEEE Transactions on Education,* vol. 56, pp. 430-435, 2013.
[23] D. A. Bligh, *What's the Use of Lectures?*: Intellect books, 1998.
[24] T. Litzinger, L. R. Lattuca, R. Hadgraft, and W. Newstetter, "Engineering education and the development of expertise," *Journal of Engineering Education,* vol. 100, pp. 123-150, 2011.
[25] J. E. McLaughlin, M. T. Roth, D. M. Glatt, N. Gharkholonarehe, C. A. Davidson, L. M. Griffin*, et al.*, "The flipped classroom: a course redesign to foster learning and engagement in a health professions school," *Academic Medicine,* vol. 89, pp. 236-243, 2014.
[26] J. R. Savery, "Overview of problem-based learning: Definitions and distinctions," *Essential readings in problem-based learning: Exploring and extending the legacy of Howard S. Barrows,* pp. 5-15, 2015.
[27] S. A. Gallagher and J. J. Gallagher, "Using problem-based learning to explore unseen academic potential," *Interdisciplinary Journal of Problem-based Learning,* vol. 7, p. 9, 2013.
[28] V. Guillén-Nieto and M. Aleson-Carbonell, "Serious games and learning effectiveness: The case of It'sa Deal!," *Computers & Education,* vol. 58, pp. 435-448, 2012.
[29] S. Erhel and E. Jamet, "Digital game-based learning: Impact of instructions and feedback on motivation and learning effectiveness," *Computers & Education,* vol. 67, pp. 156-167, 2013.
[30] A. M. Morcke, T. Dornan, and B. Eika, "Outcome (competency) based education: an exploration of its origins, theoretical basis, and empirical evidence," *Advances in Health Sciences Education,* vol. 18, pp. 851-863, 2013.
[31] H. Rozeboom, "Competence based learning," *Bulletin of University of Agricultural Sciences and Veterinary Medicine Cluj-Napoca. Horticulture,* vol. 63, pp. 365-367, 2006.
[32] M. Brown, "An introduction to the discourse on competency-based training (CBT)," *A collection of readings related to competency-based training,* pp. 1-17, 1994.

[33] A. Soska, J. Mottok, and C. Wolff, "Playful learning in academic software engineering education," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 324-332.
[34] J. R. Frank, L. S. Snell, O. T. Cate, E. S. Holmboe, C. Carraccio, S. R. Swing*, et al.*, "Competency-based medical education: theory to practice," *Medical teacher,* vol. 32, pp. 638-645, 2010.
[35] M. Margaritis, "What pedagogical content knowledge competencies do computer science pre-service teachers have?," in *Proceedings of the tenth annual conference on International computing education research*, 2014, pp. 161-162.
[36] I. Baumgartner, "Making learning outcomes and competencies truly relevant for the practical course design and delivery: insights for university-level computing program educator," in *Joint International Conference on Engineering Education & International Conference on Information Technology ICEE/ICIT-2014*, 2014.
[37] M. Mulder, T. Weigel, and K. Collins, "The concept of competence in the development of vocational education and training in selected EU member states: a critical analysis," *Journal of Vocational Education & Training,* vol. 59, pp. 67-88, 2007.
[38] R. Klein-Collins, "Sharpening our focus on learning: The rise of competency-based approaches to degree completion," *Occasional Paper,* vol. 20, 2013.
[39] K. Ford, "Competency-based education: History, opportunities, and challenges," *UMUC Center for Innovation in Learning and Student Success,* 2014.
[40] M. Kuhn, C. Gold, and J. Abke, "A new problem based Blended Learning approach in Software Engineering," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 920-925.
[41] R. Walker, J. Voce, and J. Ahmed, "Survey of technology enhanced learning for higher education in the UK," *Oxford: Universities and Colleges Information Systems Association. Accessed January,* vol. 18, p. 2013, 2012.
[42] A. Kirkwood and L. Price, "Technology-enhanced learning and teaching in higher education: what is 'enhanced'and how do we know? A critical literature review," *Learning, media and technology,* vol. 39, pp. 6-36, 2014.
[43] D. Glušac, D. Karuović, and D. Milanov, "Massive open online courses-pedagogical overview," in *Carpathian Control Conference (ICCC), 2015 16th International*, 2015, pp. 142-146.
[44] M. Beerkens, E. Mägi, and L. Lill, "University studies as a side job: causes and consequences of massive student employment in Estonia," *Higher Education,* vol. 61, pp. 679-692, 2011.
[45] A. Fernández, D. Peralta, J. M. Benítez, and F. Herrera, "E-learning and educational data mining in cloud computing: an overview," *International Journal of Learning Technology,* vol. 9, pp. 25-52, 2014.
[46] V. Tam, E. Y. Lam, S. Fung, W. Fok, and A. H. Yuen, "Enhancing educational data mining techniques on online educational resources with a semi-supervised learning approach," in *Teaching, Assessment, and Learning for*

Engineering (TALE), 2015 IEEE International Conference on, 2015, pp. 203-206.
[47] E. Yukselturk, S. Ozekes, and Y. K. Türel, "Predicting dropout student: an application of data mining methods in an online education program," *European Journal of Open, Distance and E-learning,* vol. 17, pp. 118-133, 2014.
[48] W. Hämäläinen, T. Laine, and E. Sutinen, "Data mining in personalizing distance education courses," *Data Mining in E-learning,* vol. 4, p. 157, 2006.
[49] L. Billard and E. Diday, "Symbolic Data Analysis: Conceptual Statistics and Data Mining John Wiley," ed: Chichester, 2006.
[50] J. A. Bowden, "Competency-based learning," *Connotative Learning: The Trainer's Guide to Learning Theories and Their Practical Application to Training Design,* pp. 91-100, 2004.
[51] I. Baumgartner and V. Shankararaman, "Actively linking learning outcomes and competencies to course design and delivery: experiences from an undergraduate Information Systems program in Singapore," in *Global Engineering Education Conference (EDUCON), 2013 IEEE*, 2013, pp. 238-246.
[52] J. Warn and P. Tranter, "Measuring quality in higher education: a competency approach," *Quality in Higher Education,* vol. 7, pp. 191-198, 2001.
[53] R. A. Voorhees, "Competency-Based learning models: A necessary future," *New directions for institutional research,* vol. 2001, pp. 5-13, 2001.
[54] D. Sampson and D. Fytros, "Competence models in technology-enhanced competence-based learning," in *Handbook on information technologies for education and training*, ed: Springer, 2008, pp. 155-177.
[55] D. C. McClelland, "Testing for competence rather than for" intelligence."," *American psychologist,* vol. 28, p. 1, 1973.
[56] S. B. Parry, "The Quest for Competencies," *Training,* vol. 33, p. 48, 1996.
[57] L. Sánchez-Ruiz, M. Edwards, and E. Ballester-Sarrias, "Competence learning challenges in Engineering Education in Spain: from theory to practice," in *9th International Conference on Engineering Education (ICEE). San Juan, Puerto Rico*, 2006.
[58] A. Villa Sanchez and M. Poblete Ruiz, "Competence-based learning: a proposal for the assessment of generic competences," *Bilbao: Universidad de Deusto,* 2008.
[59] B. Mansfield, "Competence in transition," *Journal of European Industrial Training,* vol. 28, pp. 296-309, 2004.
[60] P. Perrenoud, "Développer des compétences, mission centrale ou marginale de l'université," *Congrès de l'AIPU,* 2005.
[61] F. E. Weinert, "Concept of competence: A conceptual clarification," 2001.
[62] M. Jaanus, *The interactive learning environment for mobile laboratories*: Tallinn University of Technology, 2011.
[63] K. Umbleja, "Students' grading control and visualisation in competence-based learning approach," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 287-296.
[64] M. Talbot, "Monkey see, monkey do: a critique of the competency model in graduate medical education," *Medical education,* vol. 38, pp. 587-592, 2004.

[65] J. N. Neem, "A university without intellectuals: Western Governors University and the academy's future," *Thought & Action,* p. 63, 2012.
[66] C. Carraccio, S. D. Wolfsthal, R. Englander, K. Ferentz, and C. Martin, "Shifting paradigms: from Flexner to competencies," *Academic Medicine,* vol. 77, pp. 361-367, 2002.
[67] I. K. Davies, *Objectives in curriculum design*: McGraw-Hill New York, 1976.
[68] E. A. Jones and R. A. Voorhees, "Defining and Assessing Learning: Exploring Competency-Based Initiatives. Report of the National Postsecondary Education Cooperative Working Group on Competency-Based Initiatives in Postsecondary Education. Brochure [and] Report," 2002.
[69] E. Tuxworth and J. Burke, "Competence based education and training: background and origins," *Competency based education and training,* pp. 9-22, 1989.
[70] R. Klein-Collins, "Competency-Based Degree Programs in the US," *Council for Adult and,* 2012.
[71] J. Calhoun, E. Vincent, G. Baker, P. Butler, M. Sinioris, and S. Chen, "Competency identification and modeling in healthcare leadership," *The Journal of health administration education,* vol. 21, pp. 419-440, 2003.
[72] R. A. Wolff, "Future directions for US higher education accreditation," in *Higher Education in Asia/Pacific*, ed: Springer, 2009, pp. 79-98.
[73] S. Adam and U. B. Expert, "Learning outcomes current developments in Europe: update on the issues and applications of learning outcomes associated with the Bologna process," in *Edinburgh: Scottish Government. Presented to the Bologna Seminar: Learning outcomes based higher education: the Scottish Experience (February 2008, Edinburgh)*, 2008.
[74] G. D. Kuh, N. Jankowski, S. O. Ikenberry, and J. Kinzie, "Knowing what students know and can do: The current state of student learning outcomes assessment in US colleges and universities," *Urbana, IL: University of Illinois and Indiana University, National Institute for Learning Outcomes Assessment (NILOA),* 2014.
[75] M. Milosz and E. Lukasik, "Reengineering of computer science curriculum according to technology changes and market needs," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 689-693.
[76] A. Krasniewski, "Introducing Learning Outcomes Based Engineering Curricula: A National and Institutional Perspective," in *Joint International Conference on Engineering Education & International Conference on Information Technology ICEE/ICIT-2014*, 2014.
[77] H. Schaeper, "Development of competencies and teaching–learning arrangements in higher education: findings from Germany," *Studies in Higher Education,* vol. 34, pp. 677-697, 2009.
[78] P. Beneitone. (2016). *TUNING Educational Structures in Europe*. Available: http://www.unideusto.org/tuningeu/
[79] A. K. Isaacs, "Tuning Tools and Insights for Modern Competence-Based Third-Cycle Programs," in *The European Higher Education Area*, ed: Springer, 2015, pp. 561-572.

[80] S. M. Sackey, V. R. Ancha, M. Chinyama, C. A. Onana, R. Danwe, M. M. Megahed*, et al.*, "Collaborative meta-profile development to harmonise mechanical engineering education in Africa," *Tuning Journal for Higher Education,* vol. 2, pp. 161-178, 2014.
[81] A. Serbati, "Implementation of Competence-Based Learning Approach: stories of practices and the Tuning contribution to academic innovation," *Tuning Journal for Higher Education,* vol. 3, pp. 19-56, 2015.
[82] K. Hahn and D. Teferra, "Tuning as instrument of systematic Higher Education reform and quality enhancement: the African experience," *Tuning journal for higher education,* vol. 1, pp. 127-163, 2014.
[83] P. Beneitone and M. Yarosh, "Tuning impact in Latin America: is there implementation beyond design?," *Tuning Journal for Higher Education,* vol. 3, pp. 187-216, 2015.
[84] E. Ganzglass, K. Bird, and H. Prince, "Giving Credit where Credit Is Due: Creating a Competency-Based Qualifications Framework for Postsecondary Education and Training," *Center for Law and Social Policy, Inc.(CLASP),* 2011.
[85] P. T. Ewell, "The Lumina Degree Qualifications Profile (DQP): Implications for Assessment. Occasional Paper# 16," *National Institute for Learning Outcomes Assessment,* 2013.
[86] M. Luimula and J. Roslöf, "Innovation Competences in Game Technology Education," in *Proceedings of the ICEE Conference*, 2012.
[87] K. Pata, K. Tammets, V. Tomberg, M. Al-Smadi, and M. Laanpere, "Three Curriculum Maturing Cycles in Academic Curriculum Management Systems," in *International Workshop on Learning Technology for Education in Cloud*, 2016, pp. 289-300.
[88] R. Koper and M. Specht, "Ten-competence: Life-long competence development and learning," 2006.
[89] D. López-Fernández, P. P. Alarcón, and E. Tovar, "Motivation in engineering education a framework supported by evaluation instruments and enhancement resources," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 421-430.
[90] M. Jiang, S. Parent, and D. Eastmond, "Effectiveness of web-based learning opportunities in a competency-based program," *International Journal on ELearning,* vol. 5, p. 353, 2006.
[91] B. Cheng, M. Wang, S. J. Yang, and J. Peng, "Acceptance of competency-based workplace e-learning systems: Effects of individual and peer learning support," *Computers & Education,* vol. 57, pp. 1317-1333, 2011.
[92] M. Wang, "Integrating organizational, social, and individual perspectives in Web 2.0-based workplace e-learning," *Information Systems Frontiers,* vol. 13, pp. 191-205, 2011.
[93] M. Wang, H. Jia, V. Sugumaran, W. Ran, and J. Liao, "A web-based learning system for software test professionals," *IEEE Transactions on Education,* vol. 54, pp. 263-272, 2011.
[94] V. Bremgartner and J. de Magalhaes Netto, "An adaptive strategy to help students in e-Learning systems using competency-based ontology and agents," in

Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on, 2011, pp. 978-983.
[95] V. Bremgartner and J. de Magalhães Netto, "Improving collaborative learning by personalization in Virtual Learning Environments using agents and competency-based ontology," in *2012 Frontiers in Education Conference Proceedings*, 2012, pp. 1-6.
[96] Moodle. (2016). *Competencies*. Available: https://docs.moodle.org/31/en/Competencies
[97] Y. K. Hu, T; Singhose, W; Frakes, D; Wu, B;, "Design of a Hands-on Mechantronic Project Integrated into an Introductory course," in *International Conference on Engineering Education ICEE 2015*, Zagreb, Croatia, 2015.
[98] M. H. Bredol, T;, "Making "Physical Chemistry" a more interactive and cooperative experience," in *International Conference on Engineering Education ICEE 2015*, Zagreb, Croatia, 2015.
[99] P. W. Yi, J; Huang, T; Wu, B, "An autonomous learning ability based laboratory course for undergraduate students studying in measurement and control subjects," in *International Conference on Engineering Education ICEE 2015*, Zagreb, Croatia, 2015.
[100] M. Margaritis, J. Magenheim, P. Hubwieser, M. Berges, L. Ohrndorf, and S. Schubert, "Development of a competency model for computer science teachers at secondary school level," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 211-220.
[101] E. Bender, P. Hubwieser, N. Schaper, M. Margaritis, M. Berges, L. Ohrndorf*, et al.*, "Towards a Competency Model for Teaching Computer Science," *Peabody Journal of Education,* vol. 90, pp. 519-532, 2015.
[102] V. Kukk, M. Jaanus, K. Umbleja, and O. Shvets, "Developing integrated learning environment–analysis," in *International Conference on Engineering Education, ICEE*, 2010, pp. 18-22.
[103] K. Umbleja, V. Kukk, and M. Jaanus, "Competence-based approach to learning," in *Global Engineering Education Conference (EDUCON), 2013 IEEE*, 2013, pp. 552-559.
[104] L. D. Feisel and A. J. Rosa, "The role of the laboratory in undergraduate engineering education," *Journal of Engineering Education,* vol. 94, pp. 121-130, 2005.
[105] R. V. Krivickas and J. Krivickas, "Laboratory instruction in engineering education," *Global J. Eng. Educ,* vol. 11, pp. 191-196, 2007.
[106] R. M. Felder and R. Brent, "Active learning: An introduction," *ASQ Higher Education Brief,* vol. 2, pp. 1-5, 2009.
[107] D. R. Krathwohl, "A revision of Bloom's taxonomy: An overview," *Theory into practice,* vol. 41, pp. 212-218, 2002.
[108] Z. Behzadi and A. Ghaffari, "Characteristics of online education and traditional education," *Life Science Journal,* vol. 8, pp. 54-58, 2011.
[109] E. Fabregas, G. Farias, S. Dormido-Canto, S. Dormido, and F. Esquembre, "Developing a remote laboratory for engineering education," *Computers & Education,* vol. 57, pp. 1686-1697, 2011.

[110] G. D. Peterson and L. D. Feisel, "e-Learning: The challenge for engineering education," 2002.
[111] K. Umbleja, V. Kukk, M. Jaanus, and O. Shvets, "Laboratory Experience with Competence Based Learning," in *ICEER 2013*, Marrakesh, Morocco, 2013, pp. 487–494.
[112] M. Gaebel, V. Kupriyanova, R. Morais, and E. Colucci, "E-learning in European Higher Education Institutions: Results of a mapping survey conducted in October-December 2013," *European University Association Publications: Brussels. Available online at,* 2014.
[113] A.-M. Tîrziu and C. Vrabie, "Education 2.0: e-learning methods," *Procedia-Social and Behavioral Sciences,* vol. 186, pp. 376-380, 2015.
[114] C. Pappas, "Top 10 e-learning statistics and facts for 2015 you need to know. ELearningIndustry," ed, 2015.
[115] M. Tayebinik and M. Puteh, "Blended Learning or E-learning?," *Tayebinik, M., & Puteh, M.(2012). Blended Learning or E-learning,* pp. 103-110, 2013.
[116] P. K. Udupi, P. Malali, and H. Noronha, "Big data integration for transition from e-learning to smart learning framework," in *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, 2016, pp. 1-4.
[117] É. B. de Aquino Mattos, I. M. Guimarães, A. G. da Silva, C. M. B. Barreto, and G. A. P. B. Teixeira, "Smart Device Clickers: Learning Basic Sciences and Biotechnology," *Wearable Technology and Mobile Innovations for Next-Generation Education,* p. 295, 2016.
[118] E. Mägi, A. Aidla, A. Reino, K. Jaakson, and L. Kirss, "Üliõpilaste töötamise fenomen Eesti kõrghariduses," 2011.
[119] N. Kornell, "Optimising learning using flashcards: Spacing is more effective than cramming," *Applied Cognitive Psychology,* vol. 23, pp. 1297-1317, 2009.
[120] T. M. Brinthaupt and C. M. Shin, "The relationship of academic cramming to flow experience," *College Student Journal,* vol. 35, 2001.
[121] J. W. You, "Examining the effect of academic procrastination on achievement using lms data in e-learning," *Educational Technology & Society,* vol. 18, pp. 64-74, 2015.
[122] M. Forehand, "Bloom's taxonomy," *Emerging perspectives on learning, teaching, and technology,* pp. 41-47, 2010.
[123] V. Kukk and K. Umbleja, "Analysis of forgetting in a learning environment," in *2012 13th Biennial Baltic Electronics Conference*, 2012, pp. 335-338.
[124] P. Woźniak, E. Gorzelańczyk, and J. Murakowski, "Two components of long-term memory," *Acta neurobiologiae experimentalis,* vol. 55, pp. 301-305, 1994.
[125] R. C. Atkinson and R. M. Shiffrin, *The control processes of short-term memory*: Citeseer, 1971.
[126] H. Ebbinghaus, "Memory: A contribution to experimental psychology," *Annals of neurosciences,* vol. 20, p. 155, 2013.

[127] J. T. Wixted and S. K. Carpenter, "The Wickelgren power law and the Ebbinghaus savings function," *Psychological Science,* vol. 18, pp. 133-134, 2007.
[128] J. T. Wixted, "On Common Ground: Jost's (1897) law of forgetting and Ribot's (1881) law of retrograde amnesia," *Psychological review,* vol. 111, p. 864, 2004.
[129] K. G. White, "Characteristics of forgetting functions in delayed matching to sample," *Journal of the experimental analysis of behavior,* vol. 44, pp. 15-34, 1985.
[130] J. R. Anderson, "Learning and memory," 2000.
[131] I. J. Myung, C. Kim, and M. A. Pitt, "Toward an explanation of the power law artifact: Insights from response surface analysis," *Memory & Cognition,* vol. 28, pp. 832-840, 2000.
[132] C.-M. Chen, "Intelligent web-based learning system with personalized learning path guidance," *Computers & Education,* vol. 51, pp. 787-814, 2008.
[133] C.-M. Chen, H.-M. Lee, and Y.-H. Chen, "Personalized e-learning system using item response theory," *Computers & Education,* vol. 44, pp. 237-255, 2005.
[134] V. Kukk, "Course Implementation: Value-Added Mode," in *International Workshop on Learning Technology for Education in Cloud*, 2016, pp. 265-273.
[135] D. M. Bunce, E. A. Flens, and K. Y. Neiles, "How long can students pay attention in class? A study of student attention decline using clickers," *Journal of Chemical Education,* vol. 87, pp. 1438-1443, 2010.
[136] K. Umbleja, V. Kukk, M. Jaanus, A. Udal, and B. Gordon, "Analyzes of Competence Based Approach to Learning " in *ICEE-2012*, Turku, Finland, 2012, pp. 418–424.
[137] D. M. Sluijsmans, G. J. Straetmans, and J. J. Van Merriënboer, "Integrating authentic assessment with competence-based learning in vocational education: the protocol portfolio scoring," *Journal of Vocational Education and Training,* vol. 60, pp. 159-172, 2008.
[138] M. Guenaga, J. K. Longarte, and A. Rayon, "Standardized enriched rubrics to support competeney-assessment through the SCALA methodology and dashboard," in *2015 IEEE Global Engineering Education Conference (EDUCON)*, 2015, pp. 340-347.
[139] T. Daradoumis, R. Bassi, F. Xhafa, and S. Caballé, "A review on massive e-learning (MOOC) design, delivery and assessment," in *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on*, 2013, pp. 208-213.
[140] E. Brahim, K. Mohammed, and B. Samir, "A formative assessment model within the competency-based-approach for an individualized e-learning path," *World Acad. Sci. Eng. Technol,* vol. 40, pp. 208-212, 2010.
[141] B. S. Bloom, "Handbook on formative and summative evaluation of student learning," 1971.

[142] T.-H. Wang, "Developing an assessment-centered e-Learning system for improving student learning effectiveness," *Computers & Education,* vol. 73, pp. 189-203, 2014.
[143] D. M. Sluijsmans, F. J. Prins, and R. L. Martens, "The design of competency-based performance assessment in e-learning," *Learning environments research,* vol. 9, pp. 45-66, 2006.
[144] T. P. Webb, T. R. Merkley, T. J. Wade, D. Simpson, R. Yudkowsky, and I. Harris, "Assessing competency in practice-based learning: a foundation for milestones in learning portfolio entries," *Journal of surgical education,* vol. 71, pp. 472-479, 2014.
[145] F. Altahawi, B. Sisk, S. Poloskey, C. Hicks, and E. F. Dannefer, "Student perspectives on assessment: Experience in a competency-based portfolio system," *Medical Teacher,* vol. 34, pp. 221-225, 2012.
[146] K. Umbleja, V. Kukk, M. Jaanus, and A. Udal, "New concepts of automatic answer evaluation in competence based learning," in *2014 IEEE Global Engineering Education Conference (EDUCON)*, 2014, pp. 922-925.
[147] K. Umbleja, V. Kukk, and M. Jaanus, "Answer evaluation in competence based learning," in *ICEE 2011 : An International Conference on Engineering Education*, Belfast, UK, 2011.
[148] C. L. Forgy, "Rete: A fast algorithm for the many pattern/many object pattern match problem," *Artificial intelligence,* vol. 19, pp. 17-37, 1982.
[149] P. P. Nayak, A. Gupta, and P. S. Rosenbloom, "Comparison of the Rete and Treat Production Matchers for Soar," in *AAAI*, 1988, pp. 693-698.
[150] M. L. Lee, T. W. Ling, and W. L. Low, "IntelliClean: a knowledge-based intelligent data cleaner," in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2000, pp. 290-294.
[151] M. Bali, *Drools JBoss Rules 5.0 Developer's Guide*: Packt Publishing Ltd, 2009.
[152] E. G. Agulla, L. A. Rifón, J. L. A. Castro, and C. G. Mateo, "Is my student at the other side? applying biometric web authentication to e-learning environments," in *2008 Eighth IEEE International Conference on Advanced Learning Technologies*, 2008, pp. 551-553.
[153] M. J. Zaki and W. Meira Jr, *Data mining and analysis: fundamental concepts and algorithms*: Cambridge University Press, 2014.
[154] C. Romero, S. Ventura, M. Pechenizkiy, and R. S. Baker, *Handbook of educational data mining*: CRC Press, 2010.
[155] O. Scheuer and B. M. McLaren, "Educational data mining," in *Encyclopedia of the Sciences of Learning*, ed: Springer, 2012, pp. 1075-1079.
[156] C. Romero and S. Ventura, "Educational data mining: A survey from 1995 to 2005," *Expert systems with applications,* vol. 33, pp. 135-146, 2007.
[157] C. Romero and S. Ventura, "Educational data mining: a review of the state of the art," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),* vol. 40, pp. 601-618, 2010.
[158] B. K. Baradwaj and S. Pal, "Mining educational data to analyze students' performance," *arXiv preprint arXiv:1201.3417,* 2012.

[159] A. Dutt, S. Aghabozrgi, M. A. B. Ismail, and H. Mahroeian, "Clustering algorithms applied in educational data mining," *International Journal of Information and Electronics Engineering,* vol. 5, p. 112, 2015.
[160] J. D. Gobert, M. Sao Pedro, J. Raziuddin, and R. S. Baker, "From log files to assessment metrics: Measuring students' science inquiry skills using educational data mining," *Journal of the Learning Sciences,* vol. 22, pp. 521-563, 2013.
[161] M. M. A. Tair and A. M. El-Halees, "Mining educational data to improve students' performance: a case study," *International Journal of Information,* vol. 2, 2012.
[162] K. Umbleja and M. Ichino, "Predicting Students' Behavior During an E-Learning Course Using Data Mining," in *ICL 2016 - International Conference on Interactive Collaborative Learning*, 2016.
[163] D. Zehetmeier, M. Kuhrmann, A. Böttcher, K. Schlierkamp, and V. Thurner, "Self-assessment of freshmen students' base competencies," in *2014 IEEE Global Engineering Education Conference (EDUCON)*, 2014, pp. 429-438.
[164] M. Bostock. (2015). *Data-Driven Documents*. Available: http://d3js.org/
[165] S. Inc. (2012). *Crossfilter*. Available: http://square.github.io/crossfilter/
[166] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. 20th int. conf. very large data bases, VLDB*, 1994, pp. 487-499.
[167] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*: John Wiley & Sons, 2015.
[168] F. Kaytez, M. C. Taplamacioglu, E. Cam, and F. Hardalac, "Forecasting electricity consumption: a comparison of regression analysis, neural networks and least squares support vector machines," *International Journal of Electrical Power & Energy Systems,* vol. 67, pp. 431-438, 2015.
[169] S. B. Kotsiantis, "Use of machine learning techniques for educational proposes: a decision support system for forecasting students' grades," *Artificial Intelligence Review,* vol. 37, pp. 331-344, 2012.
[170] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics,* vol. 2, pp. 433-459, 2010.
[171] I. T. Jolliffe, "Principal component analysis and factor analysis," *Principal component analysis,* pp. 150-166, 2002.
[172] L. Billard, "Symbolic data analysis: what is it?," in *Compstat 2006-Proceedings in Computational Statistics*, ed: Springer, 2006, pp. 261-269.
[173] M. Ichino and H. Yaguchi, "Generalized Minkowski metrics for mixed feature-type data analysis," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 24, pp. 698-708, 1994.
[174] E. Diday and F. Esposito, "An introduction to symbolic data analysis and the SODAS software," *Intelligent Data Analysis,* vol. 7, pp. 583-601, 2003.
[175] M. Chavent and Y. Lechevallier, "Dynamical clustering of interval data: optimization of an adequacy criterion based on Hausdorff distance," in *Classification, clustering, and data analysis*, ed: Springer, 2002, pp. 53-60.

[176] R. M. de Souza and F. d. A. De Carvalho, "Clustering of interval data based on city–block distances," *Pattern Recognition Letters,* vol. 25, pp. 353-365, 2004.
[177] H. Ling and K. Okada, "Diffusion distance for histogram comparison," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006, pp. 246-253.
[178] A. Irpino and R. Verde, "A new Wasserstein based distance for the hierarchical clustering of histogram symbolic data," in *Data science and classification*, ed: Springer, 2006, pp. 185-192.
[179] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision,* vol. 40, pp. 99-121, 2000.
[180] Y. Ma, X. Gu, and Y. Wang, "Histogram similarity measure using variable bin size distance," *Computer Vision and Image Understanding,* vol. 114, pp. 981-989, 2010.
[181] J. Kim and L. Billard, "Dissimilarity measures for histogram-valued observations," *Communications in Statistics-Theory and Methods,* vol. 42, pp. 283-303, 2013.
[182] M. Ichino and P. Brito, "A hierarchical conceptual clustering based on the quantile method for mixed feature-type data," 2015.
[183] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR),* vol. 31, pp. 264-323, 1999.
[184] S.-H. Cha, "Taxonomy of nominal type histogram distance measures," *City,* vol. 1, p. 1, 2008.
[185] U. S. G. Survey. (2013, 10.08). *Tables of Histogram Data. Climate-Vegetation Atlas of North America*. Available: http://pubs.usgs.gov/pp/p1650-b/datatables/hgtable.xls
[186] N. C. D. Center. (2014, 10.08). *Time Bias Corrected Divisional Temperature-Precipitation-Drought Index* Available: http://www1.ncdc.noaa.gov/pub/data/cirs/drd/drd964x.tmp.txt
[187] T. Foundation. (2016, 10.08). *The R Project for Statistical Computing*. Available: https://www.r-project.org/
[188] A. Inselberg and B. Dimsdale, "Parallel coordinates," in *Human-Machine Interactive Systems*, ed: Springer, 1991, pp. 199-233.
[189] M. Ichino, "The quantile method for symbolic principal component analysis," *Statistical Analysis and Data Mining,* vol. 4, pp. 184-198, 2011.

ABSTRACT

The aim of this thesis is to develop a complete framework for competence-based learning (CBL) in e-learning, implement it and analyse the changes in students' behaviour, results and usage of competence in the system. CBL is a learning methodology that tries to overcome the problems of classical topic-based learning by concentrating on what a learner can actually do and knows as a result of learning.

CBL has been covered by many researchers but the main focus has been on curriculum development and there have been rarely any attempts to connect the desired learning outcomes with results from actual learning process. Furthermore, there have been very few attempts trying to implement CBL to technology enchanted learning. Most of the time, the definition of competences has been very vague and assessment has not been adapted to CBL and its characteristics.

Proposed framework for CBL in this thesis uses a wide range of benefits from technology enchanted learning. To ensure concrete knowledge, tasks were divided into smaller units enabling repetition in order to move new knowledge from short term memory to long term memory. Furthermore, memory model is applied to students' results, lowering them over time, thus mimicking students' forgetting. 128-level scale is used for grading competences instead of classical 6-level scale to avoid results' discretization. As every student has a different learning style, proposed framework supports personalization by using self-regulated learning without deadlines. Framework also uses personal learning path, using learning control function that applies multiply parameters for input (including forgetting parameters). There is overlap in courses to give students a chance to choose what they wish to learn, while taking into account students' previously acquired competences. Methodology for switching from topic based e-learning to CBL has been proposed in the thesis.

This thesis covers the assessment of competences in proposed framework. Answer evaluation is automatic for constant, uninterrupted learning process. All the competences in tasks are graded separately and independently by using algorithms mimicking students' answering process. The algorithm generates feedback according to the mistakes made, lowers the impact of repeated mistakes and then ensures mistakes are not carried over.

This thesis focuses on the analysis of the proposed and then implemented framework. Currently there are more than 500 competences in the learning system. Competences have been adequately covered for tasks on different difficulty levels. Analysis on the data gathered from the system before and after the change in learning method shows that average results have improved remarkably and CBL has achieved its goal of firmer and more concrete acquired knowledge. Repetition has been achieved but due to splitting tasks into smaller units, the amount of time spent studying is reasonable and more in fit with courses' credit units. The analysis also revealed that students' behaviour is

changing with time. Three distinct students' behaviour periods were found from the analysed timeframe.

The analysis also showed that despite better results and firmer knowledge, the percentage of students who did not finish the course was very high. Self-regulated learning worked well in topic-based approach where amount of work towards the grade was linear and easily predictable. With CBL, the amount of work required to finish the course varied a lot and was not linear. To tackle the problem, multiply visualization tools were developed to help students follow their progress. Despite initial positive feedback, the data showed no real impact to the learning behaviour. Students stated that they want answers, not use provided information to find them by themselves.

Therefore, a model predicting students' study behaviour during the course was developed. Histogram valued symbolic data was used to describe the learning behaviour and a new method, derived from mixed-feature type Cartesian space model for finding dissimilarities between histogram valued data, was developed as no general method for histograms with varying bins and bin widths existed beforehand. Also the process of clustering based on compactness for proposed methodology was described. Using PCA, 9 most important features were deducted from the e-learning system to build the model. Clustering all the students together did not yield adequate results as every course has different study patterns but clustering students by courses yielded good clusters. Despite courses having different patterns, some of the clusters were common over all courses. Developed model gave good results for predicting final grade and learning behaviour through the course. On the other hand, the model was found to be not very effective on predicting finishing time.

Therefore, it can be said that as a result of this thesis, a process of transforming topic-based learning environment into CBL has been developed with proposed methodology. The changes in students' learning behaviour, results and competence usage have been analysed. The problem of high drop-outs due to non-linear amount of work required to finish the course was discovered and tools to help students were developed. When those tools did not yield desired results, a model to predict students' progress based on previous students' study behaviours was developed. To use histogram valued data to describe students' learning process and clustering, a new method for dissimilarity was developed. Achieved model worked well to predict students' final grade.

KOKKUVÕTE

Antud väitekirja eesmärgiks oli välja töötada terviklik platvorm kompetentsi põhiseks õppeks (KPÕ) e-õppe keskkonnas, see implementeerida ning analüüsida muutusi õpilaste käitumises, tulemustes ja kompetentside kasutuses süsteemis. CBL on õppimise metoodika, mis keskendub sellele mida õppija päriselt õppeprotsessi tulemusena oskab.

KPÕd on kajastatud paljude teadlaste poolt, kuid enamikel juhtudel on rõhk olnud kompetentside sidumisega õppekavade arendamisel ning harva on seotud õpiväljundeid reaalse õppeprotsessi tulemustega. Lisaks on KPÕt rakendatud tehnoloogiaga toetatud õppes vaid üksikutel juhtudel. Samuti on kompetentsid enamikel juhtudel väga umbmääraselt defineeritud ja hindamist ei ole kohandatud KPÕle ja selle eripäradele.

Väitekirjas väljatöötatud raamistik kasutab laialdaselt KPÕks tehnoloogia võimalusi parima õppetulemuse saavutamiseks. Ülesanded on jagatud väiksemateks ühikuteks, et soodustada kordamist, mis liigutaks omandatud teadmised lühimälust pikaajalisse mällu. Samuti on süsteemis implementeeritud mälumudel, mis langetab aja jooksul õppijate taset, imiteerides õpilaste unustamist. Süsteemis on kasutusel 128 tasemeline hindamine tavalise 6 taseme asemel, et vältida diskreetimist. Kuna igal õpilasel on erinev õppimisstiil, siis toetab pakutud raamistik personaalset lähenemist, kasutades tähtaegadeta ise-reguleeritud õppimist - personaalset teekonda läbi õpiobjektide, kasutades kontrollfunktsiooni, mis võtab sisendiks mitmeid parameetreid (kaasa arvatud unustamise parameetrid). Kursustel on üle täituvus, mis laseb õppijal valida, mida ta täpselt õppida soovib ning süsteem võtab arvesse ka õppijate varem omandatud kompetentse. Väitekirjas on välja toodud meetod, kuidas klassikalist e-õppe keskkonda viia üle KPÕle.

Väitekiri kajastab hindamist KPÕ raamistikus. Hindamine on katkematuks õppeprotsessiks automaatne. Kõiki kompetentse, mis ülesannetes esinevad, hinnatakse eraldi ja iseseisvalt kasutades algoritme, mis imiteerivad õpilase käitumist. Algoritm genereerib tagasisidet vastavalt tehtud vigadele, vähendab korduvate vigade mõju hindele ning kannab hoolt, et viga ühes kohas ei kandu vastuses üle.

Töös analüüsitakse KPÕ raamistiku kasutamist ja selle mõju. Hetkel on süsteemis ligikaudu 500 kompetentsi, mis on kaetud piisavalt ülesannetega erinevatel raskusastmetel. Enne ja pärast õppemeetodi vahetust kogutud andmete analüüs näitab, et keskmine tulemus on märgatavalt tõusnud ja KPÕ on saavutanud oma eesmärgi. Samuti on saavutatud kordamine, aga kuna ülesanded sai jagatud väiksemateks osadeks, siis aeg veedetud õppesüsteemis ei ole märgatavalt tõusnud, vaid vastab rohkem ainepunktidele. Analüüs tuvastas, et õpilaste käitumine on ajas muutuv – andmetest leiti kolm erinevate käitumismustritega ajaperioodi.

Samuti näitas analüüs, et hoolimata paramatest tulemustest ei ole vähenenud õpilaste protsent, kes ainet ei läbi. Ise-reguleeritud õppimine toimis edukalt klassikalises õppes, kus vajamineva töö kogus oli lihtsasti ette aimatav. KPÕga varieerub vajaliku töö maht tugevasti ja see ei ole lineaarselt ennustatav. Selleks, et aidata õpilastel aimata vajamineva töö mahtu, sai välja töötatud erinevaid visualiseerimistööriistu õppija progressi jälgimiseks. Hoolimata esialgsest positiivsest tagasisidest, näitas andmeanalüüs, et neil tööriistadel ei olnud olulist mõju õpilaste käitumisele. Õpilased on öelnud, et nad soovivad konkreetseid vastuseid, mitte informatsiooni, millest ise järeldusi teha.

Selleks, et võimaldada õpilastele konkreetseid järeldusi, sai välja arendatud õpilaste käitumist ennustav mudel. Histogramm kujul sümboolsed andmed sai valitud kirjeldamaks õpilaste käitumist ning uus meetod leidmaks histogrammidena esitatud andmete vahelist erinevust sai välja töötatud, kuna senimaani ei eksisteerinud selleks üldist meetodit, mis lubaks erineval arvul ja erineva laiusega tulpasid. Samuti sai välja töötatud uuel meetodil baseeruv klasterdamine. Kasutades peakomponendi analüüsi sai tuvastatud 9 kõige olulisemat omadust, mida kasutati loomisel. Kõigi õpilaste andmete koos grupeerimine ei andnud head tulemust, sest igal ainel on erinevad õppimisemustrid ja sisu. Grupeerides õpilasi kursuste kaupa andis aga häid tulemusi ning hoolimata sellest, et ained on erinevad, siis mõned grupid olid kõigi ainete peale sarnased. Välja töötatud mudel sobis lõpphinde ennustamiseks, aga ei olnud väga tõhus kursuse lõpetamise aja ennustamiseks.

Antud väitekirja tulemusena sai kirjeldatud protsessi, kuidas viia klassikaline õppekeskkond üle KPÕle. Töö tulemusena sai analüüsitud muutusi õpilaste käitumises, tulemustes ja kompetentside kasutuses. Analüüsi käigus sai tuvastatud probleem, et paljudel õpilastel jääb kursus lõpetamata, kuna nad ei suuda hinnata vajamineva töö mahtu. Selle vähendamiseks sai välja töötatud erinevaid progressi visualiseerimistööriistu ning mudel, mis ennustab tulemusi eelmiste õpilaste käitumise baasil. Selleks, et kasutada mudelis histogrammi kujul andmeid, sai välja töötatud mudel sellisel kujul kirja pandud andmete erinevuse määramiseks.

ACKNOWLEDGEMENTS

The author wants to express her deep gratitude to her supervisors, Dr. Vello Kukk and Dr. Innar Liiv, for their support and guidance throughout the thesis work. Author would also like to thank Dr. Manabu Ichino for fruitful collaboration, guidance and invaluable advice connected to symbolic data analysis. Furthermore, author wants to thank her research colleagues Dr. Martin Jaanus, Dr. Andres Udal and Dr. Leo Võhandu for collaborations, advice and guidance.

The author also wants to thank her family for their lifelong support, especially her father, without whom she would have never considered taking up computer science, and her husband, for his valued advice on grammar while writing the thesis.

This work has been partially supported by Estonian Doctoral School in Informatics and Communication Technology, by Estonian IT Academy (Study IT program) and by European Social Fund's Doctoral Studies and Internationalization Program DoRa.

APPENDIX A – Radar chart algorithm

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.util.ArrayList;


public class vis1 {

        static int cnt=131; //nr of competences in a course
        static int[][] a=new int[cnt][3];

        public static void main(String[] args) {
                read_from_file("isc0010.txt"); //file with pairs and counts
in descending order
                find_line();
        }

        /**
         * Main function that finds competences appearing often in same
tasks
         */
         private static void find_line() {
                ArrayList<Integer> exists=new ArrayList();
                ArrayList<Holder> b=new ArrayList();
                for (int i=0;i<cnt;i++) { //for all competences
                        if(exists.contains(a[i][0]) &&
exists.contains(a[i][1])){ //both already exist
                                        int o1=0;
                                        int o2=0;
                                        //find in which group they exist
                                        for(int j=0;j<b.size();j++){
                                                System.out.println(b.get(j).line());
                                                if(b.get(j).contains(a[i][0])){
                                                        o1=j;
                                                }
                                                if(b.get(j).contains(a[i][1])){
                                                        o2=j;
                                                }
                                        }
                                        //System.out.println(o1+" "+o2);
                                        if(o1!=o2){ //if they are not in the same
group
                                                //find best way how to combine
groups
                                                int
idx=find_new_group(b.get(o1),b.get(o2),a[i][0],a[i][1],i,a[i][2]);
                                                if(idx==0){
                                                        for(int
j=0;j<b.get(o2).a.size();j++){

        b.get(o1).a.add(0,b.get(o2).a.get(j));
                                                        }
                                                }else if(idx==1){
                                                        for(int j=b.get(o2).a.size()-
1;j>=0;j--){

        b.get(o1).a.add(0,b.get(o2).a.get(j));
                                                        }
                                                }else if(idx==2){
                                                        for(int
j=0;j<b.get(o2).a.size();j++){

        b.get(o1).a.add(b.get(o2).a.get(j));
```

```java
                                                    }
                                            }else{
                                                    for(int j=b.get(o2).a.size()-
1;j>=0;j--){

        b.get(o1).a.add(b.get(o2).a.get(j));
                                                    }
                                            }
                                            b.remove(o2);
                                    }
                            }else if(exists.contains(a[i][0])){ //if first
exists in a group
                                    for(int j=0;j<b.size();j++){
                                            if(b.get(j).contains(a[i][0])){
                                                    //add second one to existing
group (to the front or to the end)

        if(find_new(b.get(j),a[i][0],a[i][1],i,a[i][2])==0){

        b.get(j).a.add(0,a[i][1]);
                                                    }else{

        b.get(j).a.add(a[i][1]);
                                                    }
                                                    //update existing competences
                                                    exists.add(a[i][1]);
                                                    break;
                                            }
                                    }
                            }else if(exists.contains(a[i][1])){ //if second one
exsists in a group
                                    for(int j=0;j<b.size();j++){
                                            if(b.get(j).contains(a[i][1])){
                                                    //add first one to the group
(to the front or to the end)

        if(find_new(b.get(j),a[i][1],a[i][0],i,a[i][2])==0){

        b.get(j).a.add(0,a[i][0]);
                                                    }else{

        b.get(j).a.add(a[i][0]);
                                                    }
                                                    //update existing competences
                                                    exists.add(a[i][0]);
                                                    break;
                                            }
                                    }
                            }else{ //if none are in a group, make new group for
them
                                    b.add(new Holder(a[i][0],a[i][1]));
                                    exists.add(a[i][0]);
                                    exists.add(a[i][1]);
                            }
                    }
                    //print results
                    for( int i=0;i<b.size();i++){
                            System.out.println(b.get(i).line());
                    }
        }

         /**
          * Finds how to combine two groups
          * @param b1 - group 1
```

```
          * @param b2 - group 2
          * @param el1 - competence in group 1
          * @param el2 - competence in group 2
          * @param counter - nr of the pair in file
          * @param nr_of_pairs - nr of times pair exists together
          * @return best way how to combine groups
          */
         private static int find_new_group(Holder b1, Holder b2, int el1,
int el2, int counter, int nr_of_pairs) {

                //data about after and before the point are saved
                int[] cnt=new int[4];
                int[] many=new int[4];
                //find locations of both elements in their groups
                int o1=b1.a.indexOf(el1);
                int o2=b2.a.indexOf(el2);
                //first halves of both groups
                for(int i=0;i<=o1;i++){
                        for(int j=0;j<=o2;j++){

        cnt[0]+=similarity(b1.a.get(i),b2.a.get(j),counter);
                                many[0]++;
                        }
                }
               //first half of first group and second half of second group
                for(int i=0;i<=o1;i++){
                        for(int j=o2;j<b2.a.size();j++){

        cnt[1]+=similarity(b1.a.get(i),b2.a.get(j),counter);
                                many[1]++;
                        }
                }
                //first half of second group and second half of first
group

                for(int i=o1;i<b1.a.size();i++){
                        for(int j=0;j<=o2;j++){

        cnt[2]+=similarity(b1.a.get(i),b2.a.get(j),counter);
                                many[2]++;
                        }
                }
                //second halves of both group
                for(int i=o1;i<b1.a.size();i++){
                        for(int j=o2;j<b2.a.size();j++){

        cnt[3]+=similarity(b1.a.get(i),b2.a.get(j),counter);
                                many[3]++;
                        }
                }

                //find best matcj
                double best=(double)cnt[0]/many[0];
                int idx=0;
                for(int i=0;i<4;i++){
                        if((double)cnt[i]/many[i]>best){
                                best=(double)cnt[i]/many[i];
                                idx=i;
                        }
                }
                return idx;
        }

         /**
          * @param g - group where existing object is
```

```
         * @param existing - existing competence
         * @param new_comp - new competence
         * @param counter - nr of the pair in file
         * @param nr_of_pairs - nr of times pair exists together
         * @return if new competence will be added to the start or to the
end
         */
       private static int find_new(Holder g,int existing, int new_comp,
int counter, int nr_of_pairs){
               int loc=0; //point where existing element in a group is
               //data about after and before the point are saved
               int[] cnt=new int[2];
               int[] many=new int[2];
               //for all the elements in a group
               for(int i=0;i<g.a.size();i++){
                       if(g.a.get(i)==existing){ //if it is the one in a
group
                               cnt[loc]+=nr_of_pairs;
                               many[loc]++;
                               loc=1;
                               cnt[loc]=nr_of_pairs;
                               many[loc]++;
                       }else{ //in other cases, find how many times they
exist together

       cnt[loc]+=similarity(g.a.get(i),new_comp,counter);
                              many[loc]++; //count how many
                       }
               }
               if ((double)cnt[0]/many[0]>=(double)cnt[1]/many[1]){ //to
the beginning of the group
                       return 0;
               }else{ //to the end of the group
               return 1;
               }
        }
       /**
        *
        * @param existing - existing competence in a group
        * @param new_comp - new competence to be added
        * @param counter - point where pair was found
        * @return number of times competences are together
        */
       private static int similarity(int existing, int new_comp, int
counter) {
              for(int i=counter+1;i<cnt;i++){
                     if(a[i][0]==existing && a[i][1]==new_comp ||
a[i][0]==new_comp && a[i][1]==existing){ //if they exist together
                            return a[i][2];
                     }
              }
              return 0; //if no, return 0
       }

       /**
        * read from file
        * @param name - filename of the file where to read pairs
        */
       public static void read_from_file(String name){
               try{
                   BufferedReader br = new BufferedReader(new
FileReader(name));
                   String line;
                   String b[];
```

```
                        int count=0;
                        while((line = br.readLine())!= null){ //while lines
                            b=line.split("\t"); //split
                            a[count][0]=Integer.parseInt(b[0]);
                            a[count][1]=Integer.parseInt(b[1]);
                            a[count][2]=Integer.parseInt(b[2]);
                            count++; //
                        }
                        //System.out.println("count:"+count);
                        br.close();
                    } catch(Exception e){
                            System.out.println(e);
                    }
               }
}
```

Class Holder:

```
import java.util.ArrayList;

/**
 * saves merged competences in a group
 */
public class Holder {

        public ArrayList<Integer> a=new ArrayList();

        public Holder(int i, int j) {
                a.add(i);
                a.add(j);
        }

        /**
         * @return line with similar competenes
         */
        public String line(){
                String t="";
                for(int i=0;i<a.size();i++){
                        t=t+a.get(i)+",";
                }
                return t;
        }

        /**
         * @param b competence
         * @return if competence is in the arrayList
         */
        public Boolean contains(int b){
                return a.contains(b);
        }
}
```

APPENDIX B - Algorithm for finding dissimilarity between two histograms

Assume we have two histogram valued objects ω_i, and ω_l described by p features F_j, (j = 1, 2,…, p). Both of these objects contain p histograms. For j-th feature, these histograms are defined by: $E_{ij} = \{[a_{ijk}, b_{ijk}), p_{ijk}; k = 1,2 \ldots n_{ij}\}$ and $E_{lj} = \{[a_{ljk}, b_{ljk}), p_{ljk}; k = 1,2 \ldots n_{lj}\}$.

We can find the dissimilarity between objects ω_i, and ω_l by the following algorithm (pseudo-code written in java/c style):

```
Dissimilarity(ωi, ωl){
   global_dissimilarity = 0;

   for(j=1 to p){ //for every feature
     dissimilarity=0 //dissimilarity for current feature
     culmutiveProbMeet=0
     culmutiveProbJoin=0
     k=1; //marker for histogram Eij for feature j
     m=1; //marker for histogram Elj for feature j

      start=min(next(Eij,k),next(Elj,m)) //find first bin with probability greater than 0
      if(start== next(Eij,k)){
         k++ //move marker for Eij
      }
     if( start== next(Elj,m)) { //both may have to move if they have the same intervals
         m++ //move marker for Elj
      }

      while(k<=nij + 1 && m<=nlj + 1){ //+1 will give us the end of last bin.
         end= min(next(Eij,k),next(Elj,m)) //find next bin start from histograms
         culmutiveProbMeet+=
min(probability(Eij,k,start,end),probability(Elj,m,start,end))
         culmutiveProbJoin+=
(probability(Eij,k,start,end)+probability(Elj,m,start,end))/2
         meet=(end-start)*culmutiveProbMeet
         join=(end-start)*culmutiveProbJoin
        dissimilarity+=join-meet //calculate dissimilarity between those areas
        if(start== next(Eij,k)){
           k++ //move marker for Eij
         }
        if( start== next(Elj,m)) { //both may have to move if they have the same intervals
           m++ //move marker for Elj
        }
        start=end
```

```
        }

        if(k>n_ij+1){ //histogram E_ij reached end before E_lj
           while(m<=n_lj + 1){
              end=next(E_lj,m) //only one histogram has bins
              culmutiveProbJoin+=probability(E_lj,m,start,end)/2   //No   update   to
meet as its probability is 0
              dissimilarity+=(end-start)*culmutiveProbJoin-(end-
start)*culmutiveProbMeet
              start=end;  m++
           }
        }else{ //histogram E_lj reached end before E_ij
           while(k<=n_ij+1){
              end=next(E_ij,k)
              culmutiveProbJoin+=probability(E_ij,k,start,end)/2 // No update to meet
as its probability is 0
              dissimilarity+=(end-start)*culmutiveProbJoin-(end-
start)*culmutiveProbMeet
              start=end;  k++
           }
        }global_dissimilarity+=dissimilarity/|F_j end-F_j start|            //normalization,
division by feature length
    }return global_dissimilarity/p //normalization, division by number of features
} //end of Dissimilarity(ω_i, ω_l)

probability(E_ij,k,start,end){ //return probability on current comparable position
    if(k<=1 || k>n_ij+1 || end<a_ijk){ //out of index or current bin smaller than next
bin for E_ij
        return 0
    }else{ //start>=a_ijk and end<=b_ijk
        return  p_ij(k-1) * (end - start)/(b_ijk - a_ijk)  //k  points  to  the  end  of
current interval/start of next interval, therefore k-1 is used
    }
}

next(E_ij,k){ //return next bin position
    if(k<=n_ij){
        return a_ijk //return start of next bin
    }elseif(k==n_ij + 1){ //last bin reached
        return b_ijn_ij //return last bin's end
    }
}
```

APPENDIX C - Proof of the triangle law

For any set of objects ω_u, ω_v, and ω_r containing histograms E_{uj}, E_{vj} and E_{rj} for feature j following 8 different cases how histograms could be connected can be identified as can be seen in Fig. 8.1:

1. E_{uj} and E_{vj} have no intersection and E_{rj} is situated in area between them
2. E_{uj} and E_{vj} have no intersection and E_{rj} is not situated in area between them
3. E_{uj} and E_{vj} have no intersection but E_{rj} has intersection with E_{vj}
4. E_{uj} and E_{vj} have no intersection but E_{rj} has intersection with E_{uj}
5. E_{uj} and E_{vj} have no intersections but both of them have intersection with E_{rj}
6. E_{uj} and E_{vj} have intersection but both of them have no intersections with E_{rj}
7. E_{uj} and E_{vj} have intersection and both of them have intersection with E_{rj}. E_{rj} covers area that is larger than area of E_{uj} and E_{vj} intersection.
8. E_{uj} and E_{vj} have intersection and both of them have intersection with E_{rj} that is within the area of E_{uj} and E_{vj} intersection.

Figure 8.1: Graphical representation of 8 cases how histograms E_{uj}, E_{vj} and E_{rj} may appear together.

Dissimilarity between histograms is defined as size of join subtracted size of meet.

Case 1: It is clearly visible that all the meets have size 0 and there are no overlaps. Therefore from proposition 2: $\Phi(E_{uj}, E_{rj}) = 0.5 \times |E_{uj}| + 0.5 \times |E_{rj}| + 0.5 \times |b_{ujnr_u} - b_{rjnr_r}|$, $\Phi(E_{rj}, E_{vj}) = 0.5 \times |E_{vj}| + 0.5 \times |E_{rj}| + 0.5 \times |b_{rjnr_r} -$

$b_{vjnr_v}|$and$\Phi\left(E_{uj},E_{vj}\right)=0.5\times|E_{vj}|+0.5\times|E_{uj}|+0.5\times|b_{vjnr_v}-b_{ujnr_u}|$. The triangle law takes the following form:

$\Phi\left(E_{uj},E_{vj}\right)=0.5\times|b_{vjnr_v}-b_{ujnr_u}|\leq|E_{rj}|+0.5\times|b_{ujnr_u}-b_{rjnr_r}|+0.5\times|b_{rjnr_r}-b_{vjnr_v}|=\Phi\left(E_{uj},E_{rj}\right)+\Phi\left(E_{rj},E_{vj}\right)$

where we should note that $|b_{vjnr_v}-b_{ujnr_u}|=|b_{rjnr_r}-b_{ujnr_u}|+|b_{vjnr_v}-b_{rjnr_r}|$.

Case 2: Same as case 1.

Case 3: $\Phi\left(E_{uj},E_{rj}\right)=0.5\times|E_{uj}|+0.5\times|E_{rj}|+0.5\times|b_{ujnr_u}-b_{rjnr_r}|$, $\Phi\left(E_{uj},E_{vj}\right)=0.5\times|E_{vj}|+0.5\times|E_{uj}|+0.5\times|b_{vjnr_v}-b_{ujnr_u}|$and $0\leq\Phi\left(E_{rj},E_{vj}\right)=|E_{rj}\boxplus E_{vj}|-|E_{rj}\boxtimes E_{vj}|\leq 0.5\times|E_{vj}|+0.5\times|E_{rj}|+0.5\times|b_{vjnr_v}-b_{rjnr_r}|$. The triangle law takes the following form for maximum case:

$\Phi\left(E_{uj},E_{vj}\right)=0.5\times|b_{vjnr_v}-b_{ujnr_u}|\leq|E_{rj}|+0.5\times|b_{ujnr_u}-b_{rjnr_r}|+0.5\times|b_{rjnr_r}-b_{vjnr_v}|=\Phi\left(E_{uj},E_{rj}\right)+\Phi\left(E_{rj},E_{vj}\right)$,

where we should note that $|b_{vjnr_v}-b_{ujnr_u}|\leq|b_{rjnr_r}-b_{ujnr_u}|+|b_{vjnr_v}-b_{rjnr_r}|$.

For the minimum case ($E_{rj}=E_{vj}$), triangle law takes following for:

$\Phi\left(E_{uj},E_{vj}\right)=0.5\times|E_{vj}|+0.5\times|E_{uj}|+0.5\times|b_{vjnr_v}-b_{ujnr_u}|\leq 0.5\times|E_{uj}|+0.5\times|E_{rj}|+0.5\times|b_{ujnr_u}-b_{rjnr_r}|+0=\Phi\left(E_{uj},E_{rj}\right)+\Phi\left(E_{rj},E_{vj}\right)$

As $0.5\times|b_{vjnr_v}-b_{ujnr_u}|-0.5\times|b_{ujnr_u}-b_{rjnr_r}|=0$ as $E_{rj}=E_{vj}$ and $|E_{rj}|=|E_{vj}|$, it holds.

Case 4: Similar to case 3.

Case 5: $0\leq\Phi\left(E_{uj},E_{rj}\right)\leq 0.5\times|E_{uj}|+0.5\times|E_{rj}|+0.5\times|b_{ujnr_u}-b_{rjnr_r}|$, $0\leq\Phi\left(E_{rj},E_{vj}\right)\leq 0.5\times|E_{vj}|+0.5\times|E_{rj}|+0.5\times|b_{vjnr_v}-b_{rjnr_r}|$ and $\Phi\left(E_{uj},E_{vj}\right)=0.5\times|E_{vj}|+0.5\times|E_{uj}|+0.5\times|b_{vjnr_v}-b_{ujnr_u}|$. For the maximum case, it is similar to case 1. As E_{uj} has no intersection with E_{vj}, both minimum cases cannot appear in the same time. Assume the minimum case is ($E_{rj}=E_{vj}$) and therefore it is similar to case 3. Triangle rule holds.

Case 6: $\Phi\left(E_{uj},E_{rj}\right)=0.5\times|E_{uj}|+0.5\times|E_{rj}|+0.5\times|b_{ujnr_u}-b_{rjnr_r}|$, $\Phi\left(E_{rj},E_{vj}\right)=0.5\times|E_{vj}|+0.5\times|E_{rj}|+0.5\times|b_{vjnr_v}-b_{rjnr_r}|$and $0\leq\Phi\left(E_{uj},E_{vj}\right)\leq 0.5\times|E_{vj}|+0.5\times|E_{uj}|+0.5\times|b_{vjnr_v}-b_{ujnr_u}|$. For the maximum case, it is similar to case 1. For the minimum case ($E_{vj}=E_{uj}$) triangle rule takes the following form:

$\Phi\left(E_{uj}, E_{vj}\right) = 0 \ \leq \left|E_{rj}\right| + 0.5 \times |E_{uj}| + 0.5 \times |b_{ujnr_u} - \ b_{rjnr_r}| + 0.5 \times |E_{vj}| + 0.5 \times |b_{vjnr_v} - \ b_{rjnr_r}| = \ \Phi(E_{uj}, E_{rj}) + \Phi\left(E_{rj}, E_{vj}\right)$

It holds.

Case 7: $0 \leq \Phi\left(E_{uj}, E_{vj}\right) \leq 0.5 \times |E_{vj}| + 0.5 \times \left|E_{uj}\right| + 0.5 \times |b_{vjnr_v} - b_{ujnr_u}|$, $0 \leq \Phi(E_{uj}, E_{rj}) \leq 0.5 \times |E_{uj}| + 0.5 \times \left|E_{rj}\right| + 0.5 \times |b_{ujnr_u} - \ b_{rjnr_r}|$ and $0 \leq \Phi\left(E_{rj}, E_{vj}\right) \leq 0.5 \times |E_{vj}| + 0.5 \times \left|E_{rj}\right| + 0.5 \times |b_{vjnr_v} - \ b_{rjnr_r}|$. Maximum case is similar to case 1. For minimum case ($E_{vj} = E_{uj} = E_{rj}$), triangle rule holds as all dissimilarities are 0.

Triangle rule holds.

Case 8: Same as case 7.

APPENDIX D – Application on Real Data Sets

To verify that the proposed dissimilarity measure works and gives adequate results, two well-known data sets used by data-miners are used for verification.

Hardwood Data

In this section, a small example using hardwood data by U.S. Geological survey [185] is represented. Ten groups of trees (5 species with west and east coast groups) were chosen with eight features. The same data was used in [182] and in [189]. In the following, we summarize the procedures of the proposed clustering methods based on this data.

The following *eight* features describe the Hardwood data:

- F_1: Annual Temperature (*ANNT*) (ºC);
- F_2: January Temperature (*JANT*) (ºC);
- F_3: July Temperature (*JULT*) (ºC);
- F_4: Annual Precipitation (*ANNP*) (mm);
- F_5: January Precipitation (*JANP*) (mm);
- F_6: July Precipitation (*JULP*) (mm);
- F_7: Mean Temperature of the Coldest Month (°C);
- F_8: Growing Degree Days on 5ºC base ×1000 (*GDC5*);
- F_9: Moisture Index (*MITM*).

The features are given by cumulative percentages that are transformed into histograms with variety width bins. The number of bins is constant as are the percentages for k-th bin (k =1 to 9). Table 1 shows histogram-valued variable for feature F_1. The data format for other features ($F_2 – F_9$) is the same as shown in Table 9.1.

Table 9.1: Data from hardwood set transformed into histogram format for feature F_1*: Annual Temperature.*

Taxon name	Histogram for F_1: Annual Temperature (*ANNT*) (ºC)
ACER EAST	{[-2.3, 0.6]0.1; [0.6, 3.8]0.15; [3.8, 9.2]0.25; [9.2, 14.4]0.25; [14.4, 17.9]0.15; [17.9, 23.8]0.1}
ACER WEST	{[-3.9, 0.2]0.1; [0.2, 1.9]0.15; [1.9, 4.2]0.25; [4.2, 7.5]0.25; [7.5, 10.3]0.15; [10.3, 20.6]0.1}
ALNUS EAST	{[-10.2, -4.4]0.1; [-4.4, -2.3]0.15; [-2.3, 0.6]0.25; [0.6, 6.1]0.25; [6.1, 15.0]0.15; [15.0, 20.9]0.1}
ALNUS WEST	{[-12.2, -4.6]0.1; [-4.6, -3.0]0.15; [-3.0, 0.3]0.25; [0.3, 3.2]0.25; [3.2, 7.6]0.15 ;[7.6, 18.7]0.1}

FRAXINUS EAST	{[-2.3, 1.4]0.1; [1.4, 4.3]0.15; [4.3, 8.6]0.25; [8.6, 14.1]0.25; [14.1, 17.9]0.15; [17.9, 23.2]0.1}
FRAXINUS WEST	{[2.6, 9.4]0.1; [9.4, 11.5]0.15; [11.5, 17.2]0.25; [17.2, 21.2]0.25; [21.2, 22.7]0.15; [22.7, 24.4]0.1
JUGLANS EAST	{[1.3, 6.9]0.1; [6.9, 9.1]0.15; [9.1, 12.4]0.25; [12.4, 15.5]0.25; [15.5, 17.6]0.15; [17.6, 21.4]0.1}
JUGLANS WEST	{[7.3, 12.6]0.1; [12.6, 14.1]0.15; [14.1, 16.3]0.25; [16.3, 19.4]0.25; [19.4, 22.7]0.15; [22.7, 26.6]0
QUERCUS EAST	{[-1.5, 3.4]0.1; [3.4, 6.3]0.15; [6.3, 11.2]0.25; [11.2, 16.4]0.25; [16.4, 19.1]0.15; [19.1, 24.2]0.1}
QUERCUS WEST	{[-1.5, 6.0]0.1; [6.0, 9.5]0.15; [9.5, 14.6]0.25; [14.6, 17.9]0.25; [17.9, 19.9]0.15; [19.9, 27.2]0.1}

For clustering, proposed dissimilarity is used for metric. For example, dissimilarity between objects "Acer East" and "Acer West" is calculated using formula (6.15). The dissimilarity for each feature is calculated and normalized. Then those intermediate dissimilarities are summed up and normalized again. Therefore, we use the following procedure to calculate the dissimilarity between two histograms for "Acer East"$_1$ and "Acer West"$_1$ under feature F_1:

- Join and meet are found according to the procedure given in Appendix 1. We first set current position counters, k and m, to *zero*. Also, variables holding current cumulative percentages for join and meet are set to zero. The number of bins of these histograms is *six*.
- The minimal starting position of histograms "Acer East"$_1$ and "Acer West"$_1$ is found. In current case, it is -3.9, starting position of histogram "Acer West"$_1$. Counter k is advanced to point to next start of the interval within histogram "Acer West"$_1$.
- End position of current interval is found comparing next bin start positions for both histograms and taking the minimum value. Currently it is -2.3, the start of histogram "Acer West"$_1$.
- Now the *join* and the *meet* for current bin are found using Definitions 1 and 2. Probability of join is $(\frac{0.1\times(-2.3+3.9)}{(0.2+3.9)}+0)/2=0.02$. The Cartesian join is [-3.9, -2.3]0.02, and the Cartesian meet is [-3.9, -2.3]0 as histograms have no overlap. Counter for m is advanced. Those probabilities are saved with variables holding cumulative percentages (0.02 and 0 currently for join and meet).
- Next, dissimilarity in current bin is found using size. Size of join is (-2.3+3.9)×0.02=0,032 and size of meet is (-2.3+3.9)×0. Therefore, dissimilarity is 0.032-0=0.032.

- Then, next bin is found that is [-2.3, 0.2]. Probability for join is found by $(\frac{0.1\times(0.2+2.3)}{(0.2+3.9)}+\frac{0.1\times(0.2+2.3)}{(0.6+2.3)})/2=0.074$ and probability for meet is found by $\min\left(\frac{0.1\times(0.2+2.3)}{(0.2+3.9)},\frac{0.1\times(0.2+2.3)}{(0.6+2.3)}\right)=0.061$. Therefore, the Cartesian join is [-2.3, 0.2]0.074, and the Cartesian meet is [-2.3, 0.2]0.061. Cumulative percentages are updated to 0.094 for join and 0.061 for meet. Counter for m is advanced. Dissimilarity for that bin is (0.2+2.3)×0.094-(0.2+2.3)×0.061=0.235-0.1525=0,0825.
- Next bin [0.2, 0.5] is found. The join and the meet for this bin are [0.2,0.6]0.025 and [0.2, 0.6]0.014. Counter for k is advanced. Dissimilarity for current bin is (0.6-0.2)×0.119-(0.6-0.2)×0.075=0.0476-0,03=0.0176
- The same procedure is done over and over again until bin [17.9, 20.6] has been processed. Histogram "Acer West"$_1$ has run out of bins (counter m is advanced further than number of bins in histogram). Then, "leftover" bins from another histogram, this case "Acer East"$_1$, are considered. Only one of his kind of bin remains – [20.6, 23.8]. As there is no overlap, the meet is 0 and join is [20.6, 23.8]0.027. Cumulative percentages are now 1 for join and 0.628 for meet. Dissimilarity for current bin is (23.8-20.6)×1-(23.8-20.6)×0.628=3.2-2.0096=1.1904
- Bin wise dissimilarities are summed up equalling to the result of formula 6.14 $\Phi(\text{"Acer East"}_1, \text{"Acer West"}_1) = 0.032+0.08+0.017+0.091+0.244+0.056+0.626+0.333+0.219+1.138+1.174+0.932+1.19=6.13$
- The length of F_1 is 39.4. Therefore, the normalized dissimilarity is $\Phi_N(\text{"Acer East"}_1, \text{"Acer West"}_1)$=0.156.
- Then, the dissimilarities for other features ($F_2 - F_9$) are found. Table 9.2 summarizes the results.

Table 9.2: Dissimilarities between objects "Acer East" and "Acer West" for features F_1-F_9.

j	$\Phi(\text{"Acer East"}_j, \text{"Acer West"}_j)$	$\lvert F_j \rvert$	$\Phi_N(\text{"Acer East"}_j, \text{"Acer West"}_j)$
1	6.132	39.4	0.156
2	8.299	57.1	0.146
3	6.541	26.7	0.245

4	1809.594	460 0	0.393
5	156.861	666	0.236
6	132.382	452	0.293
7	8.293	57.7	0.144
8	2.33	8.4	0.277
9	0.152	0.92	0.165

These obtained dissimilarities are then summed up 2.055 and divided by the number of features, *nine*. The resultant normalized dissimilarity between objects "Acer East" and "Acer West" becomes 0.228.

After comparing all pairs of objects, two objects with the smallest dissimilarity are merged. In this case, they are objects "Acer East" and "Fraxinus East". Two histograms for every feature of these objects are then combined to cluster A and descriptive object for this cluster A is formed. In both cases, previous clusters contained one element and its descriptive object was the object itself.

We use the following procedure to find descriptive object $\omega_{A_c}$for the cluster by "Acer East" and "Fraxinus East".

- For each feature, we find the descriptive histogram by a similar process used for dissimilarity.
- For feature F_1, the first bin is [-2.3, 0.6] which intercepts with "Acer East" bin [-2.3, 1.4) with probability 0.1 and "Fraxinus East" bin [-2.3, 0.6) with probability 0.1. Using formula 6.19, probability for bin [-2.3, 0.6] is ((0.1×1×(0.6+2.3)/(1.4+2.3))+(0.1×1×(0.6+2.3)/(0.6+2.3)))/2= 0.089.
- The next bin is [0.6, 1.4]. "Acer East" has probability 0.15 associated with this bin [0.6, 3.8), while "Fraxinus East" has probability 0.1 associated with bin [-2.3, 0.6). The probability of bin [0.6, 1.4) for combined descriptive histogram for F_1 becomes the weighted average: ((0.15×1×(1.4-0.6)/(3.8-0.6))+(0.15×1×(1.4-0.6)/(0.6+2.3)))/2=0.03.
- By repeating the same process until all bins are covered, we obtain the descriptive histogram of cluster A for F_1.
- After the repeated use of the above process for the remained features, we obtain the set of descriptive histograms and thus the descriptive object. Table 9.3 summarizes the results for "Acer East" and "Fraxinus East".

Table 9.3: The descriptive object for "Acer East" and "Fraxinus East".

j	Descriptive object for the cluster "Acer East" and "Fraxinus East"
1	{[-2.3,0.6]0.089; [0.6,1.4]0.03; [1.4,3.8]0.118; [3.8,4.3]0.025; [4.3,8.6]0.225; [8.6,9.2]0.028; [9.2,14.1]0.229; [14.1,14.4]0.013; [14.4,17.9]0.144; [17.9,23.2]0.095; [23.2,23.8]0.005}
2	{[-24.6,-23.8]0.006; [-23.8,-18.3]0.091; [-18.3,-18.0]0.006; [-18.0,-13.1]0.136; [-13.1,-12.3]0.024; [-12.3,-6.0]0.22; [-6.0,-5.1]0.03; [-5.1,1.7]0.225; [1.7,2.3]0.018; [2.3,7.5]0.137; [7.5,7.9]0.007; [7.9,18.1]0.094; [18.1,18.9]0.004}
3	{[11.5,13.5]0.02; [13.5,16.6]0.07; [16.6,17.4]0.048; [17.4,18.2]0.075; [18.2,19.0]0.063; [19.0,22.2]0.225; [22.2,25.7]0.247; [25.7,25.8]0.008; [25.8,27.3]0.141; [27.3,27.4]0.008; [27.4,28.8]0.08; [28.8,29.5]0.017}
4	{[270.0,410.0]0.05; [410.0,415.0]0.002; [415.0,655.0]0.113; [655.0,720.0]0.043; [720.0,835.0]0.131; [835.0,910.0]0.09; [910.0,1010.0]0.128; [1010.0,1130.0]0.147; [1130.0,1200.0]0.074; [1200.0,1320.0]0.105; [1320.0,1355.0]0.023; [1355.0,1630.0]0.094}
5	{[6.0,10.0]0.033; [10.0,12.0]0.024; [12.0,21.0]0.11; [21.0,23.0]0.015; [23.0,40.0]0.138; [40.0,55.0]0.12; [55.0,69.0]0.119; [69.0,85.0]0.141; [85.0,96.0]0.076; [96.0,118.0]0.103; [118.0,127.0]0.031; [127.0,166.0]0.091}
6	{[18.0,54.0]0.05; [54.0,56.0]0.008; [56.0,74.0]0.11; [74.0,77.0]0.026; [77.0,89.0]0.15; [89.0,94.0]0.088; [94.0,100.0]0.122; [100.0,108.0]0.148; [108.0,113.0]0.068; [113.0,127.0]0.103; [127.0,135.0]0.032; [135.0,218.0]0.093; [218.0,222.0]0.002}
7	{[-24.6,-23.8]0.006; [-23.8,-18.3]0.091; [-18.3,-18.0]0.006; [-18.0,-13.1]0.136; [-13.1,-12.3]0.024; [-12.3,-6.1]0.22; [-6.1,-5.2]0.03; [-5.2,1.7]0.226; [1.7,2.3]0.018; [2.3,7.5]0.137; [7.5,7.9]0.007; [7.9,18.0]0.094; [18.0,18.8]0.004}
8	{[0.5,0.8]0.021; [0.8,1.2]0.069; [1.2,1.3]0.035; [1.3,1.5]0.1; [1.5,1.6]0.038; [1.6,2.4]0.225; [2.4,2.5]0.024; [2.5,3.5]0.227; [3.5,3.6]0.017; [3.6,4.8]0.144; [4.8,6.7]0.098; [6.7,6.8]0.002}
9	{[0.39,0.6]0.05; [0.6,0.62]0.007; [0.62,0.83]0.107; [0.83,0.89]0.074; [0.89,0.94]0.127; [0.94,0.95]0.052; [0.95,0.97]0.167; [0.97,0.98]0.104; [0.98,0.99]0.138; [0.99,1.0]0.175}

Figure 90.1 (a) shows the result of clustering using the proposed dissimilarity.

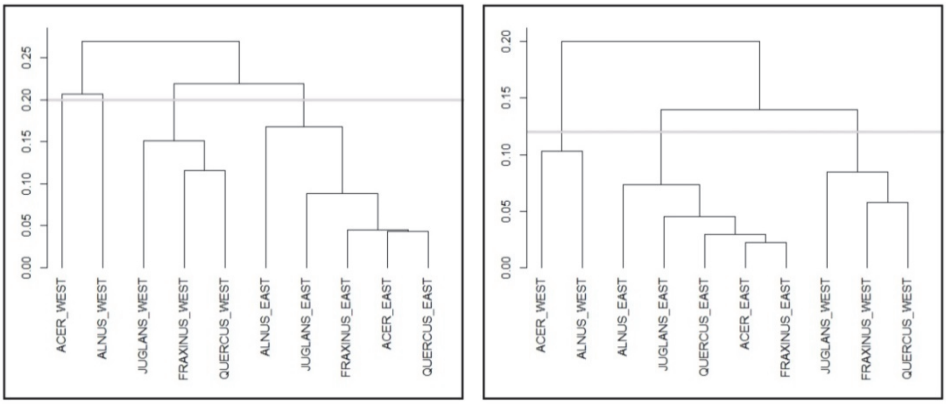
Table 9.4: The cluster compactness for cluster "Acer East" and "Fraxinus East".

j	$\Phi_N(\omega_{A_{c_j}}, \omega_{\text{Acer East}_j})$	$\Phi_N(\omega_{A_{c_j}}, \omega_{\text{Fraxinus East}_j})$	$C(A)$
1	0,016	0,017	0,016
2	0,013	0,013	0,013
3	0,025	0,024	0,025
4	0,021	0,022	0,022
5	0,015	0,015	0,015
6	0,033	0,033	0,033
7	0,013	0,013	0,013
8	0,031	0,032	0,031
9	0,032	0,032	0,032
Sum/nr of features:	0.022	0.022	0,022

Now, we describe the clustering based on the compactness (21):

- First, for each pair of clusters, they are combined into a cluster *A* using the same procedure as described in the linkage method of clustering by the dissimilarity. For each new cluster, we find new descriptive object in such a way as Table 9.4 for "Acer East" and "Fraxinus East".
- Then, we find the distance between each object and the descriptive object in the cluster *A* by using the normalized dissimilarity (22). During the first iteration, there are always two objects in each cluster *A*. Table 4 summarizes the results to find the compactness for the cluster "Acer East" and "Fraxinus East".
- We find the combined clusters pair that has the minimum compactness among all pairs of clusters. We add the combined new cluster to the set *U*, and we remove the two clusters used to form the new cluster from *U*. In the current case, two clusters are the clusters containing "Acer East" and "Fraxinus East", respectively.

Then, we repeat the same procedure until only one cluster is remained in *U* and it contains all objects. Figure 90.1 (b) is the resultant dendrogram.



(a) Dendrogram by the dissimilarity (b) Dendrogram by the compactness

Figure 90.1: Results of clustering of the hardwood data.

Figure 90.1 suggests the existence of three clusters. The difference of metric used during clustering impacts in which order those clusters are combined. In this figure, one of the clusters contains all the species of east coast. Other two clusters contain species from west coast. East coast cluster has strong correlation in three features connected with precipitation. Cluster containing Acer West and Alnus West has similar June temperature and January precipitation. Also, there is strong correlation for moisture index. Another west coast cluster can be characterised by similar January and July temperatures and growing degree-days. West coast clusters have strong difference in January and annual temperatures (with Acer and Alnus being characterized with lower temperature) and difference in growing degree-days and moisture index. Both west coast clusters have similar precipitations all year round.

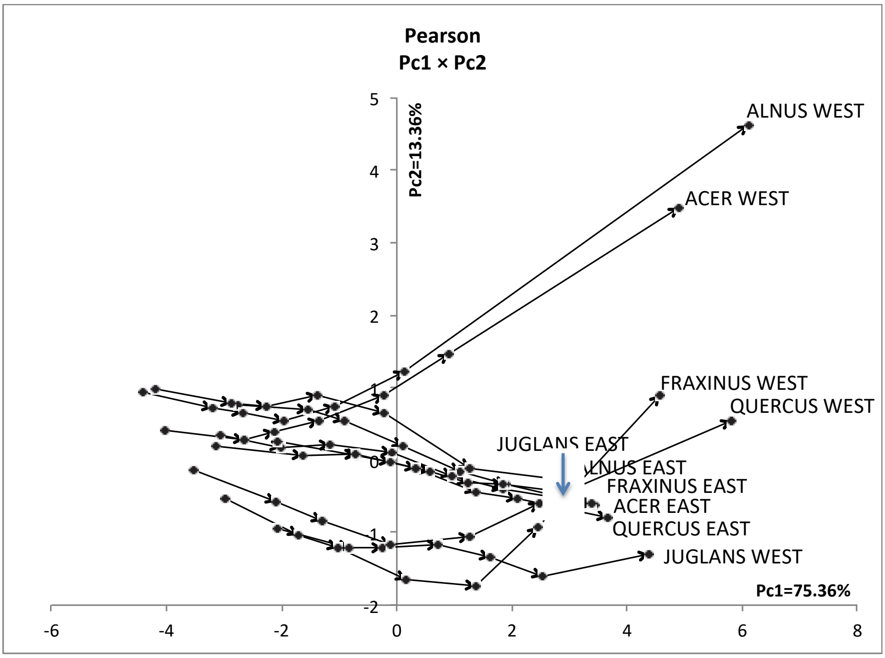


Figure 9.2: Results of PCA on hardwood dataset [189].

Figure 9.2 is the result of the quantile method for PCA for the Hardwood data [189]. Each hard wood is described by a series of *six* arrow lines that connect *eight* dimensional quantile vectors from the smallest to the largest in the first factor plane. This result may support the results of clustering obtained in Figure 90.1.

The US States Temperature Data

This section describes the results of proposed methodology on climate dataset [186]. The dataset contains sequential monthly "time bias corrected" average temperature data for 48 states of USA (Alaska and Hawaii are not represented in the dataset) from 1895 to 2009. The same dataset has also been used in [178] and [181].

The data provided was first transformed into histograms describing average temperature for every state and every month. Then, the proposed dissimilarity and compactness measures were used for clustering. Figure 9.3 and Figure 9.4 show the clusters obtained by cutting the dendrograms, respectively.

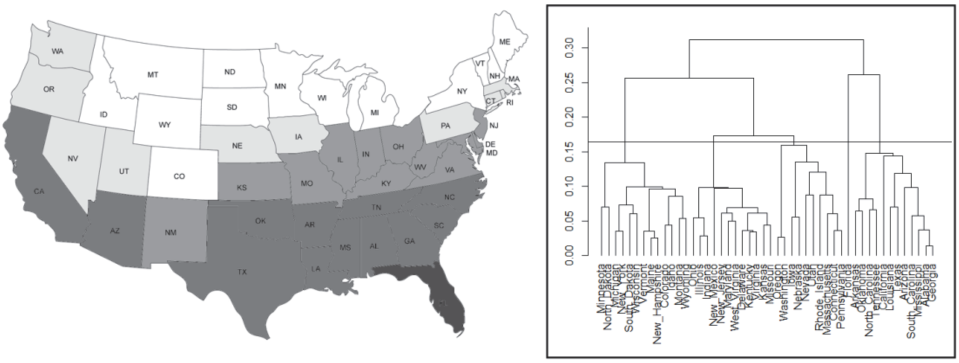


Figure 9.3: Clustering results by the minimal distance between clusters' descriptive objects using proposed dissimilarity measure.

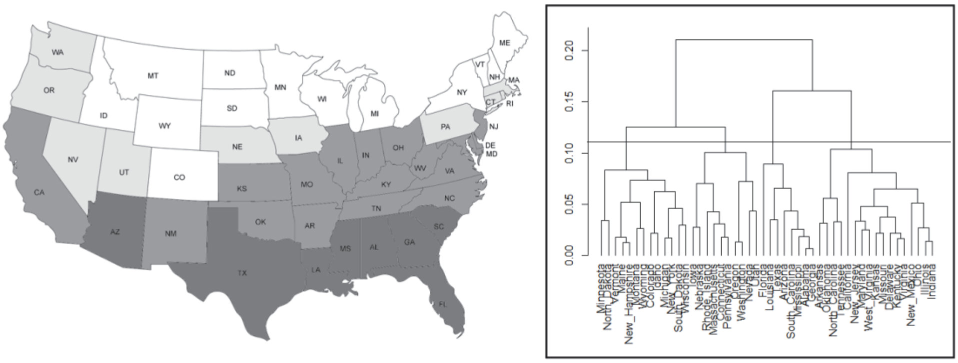


Figure 9.4: Clustering results by the minimal cluster compactness.

As in Figure 9.3 and Figure 9.4, the results are not exactly the same even though they both represent valid clusters combining states with similar weather patterns. The difference comes from the method and its goal. The method to obtain Figure 9.3 uses the distance between the descriptive objects of clusters. Figure 9.4 uses the cluster compactness as the criteria. Therefore, the resultant clusters are "better"/ "more similar" than clusters obtained in Figure 9.4 in the sense that the objects in each cluster are "closer" to the descriptive object of the cluster. The compactness method looks more inside the cluster and considers all the objects in new proposed cluster before deciding which clusters to merge. On the other hand, the method using the dissimilarity only looks to the distance between "centres" of the cluster, concentrating on the outer distance between clusters.

The results of Figure 9.4 are similar to those described in [181]. Instead of four clusters in [181], 5 clusters emerged in our case. If two most northern clusters were combined, the results would be the same.

In both cases, Florida is clustered on its own. Florida has much warmer temperature all year around than other states. Then, other southern states have their own cluster that is characterised by warm summers, especially July and August, and winters likely above 0 degrees in Celsius. Also, both spring and fall are very warm, especially September. Then, mid-range cluster containing mainly

East coast states emerges. This cluster has noticeably colder winters that are likely slightly below 0 degrees. Summers are warm and spring is noticeably colder than autumn months. In [181] the rest of the states are clustered together. In our case, two distinct clusters emerged that have unique, specific patterns. The most northern cluster contains states with really cold winters. This group has March being noticeably colder than other spring or autumn months. July is warmer than rest of the summer months. The second cluster contains states that have winter slightly below 0. Autumn and spring months are warmer than in previous group. March and November weather patterns are similar. Also, both July and August have a warm weather. Therefore, as those states have distinct patterns for weather, it is recommended to cluster them separately as in our case.

Results obtained using compactness as dissimilarity has strong similarities to results obtained in [178]. In [178] five clusters have emerged compared to four in our case. If we had made a cut at five clusters, results in our case and in [178] would be even more similar. Four clusters were chosen as a cutting point due to clearly emerging from dendrogram with greater "difference" between merging as can be seen in Figure 9.3. Also, merging points for clusters 5 and 6 are very close by. Therefore, if we would have wanted to achieve more than four clusters, dendrogram should have been cut after 6 clusters have remained.

When comparing our four emerged clusters with results in [178], two clusters of states with colder weather are identical. Difference comes from states with warmer weather pattern. Results in [178] cluster California and Arizona together with a northern part of southern states. We feel that results in our case reflect more precisely the actual weather and patterns emerging from clusters. Arizona, in our case, is clustered with southern states that include states with really warm weather like Florida. Arizona also has very warm summers and its September is also very hot, something that is common to the most southern cluster that emerged in our case. California, on the other hand, does not have that warm summers and its temperatures in autumn are also noticeably below the ones in Arizona. That's why California suits more into the cluster with other states in similar latitude where we had clustered it.

The cluster with middle states has been divided in two in [178]. We feel that states in that cluster have similar weather patterns and they could be combined into one cluster that is characterized with winters around 0 degrees, three warm summer months with June slightly colder than rest, and similar temperatures for spring and autumn, with September being slightly warmer. The two clusters in [178] vary with states slightly colder and slightly warmer weather clustered separately but the yearly pattern described before remains the same.

ELULOOKIRJELDUS

Isikuandmed

Nimi:	Kadri Umbleja
Sünniaeg ja -koht:	19.03.1987, Tallinn, Eesti
Kodakondsus:	Eesti
E-posti aadress:	kadri.umbleja@ttu.ee

Hariduskäik

Õppeasutus	Lõpetamise aeg	Haridus
Tallinna Tehnikaülikool	2012	Tehnikateaduste magister, *cum laude*
Tallinna Tehnikaülikool	2009	Tehnikateaduste bakalaureus, *cum laude*

Keelteoskus

Keel	Tase
Eesti	Emakeel
Inglise	Kõrgtase
Iiri	Kesktase
Prantsuse	Kesktase

Teenistuskäik

Töötamise aeg	Tööandja nimetus	Ametikoht
2009 – 2012	Tallinna Tehnikaülikool	Insener
2012 – 2013	Tallinna Tehnikaülikool	Nooremteadlane
2013 – ….	Tallinna Tehnikaülikool	Assistent
2010–2012	Tallinna Majanduskool	Tunnitasuline lektor

CURRICULUM VITAE

Personal data

Name: Kadri Umbleja

Date and place of birth: 19.03.1987, Tallinn, Estonia

Citizenship: Estonian

E-mail address. kadri.umbleja@ttu.ee

Education

Educational Institution	Graduation Year	Education
Tallinn University of Technology	2012	MSc, Cum Laude
Tallinn University of Technology	2009	BSc, Cum Laude

Language competence

Language	Level
Estonian	Native
English	Fluent
Irish	Average
French	Average

Professional employment

Period	Organization	Position
2009 – 2012	Tallinn University of Technology	Engineer
2012 – 2013	Tallinn University of Technology	Junior scientist
2013 – ….	Tallinn University of Technology	Teaching assistant
2010–2012	Tallinn School of Economics	Lector

PUBLICATIONS

Publication 1

Reference

K. Umbleja, V. Kukk, and M. Jaanus, "Answer evaluation in competence based learning," in ICEE 2011: An International Conference on Engineering Education, Belfast, UK, 2011.

Abstract

This paper describes competence based approach to measure students' learning results to be used in control of learning process. Most important outcome of studying is acquired knowledge. To get clear and accurate data of student's abilities and gained knowledge, exercises have been divided into elementary tasks that checks small part of knowledge. All tasks are connected to competences and analysis of answer changes the levels of abilities related to corresponding competences. Student provided answer is not only checked against teacher's provided correct answer, but the reason why student has made a mistake is analyzed. By being able to detect places and types of errors in provided answers typos can be eliminated, influence of repeated mistakes can be reduced, appropriate feedback signal can be produced and study materials can be recommended according to mistakes made. Using algorithm to mimic process of finding answers for tasks with multiple correct outcomes can be evaluated. Also, in many cases, part of answer depends on previous actions and mistake in the beginning of the answer can be avoided to be carried over to other parts of mistakes even if those parts were processed correctly. By using memory models and lowering students' ability levels by implementing forgetting model forcing students to retake tasks, learning has been personalized and moved from temporary knowledge toward long term attainments. Competence based approach enables to point student toward the certain tasks he/she needs to retake in order to fix knowledge. The process is repeated until system confirms required level of skills. Students get instant response to experiment and can repeat exercises as many times as they wish.

Answer evaluation in competence based learning

[1]K. Umbleja, [1]V. Kukk, [1]M. Jaanus
Tallinn University of Technology, Tallinn, Estonia, kadri.umbleja@dcc.ttu.ee[1]

Abstract

This paper describes competence based approach to measure students' learning results to be used in control of learning process. Most important outcome of studying is acquired knowledge. To get clear and accurate data of student's abilities and gained knowledge, exercises have been divided into elementary tasks that check~~s~~ small part of knowledge. All tasks are connected to competences and analysis of answer changes the levels of abilities related to corresponding competences. Student provided answer is not only checked against teacher's provided correct answer, but the reason why student has made a mistake is analyzed. By being able to detect places and types of errors in provided answers typos can be eliminated, influence of repeated mistakes can be reduced, appropriate feedback signal can be produced and study materials can be recommended according to mistakes made. Using algorithm to mimic process of finding answers for tasks with multiple correct outcomes can be evaluated. Also, in many cases, part of answer depends on previous actions and mistake in the beginning of the answer can be avoided to be carried over to other parts of mistakes even if those parts were processed correctly. By using memory models and lowering students' ability levels by implementing forgetting model forcing students to retake tasks, learning has been personalized and moved from temporary knowledge toward long term attainments. Competence based approach enables to point student toward the certain tasks he/she needs to retake in order to fix knowledge. The process is repeated until system confirms required level of skills. Students get instant response to experiment and can repeat exercises as many times as they wish.

1. Introduction

To measure studying outcomes and have a full and accurate overview of student knowledge classical "topic -based" learning is not sufficient. Competence based learning with grading skills separately can give much clearer picture. With new kind of approach to grading tasks, different way to analyze and evaluate answers is also required.

1.1 Background: learning environment ISC

ISC is e-learning environment used in Department of Computer Control, Tallinn University of Technology from the end of 1990-s [1].

This system is used to teach many different courses with some courses having overlapping material. Every course has poll of tasks and lab experiments. They were divided into topics (etc "Two-ports" in "Circuits, Systems, Signals" course) and every topic had large number of different variation of tasks connected to it. Tasks and lab experiments are designed to be small concentrating only in small portion of theory.

Every topic has state what is like grade showing the progress made in that field. Instead of using commonly practiced system with 5-6 grades, 128 different ability levels are used in order to avoid discretization effects and have more effective feedback control [1]. When student decides to solve some topic, a task or lab experiment is chosen for student according to topic, level and frequency of task taken. The goal is to use all possible tasks in topic's pool, try to avoid student getting the same version of certain task more than once and get task closest in its level to student's state on topic.

System is fully web based, giving students the possibility to solve tasks when and where they wish not being bound to campus. Also, the system supports HomeLabKits – portable lab equipment what gives students the possibility to complete entire course in web on time and place of their own choosing [2,3].

ISC takes "authentic" student input meaning that even if there are some multiple choice items, majority of tasks need student to fill in the answer [4]. It also has automatic evaluation in tasks and lab exercises therefore; no teacher interference is needed for solving exercises. Students get instant response thanks to algorithms used for processing answers. Those algorithms give feedback and calculate results.

2. Competence-driven learning

Most important outcome of learning is acquired knowledge what is hard to measure even if there exists many proposed metrics, they may not give proper picture [5]. Problem with one number representing the knowledge on some wider topic is that it is not measurement of elementary skill or competence rather than summation of elementary competences (summative grading). In order to give proper picture of student's abilities every task and lab evaluation should not measure the summed or averaged levels of skills but be able to see different competences within tasks. That kind of approach can result in personalization, better task and lab evaluation, clearer picture of student abilities and it can give to teacher a view of current students' abilities and learned skills. That would give teacher a better understanding of class level, what they already know and where gaps in their knowledge are. That would save time, avoid repeating material student already know (but what sadly happens quite often) and gives teacher better idea how to plan course material between lectures. To achieve more detail and clearer picture of student abilities we replaced classical topic based learning approach by skill based competence learning.

Competence learning is knowledge base methodology, what concentrates on what person can actually do as a result of learning [6,7,8]. Instead of evaluating one result for a task, every exercise is analyzed to extract smaller parts of skills what the task uses checks or requires [9]. For example, task in which the following is to be evaluated $x=40/2*5+1$ not only checks student's ability to solve mathematical equations but makes sure student can multiply, divide and add. Analysis performed before changing over to competence learning showed that in average for every task about 4 competences could be extracted [1].

Answer evaluation in competence learning differs from answer evaluation in topic based learning. As every task is related to multiple competences, answer evaluation will not give one result for whole task but results for every competence that are present in completed task. All competences and results are independent and therefore some competence levels can go up and other go down at the same time. That solves previous problem with topic based learning and answer evaluation where summative grading is difficult and can be misleading [1].

Competence learning and evaluating answers within that kind of method goes into details. Instead of huge experiments taking hours, exercises have been divided into elementary tasks that check small part of theory enabling competences to be acquired easily and reduces the amount of information acquired at the same time allowing significant increase in the total number of individual tasks [1].

With competence learning we will no longer have topic field where student chooses topic and task is chosen for him/her. Instead now student chooses a competence wishing to improve. In ISC, three levels of granularity are implemented where in 1st level every competence can be seen and accessed. On 2nd level similar competences have been assembled and 3rd level is most general but less than a course. Also, in case when students do not wish to make decision themselves, their current level of knowledge and previously solved tasks are used to recommend next competence and task with it to solve. Competence learning gives us ability to personalize learning as student has to repeat and retake tasks with competences and parts of theory he or she has not yet acquired with sufficient level [9]. With forgetting model personalization can be taken even further.

3. Processing answers

Answer evaluation the most important part of competence learning providing valuable information to both student and teacher [4]. The simplest answer evaluation is comparing student's submitted answer with teacher's provided correct answer. That gives two possibilities: the answers are the same and therefore it is presumed that student has submitted correct answer and knows how to solve the problem. The alternative is that the answers are not the same and student did not solve the problem correctly. In most cases that kind of black-and-white approach of answer analyze is not enough as tasks may contain many different competences, consist of many intermediate steps and intermediate sub-results influence final answer. When analyzing wrong answers from log files one can first determine possible errors and from that define competences that may consist in correct answer.

Perfect algorithm for answer analysis would be able to find places of mistakes, understand reasons why mistakes were made and find associated competences. It will also be able to alter given correct answer according to made mistake. Therefore one mistake made would not make rest of the answer fully wrong even if it depends on earlier calculations. There are types of tasks where more than one way to solve exists. Those tasks cannot be checked with black-and-white approach as student has more than one possibility to submit correct answer. One alternative is to have more than one correct answer provided by teacher and student provided answer is compared with all of them. That becomes inefficient if the number of possible answer increases. Therefore algorithm that mimic the answering process following the path student has used is preferred as it should be able to adapt to any path student chooses to give the correct answer.

We are particularly interested in student's incorrect answers as there is much to learn from them – what went wrong, why went wrong, where the answer went wrong, what competence should be more focused on teaching etc.

3.1 Case study

In the course(s) "Operating Systems" there is a set of tasks where students have to mimic process scheduling in CPU. They are given description of three processes with time spent in CPU and resources declared. Correspondingly to scheduling strategy given, student has to fill table and show in which moment of time where certain process is.

Process scheduling

Given
Proc 1: R1-2:P-3:R2-1:P-2:W1-2:P-2
Proc 2: P-3:R1-2:P-3:W1-1:R2-2:P-2
Proc 3: R1-2:W1-3:P-3:R1-1:P-2:R2-2
Strategy: SJN
Q=

Task
Fill the table by arranging process numbers 1,2 and 3 in appropriate place. In every row (moment of time) every process can appear only once until its work has not been finished.

Time	Ready	Processor	Resource1			Resource2		
			wait	R	W	wait	R	W
1		2		31				
2		2		31				
3	1	2			3			
4		1	2		3			
5		1	2		3			
6	3	1		2				
7		3		2			1	
8	21	3						

answer

Figure 1: Process scheduling task with filled in answer

Originally the task had control algorithm what compared input string with teacher's provided correct answer string until the first difference. Then the percentage of the answer what matched was used to calculate the result. With that kind of analysis a mistake meant end of answer evaluation. It did not matter how serious the mistake was. Even typos or carelessness mistakes like forgetting to assign an action to process in every moment of time were as crucial as misunderstanding the task or making mistakes in interpreting scheduling strategy.

That control worked efficiently if there were no mistakes or mistakes were made in the later part of the answer. Sadly, that kind of approach did not allow any proper feedback or identification of mistake made. In order to give students proper comments to their answer, understand mistakes and suggest materials according to gaps in student's knowledge, a different approach was needed. The first phase of new algorithm development contained analyzing previous answers as the only source showing where and why mistakes are made. It was learned that majority of mistakes where done with 1) process priorities, 2) resource management (many students failed to understand that one file can be read by many processes but only one process can write a file and writing is preclusive for reading and vice versa), and surprisingly 3) many typos.

The first priority was to reduce the impact of typos. All processes can be in CPU, in waiting list or performing resource action in any given time. Many students had assigned process in one moment of time for more than one action. Therefore a pre-submitting control to answer input table was added what raised an error message when a process was added more than once and removed lastly added process. That lowered typos and carelessness mistakes almost by half.

The main part of algorithm is analysis of mistakes. First, the moment where student's answer and provided correct answer start to differ. Then the line where the mistake was made is thoroughly examined. Mistakes with resource are easily determined. Types of mistakes are numbered and number of occurrences is stored in array. Every time when a certain mistake is made, that number in array is increased. So, every moment of answer evaluation, the number of different mistakes and total mistakes are known. With every type of mistake a proper comment is added to global comment variable.

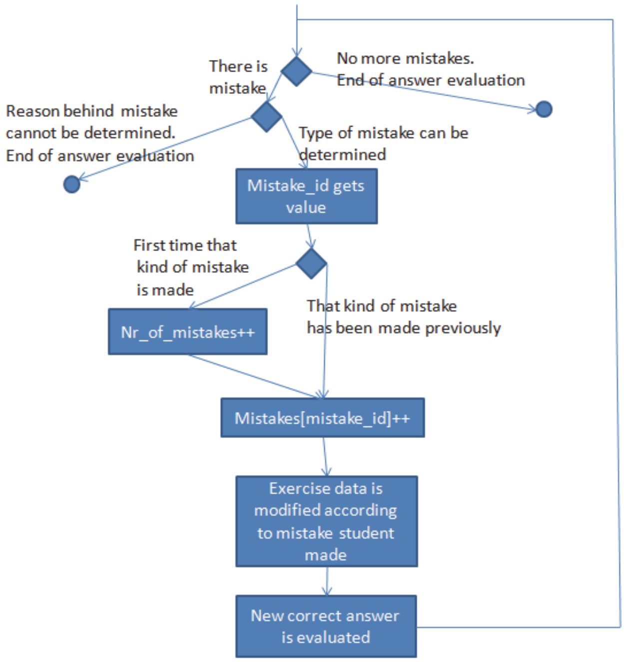


Figure 2: Answer evaluation process

For example, if it is detected that when t=4 there are processes reading and writing the same resource, it is determined that competence “Process scheduling: Can only read or write” is not followed. For example that is error type nr 3. Array for error counting is increased in proper place and comment “Line 4: Currently processes are reading and writing to the same resource at the same time. “Processes can only read simultaneously or write exclusively” is added to global message what will be printed to the screen after the answer evaluation has ended.

When mistake is determined it is recorded and that line is, in answer evaluation purpose, considered correct. Therefore the “global” correct or teacher provided correct answer cannot be used anymore and new correct answer has to be evaluated. It is sure, that the parts of correct and student provided answer to the start of the line where mistake was made are correct. The line where mistake was made is considered “correct” in student provided answer. The rest of the correct answer has to be evaluated for further analysis.

To obtain rest of new correct answer, the algorithm that is used by student for answering has to be implemented. With process scheduling that is easy as students have to follow certain rules and scheduling algorithm. That process just has to be coded and the description of processes on the time moment where mistake was made has to be known. Therefore, at every time step, the description of processes has to be altered according to the steps made. With known description of processes, number of moments known to the end of the task and algorithm to mimic answering process, new correct answer can be calculated.

With new correct answer that takes into account the data student had after the mistake was made, the process of analyzing “new correct” answer with student provided answer continues until the end of the answer is reached. When new mistake is found, new comment is added to global comment variable, amount of mistakes is increased and correct answer re-evaluated.

When the end of answer evaluation is reached, global comment is printed and result is calculated. Final result is not one number for the task as every task is related to many competences. In that example, up to 31 different types of mistakes can be determined and 12 different competences have been evaluated. Many different types of mistakes can affect one competence or one type of mistake can affect many competences. For example, concerning reading and writing at same time, affects competence “Process scheduling: Can only read or write” but it also has smaller effect on competence “Process scheduling: Resources”. Also, every mistake and total number of different mistakes is used to calculate outcome for competence “Processes” and the competence corresponding to the algorithm used for process scheduling.

Changes in every competence levels (abilities) are calculated separately using common algorithm and therefore some competences can go up and others can go down. That gives flexibility to measure different aspects of tasks and to consider the mistakes made. It is possible that student can understand the strategy used for process scheduling but does not understand that this algorithm only applies to CPU and not to the resources. So as the result ‘scheduling strategy’ knowledge competence will go up but understanding resource management will go down. That kind of new algorithm gives us possibility to find and understand the mistakes students have made, compile proper feedback signals, give proper comments, and according to mistakes (or competences) direct student to associated learning materials. Answer evaluation is automatic and enabling real-time feedback. (It is possible to engage a teacher as part of evaluation however this may cause extra delays – unwanted in feedback control). Students can solve their tasks any time and place, get real-time comments about their answers, and adopt their knowledge of the problem according to feedback. They are not limited how many times they can try to solve the task and extra tries will not raise teacher’s workload.

4. Forgetting model

Human forgetting has been studied carefully over long time and H. Ebbinghaus was first to show with his experiments that retention is very fast in the beginning and quite slow later and repeating is important to memorization [10]. When student solves some task and gets to the higher competence level, the new knowledge goes to short term memory. To achieve long term memory of certain knowledge, student has to repeat the tasks connected to that knowledge competence. As the number of times one type of task is taken rises, it is important to have many variations of the same task as students strongly oppose getting the same task they have already solved.

As answer evaluation is done automatically there is no new stress for teacher with repeated taking of task and memory model can be used for further personalization. The purpose of memory model is to lower the overall result of certain competence over time in order to mimic forgetting. The level of forgetting depends on student previous behavior and the correctness of student’s answers. When student result is lowered by forgetting model over time after first time taking task with previously unsolved competence and student retakes task with the competence and answers correctly, it is assumed he has lower threshold of forgetting than when making mistake in repeated attempt.

5. Conclusions

When forgetting model attached to old “topic based” learning was introduced in 2007 the amount of tasks taken increased 50%. Changing over to competence based learning with forgetting model in 2010, the number of tasks taken by students increased remarkably. Before introducing competence based learning, in 2009 and 2010 spring, students solved 140 tasks on average. After competence based learning was introduced in 2010 fall and 2011 spring average number of tasks solved by student raised to 507. At the same time tasks were divided into smaller units and whole number of tasks available increased too. It was also visible that students' later attempts were usually much better than their first attempts. In student's first five tries with new competence the average result was 0.16 and on their last five tries the average result was 0.58 (with result in [-1.0, 1.0]). That proves that forcing students to retake exercises their knowledge gets more long-term one and they get better understanding of material.

With improved answer evaluation and generated correct answers by algorithm, rather than just checking answer against correct one, in 71% of cases student got better result value due removing typos and lowering influence of repetitive mistakes. In 27% the result remained the same and in only 2% the new algorithm graded answer with lower result value. This proved that generated correct answers and evaluating them based on competences is more efficient in reflecting student's real progress. All that has proved that with competence learning, forgetting model, and complex answer evaluation more information for both student and teacher can be collected, better feedback can be obtained, studying can be personalized and proper picture of student abilities can be drawn.

References

1. V. Kukk, M. Jaanus, K. Umbleja and O. Shvets "Developing integrated learning environment – analysis, " *Proceedings of International Conference on Engineering Education ICEE-2010*, Gliwice, Poland, 2010.

2. M. Jaanus, V. Kukk and K. Umbleja, "Integrating Labs into Learning Environment, " *Electronics and Electrical Engineering,* Vol. 6, 2010, pp 27 - 30.

3. O. Švets, V. Kukk, M. Jaanus and K. Umbleja, "Remote laboratories and methods of their aplication in study procsess". *Journal of International Scientific Publications: Materials,Methods & Technologies*, Vol. 4, No. 2, 2010.

4. K. Scalise,D.J. Bernbaum, M. Timms, S.V. Harrell, K. Burmester, C.A. Kennedy and M. Wilson, "Assessment for e-Learning: Case studies of an emerging field", *Proceedings of International Objective Measurement Workshop*, Berkeley, CA , 2006.

5. J. Liebowitz, and C.Y. Suen, "Developing knowledge managment metrics for measuing intellectual capital", *Journal of Intellectual Capital*, Vol. 1, No. 1, 2000, pp. 54-67.

6. H. W. Van der Broeck and H. C. Skudelny, "Analysis and realization a pulse width modulator based on a voltage space vector", *IEEE Trans. Industry Applications*, Vol. IA-24, No. 1, 1988, pp. 142-150.

7. D. A. Grant, M. Stevens and J. A. Houldsworth, "The effect of word length on the harmonic content of microprocessor-based waveform generators". *IEEE Trans. Industry Application,* Vol. IA-21, No. 1, 1985, pp. 218-225.

8. P. Horowitz and W. Hill *The art of Electronics 2. edition*, Cambridge University Press, Cambridge, 1998.

9. C. Hockemeyer, O. Conlan, V. Wade and D. Albert, "Applying Competence Prerequisite Structures for eLearning and Skill Management", *Journal of Universal Computer Science*, Vol. 9, No.12, 2003, pp. 1428-1436.

10. C. Ostyn, Competency Data for Training Automation. [Online] [Cited: 04 12, 2011.] http://www.ostyn.com/standardswork/competency/CompetencyDataForTrainingAutomation.htm.

Publication 2

Reference

K. Umbleja, V. Kukk, M. Jaanus, A. Udal, and B. Gordon, "Analyzes of Competence Based Approach to Learning " in ICEE-2012, Turku, Finland, 2012, pp. 418–424.

Abstract

Since 2010, when the competence based approach was introduced in ISC e-learning environment in Tallinn University of Technology in Department of Computer Control, a lot of information about the students' behaviours is being collected and can be analysed to restructure the learning process to fit with the rapidly changing learner.

In the last two years almost 300 different competences have been identified which can be individually marked. Also, algorithms used to evaluate student answers were analysed to see if there is any connection between the structure of the algorithm and the students' result. Also, changes in student results, learning behaviour and the pattern of using different kinds of help-materials were analysed, and some surprising results were found.

Analyzes of Competence Based Approach to Learning

Kadri Umbleja[1], Vello Kukk[2], Martin Jaanus[3], Andres Udal[4], Boris Gordon[5]

[1] *Tallinn University of Technology, Tallinn, Estonia, kadri.umbleja@dcc.ttu.ee*
[2] *Tallinn University of Technology, Tallinn, Estonia, vello.kukk@dcc.ttu.ee*
[3] *Tallinn University of Technology, Tallinn, Estonia, martinj@ttu.ee*
[4] *Tallinn University of Technology, Tallinn, Estonia, audal@va.ttu.ee*
[5] *Tallinn University of Technology, Tallinn, Estonia, boris.gordon@dcc.ttu.ee*

Abstract

Since 2010, when the competence based approach was introduced in ISC e-learning environment in Tallinn University of Technology in Department of Computer Control, a lot of information about the students' behaviours is being collected and can be analysed to restructure the learning process to fit with the rapidly changing learner.

In the last two years almost 300 different competences have been identified which can be individually marked. Also, algorithms used to evaluate student answers were analysed to see if there is any connection between the structure of the algorithm and the students' result. Also, changes in student results, learning behaviour and the pattern of using different kinds of help-materials were analysed, and some surprising results were found.

Keywords: *competence learning, automatic evaluation, answer evaluation, student behavior.*

1. Introduction

Improving the quality of learning can only be based on improving all components in a closed-loop feedback system, particularly improving the system of measuring a student's abilities and behavioural characteristics. Classical "topic-based" learning is insufficient in obtaining details of a student's knowledge. A novel "competence-based" approach was offered by which the studying outcomes can be measured as skills are processed separately to give a much clearer picture. After almost two years of using that novel approach to learning, its impact and the way it changed student learning behaviours is analysed.

2. Background:

2.1. Learning Environment ISC

ISC is an e-learning environment used in Department of Computer Control, Tallinn University of Technology (TUT) from the end of 1990s [1].

The system is used in the teaching of many different courses in computer science field, not only in TUT, with some courses having overlapping material. The system started with the classical “topic-based” approach, where the set of tasks were divided into discrete topics [2]. For every topic in a course, student’s state of knowledge and level of efficiency was measured. The system is fully web-based, giving students the ability to solve tasks when and where they wish, by not being bound to campus. Also, the system supports HomeLabKits – portable lab equipment that gives students the possibility to complete the entire course via the Internet in the time and place of their own choosing [3] [4]. ISC takes the “authentic” student input. It uses automatic evaluation for the tasks and lab exercises; therefore, no teacher interference is needed to solve exercises. Students get instant responses thanks to algorithms used to process answers. These algorithms give comments and calculate results.

In the 2010 fall semester, competence based learning was introduced into the ISC system. Classical topic-based learning was replaced by skill-based competence learning to achieve a clearer and more detailed view of student abilities [2]. Since that time a lot of information has been gathered and conclusions can now be drawn.

2.2. Competence-Based Learning

Competence-based learning is a knowledge based methodology which concentrates on/measures what a person can actually do as a result of learning [6]. The main issue with measuring knowledge is that when using only one grade to represent the knowledge of some wider topic, it does not measure elementary skill or competence but instead the summation of different elementary competences (i.e. summative grading). For example, if grading the mathematical equation x=(2*5+4)/2 in the classical “topic-based” approach, we check if 'x' is correct and discard the intermediate states/skills as opposed to analysing student ability to add, multiply and divide in competence based approach. Instead of evaluating the one result for each task (‘x’), every exercise is analysed to extract smaller parts of skills (e.g. multiplication, adding) that the task uses, checks or requires [9].

Therefore, if we wish to give a proper representation of students’ abilities, instead of summating or averaging the grading of skill levels, we should look into each and every competence presented in a task and process them separately.

3. Analysis

3.1 Use of competences

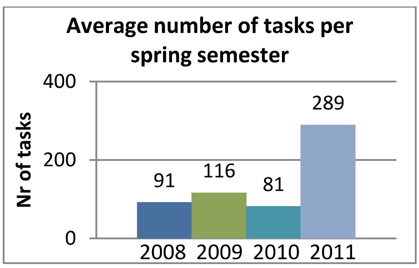
As of April 2012, there are almost 28,000 tasks with almost 300 competences connected to them in ISC system. On average, every competence has 94 tasks in its pool where the task will be chosen when that competence is requested to be activated in an exercise. On average every task has 2.5 input competences connected to it, meaning those competences can trigger a student to get that exercise. In reality, tasks have more competences they can affect – they are called non-input competences meaning that they do not trigger that exercise to be solved but they may be affected under certain circumstances. The number of competences related to a task varies from 1 to 12. There are almost one third of tasks representing only one competence. This shows that merely processing the answers alone is not enough: all tasks may have different reasons for getting an incorrect result and further development based on analysis of the results obtained can improve processing and increase the number of competences.

On average, student has been graded in 73.8 competences. In 70% of competences, the students' averages were above 77, which is considered the borderline of having acquired competence. Also, all the students' average results for all the competences are 82 which is close to expected -3dB level 90.

3.2 Results

The positive effect of competence based learning is that the student results improved remarkably with repetition. The margin between totally wrong and correct over all competences in the system fell from 17.69% to 8.34%. Wrong answers also fell from 29.15% to 23.74%, meaning that totally correct answers rose significantly from 53.16% to 67.93%. Also, when looking at competences and the students first five attempts together, the average result was 0.52. Their last five attempts with the competence gave an average result of 0.66.

One of the goals with competence based learning was to make students repeat the same competence over and over again, to be sure the skill was stored in their long term memory. Figure 1 shows clearly that this goal was achieved. Students tend to spend on average 83 seconds on each exercise (almost a minute and half). In 66% of cases, they spent less than minute, in 46% less than 30 seconds, and in 18% they only had the task opened for 10 seconds before they hit the submit button.



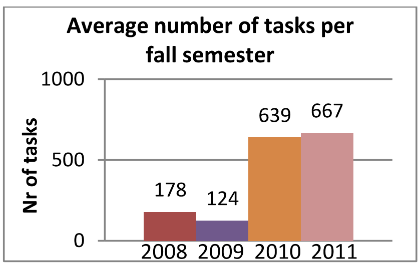


Figure 1: Average number of tasks per semester

3.3. Algorithms

Automatic evaluation is used to check student results. There are more than 1100 algorithms used in the competence based approach. When reducing algorithms into simple programming language elements (if statement, while loop, statement sequence etc.) without semantics (ignoring variable names etc.), it comes out that there are actually, in form, only 155 distinct structural types.

There were 82 simplified forms that were used by only one group of exercises. That leaves the other 73 structures (which is less than half of all identified structures) to be used in 93% of cases. Leaving out the structures used only once, on average the structure was used in 14 classes.

When comparing the structure and the student results, it is clearly visible that the more complex and specific algorithms correlate with a higher average result for the class they associate with. When looking at the top 10 algorithm structures with highest average mark, all of them, except one, associate with only 1 or 2 groups of exercises.

3.4 Use of help materials

Table 1: Use of help materials (%) by type and semester

	2008S	2008F	2009S	2009F	2010S	2010F	2011S	2011F
kup	4	10	13	12	10	19	16	22
pdf	41	30	27	41	44	36	22	37
pps	11	11	10	7	8	7	4	7
interactive slides	29	36	32	27	29	26	43	12
video	2	8	13	9	4	7	2	3
wikipedia	13	5	5	5	4	4	12	19

In the ISC system, a student has the ability to access a variety of help materials. Throughout the years, some changes in the students' behaviour in using the material are noticeable. Students tend to use less and less of the recorded lectures, even though there were no formal lectures held. This shows that students don't find lectures the best form of obtaining information. They tend to prefer getting all the required information in a small concentrated portion, and that explains why Wikipedia links are becoming more popular (table 1).

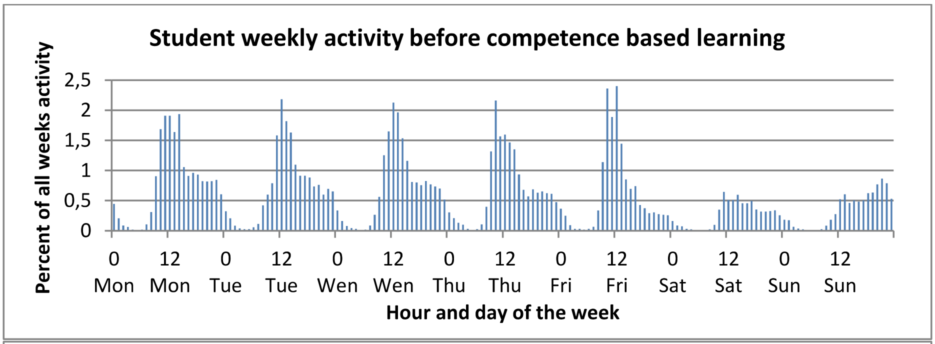
Table 2: Student results with and without hint

Hint nr	Before competence approach		After competence base approach without hint		Competence based approach with hints	
	Normalized avg results	% of correctly answers	Normalized avg result	% of correctly answers	Normalized avg results	% of correct answers
1	0,085	0,45	0,592	25,43	0,784	76,37
2	0,071	1,04	0,577	31,05	0,631	58,47
3	0,149	5,82	0,623	39,05	0,739	73,87
4	0,093	2,87	0,585	35,47	0,672	64,64
5	0,018	0,70	0,530	41,88	0,587	57,43
6	0,156	6,76	0,624	30,11	0,846	81,30
7	0,057	0,47	0,570	32,43	0,689	68,89
8	0,052	0,26	0,569	32,31	0,651	65,03
9	0,038	0,35	0,553	32,60	0,610	60,97
10	0,156	7,45	0,604	18,91	0,732	72,80

For years there has been growing drop in the average number of materials accessed. As it became clear that students don't really go and look up materials until they are in serious trouble, hints were introduced to the system. They usually contain basic information that forms a foundation for solving the task. They were only shown until the student achieved what is considered a borderline knowledge. As an experiment, 14 different hints were introduced, and the students' performance with the tasks before the competence based approach, and after the competence based with and without hints, was analysed.

It was thought that adding hints would improve the results but it was never expected that the improvement is so great (table 2). This indicates to us, that if the students have the materials they will use them, but that they are too lazy to go and look for them themselves. When they are offered, students make good use of them.

3.5. Students' study behaviour



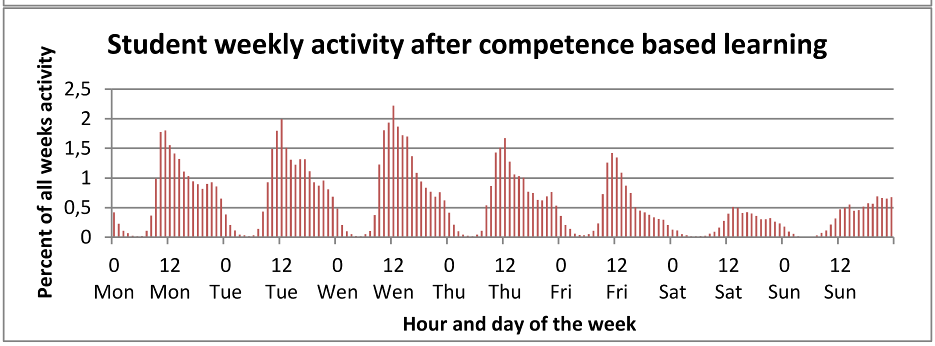


Figure 2: Student weekly activity

With competence based learning, the student's daily activities remained the same. When comparing the students' weekly activity (figure 2), there are more visible changes. Before competence based learning, all the work days seemed to have the same kind of activity. After changes, the start of the week seems to be busier than Thursday and Friday. Friday in particular showed major changes. Also, before competence based learning there seemed to be more peak times correlating with the offered lab timeslot, while. After the daily activity seems to be more divided, and the drop of activity after usual lab hours is more gradual. It can also be seen that student activities usually go on until 4 am, with the exception of Friday and Saturday. Sunday seems to have lost little of its popularity, but is still remarkable.

When comparing the average results students achieved during certain times of the day (figure 3), it is clear that with competence based learning, the average results rose. The average results seem to be steadier than before, and during lab hours the results look to be slightly better which contradicts the drop in the previous case. Therefore, it is probable that changes in lab supervision and help are the cause of the difference.

It was expected that the results during the night would not have been as good as those during the day, as the students would have been tired, and probably in a hurry when doing exercises in the middle of the night. However, the results show no clear signs of this. The results around midnight are actually better than those during the lab hours and earlier in the evening, reaching the highest peak of the day at 2 am. After this time the results reflect more of what would have been expected, as there is drop in the average results.

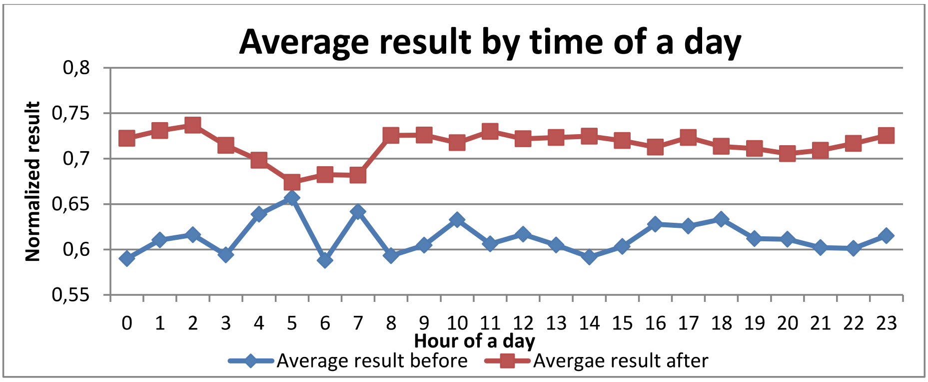


Figure 3: Average result by time of a day

Different patterns can be seen when comparing the students' behaviour and the preferences of working either in lab or on their own outside university (figure 4). More and more students tend to work outside university and complete all or the majority of the course work outside the lab and without supervised help. This shows that we have managed to give students enough ways to get help: having materials and tips next to exercises, providing options to send questions to the teacher after the result is shown, or by mail inside the system.

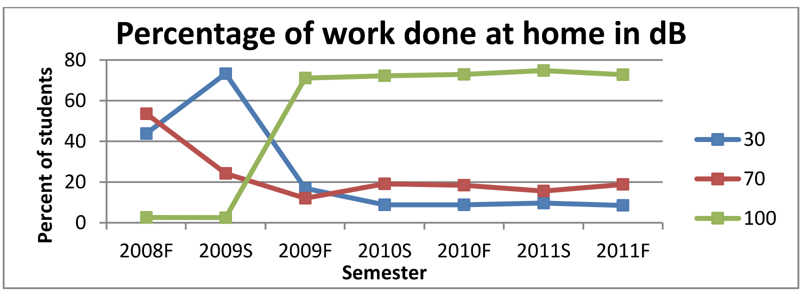


Figure 4: Student working behaviours

3.6 Coverage

Converting old tasks into a form suitable for competence analysis exhibited several problems. First, it appeared that some competences are covered very unevenly. There are several important items of knowledge that are not represented at all and many are underrepresented, i.e. there too few tasks related to them. Numerically, distribution of the number of tasks vs. the number of competences is exponential. One third of competences are represented by 8 or less tasks, and half of the competences are represented by up to 30 tasks. These are very low numbers and leads to repetition of tasks, which not only causes protests from students, but also lowers the quality of the control.

Another conclusion is the following. Students are frequently directed to very simple tasks that are connected to simple but basic concepts, and it appears that a student cannot solve these problems. This may happen later when student has worked on many complex tasks, because the system is able to detect weak points in student's abilities. This reflects the well-known fact that simple and basic concepts can be difficult even if a student can solve (somehow) more complicated tasks. This was evident from the beginning of the competence-based process. The conclusion is that many simple tasks related to basic concepts should be created, inserted, and given somehow a higher priority.

4. Conclusions

1. The competence-based learning environment allows and demands much more detailed consideration of the knowledge and skills that a student can obtain. Introduction of this approach has shown very quickly that there are two weaknesses in the traditional problem sets:
    a. Competences are very poorly covered (few tasks for many competences).
    b. There is a lack of low level tasks that would help to teach the basics of a field, and in fact, testing with complex and complicated tasks masks poor understanding of fundamentals.
2. It appeared that it is possible to extract a lot of information even from very simple tasks, which can be used for a better control of the learning process. However, for this improvement a lot of practical experience (log files) is needed to discover a learner's way of thinking
3. Competence based control produces much higher results partly because of connections between competences through tasks, and partly because a student can never decide not to touch a competence again.
4. Students' workload becomes more even both over the day and over the semester.
5. Using hints in the form of small bunches of supporting information together with problems (i.e. on the same page) helps the student more than links to material consisting of several pages.
6. Competence-based control enables much better formation of control signals (i.e. task selection) because of a more precise adaptation to the student's abilities.

Finally: transferring to competence-based models and control has benefits; however, it has also shown how much work is needed to get the most from this approach.

Acknowledgment

This work has been partly supported by the Estonian Science Foundation Project 9463.

References

[1] V. Kukk, M. Jaanus, K. Umbleja and O. Shvets "Developing integrated learning environment – analysis, " *Proceedings of International Conference on Engineering Education,* ICEE-2010, Paper No.166 ,Gliwice, Poland, 2010.

[2] K. Umbleja, V. Kukk, M. Jaanus "Processing answers in competence-based learning" *Proceedings of International Conference on Engineering Education,* ICEE-2011, Paper No. 108, Belfast, Northern Ireland, 2011.

[3] M. Jaanus, V. Kukk and K. Umbleja, "Integrating Labs into Learning Environment, " Electronics and Electrical Engineering, Vol. 6, 2010, pp 27 - 30.

[4] O. Švets, V. Kukk, M. Jaanus and K. Umbleja, "Remote laboratories and methods of their application in study process". Journal of International Scientific Publications: Materials, Methods & Technologies, Vol. 4, No. 2, 2010.

[5] K. Scalise,D.J. Bernbaum, M. Timms, S.V. Harrell, K. Burmester, C.A. Kennedy and M. Wilson, "Assessment for e-Learning: Case studies of an emerging field", *Proceedings of International Objective Measurement Workshop*, Berkeley, CA , 2006.

[6] M. Jaanus, *The Interactive Learning Environment for Mobile Laboratories*, Tallinna Tehnikaülikooli Kirjastus, Tallinn, 2011.

Publication 3

Reference

K. Umbleja, V. Kukk, and M. Jaanus, "Competence-based approach to learning," in Global Engineering Education Conference (EDUCON), 2013 IEEE, 2013, pp. 552-559.

Abstract

This paper describes competence based approach to complete learning process. The learning process is fully web-based and can be completed without attending campus. It consists of small exercises and lab experiments. Automatic evaluation by stimulating student answering process is explained. Labs are supported by HomeLabKit, small box that contains everything needed to perform lab tasks and can be lent from university. After two years of using competence-based learning in real learning, a lot of information has been collected that can now be analyzed.

Competence-Based Approach to Learning

Kadri Umbleja, Vello Kukk, Martin Jaanus
Department of Computer Control
Tallinn University of Technology
Tallinn, Estonia
kadri.umbleja@ttu.ee

Abstract— **This paper describes competence based approach to complete learning process. The learning process is fully web-based and can be completed without attending campus. It consists of small exercises and lab experiments. Automatic evaluation by stimulating student answering process is explained. Labs are supported by HomeLabKit, small box that contains everything needed to perform lab tasks and can be lent from university. After two years of using competence-based learning in real learning, a lot of information has been collected that can now be analyzed.**

Keywords-component; distance learning; competences; automatic evaluation; answer evaluation; student behavior; remote laboratories; forgetting.

I. BACKGROUND

A. *ISC-learning environment*

ISC is an e-learning environment used in the Department of Computer Control, Tallinn University of Technology (TUT) from the end of 1990s [1].

The system consists of exercises verifying theoretical knowledge and lab experiments for practical skills. It is used in the teaching of many different courses in the computer science field, not only in TUT, but also in other learning institutions in Estonia. Some of the courses have overlapping material. The system started with the classic "topic-based" approach, where the set of tasks were divided into discrete topics. For every topic in a course, student's state of knowledge and level of efficiency was measured. The system is fully web-based, giving students the ability to solve tasks when and where they wish and not being bound to the campus.

Every topic (or competence) has a state that is like a grade showing the level of knowledge in that field. Instead of using the commonly practiced system with 5-6 grades, 128 different ability levels are used in order to avoid discretization effect and have more effective feedback control [1]. Ability level 77 has been historically chosen as a threshold of knowledge acquirement. When the student decides to practice in some topic area, a task or lab experiment is chosen specifically for that student according to the topic, student's ability level, frequency of task taken and some other forgetting parameters. The goal is to use as many different tasks as possible in the topic's pool, try to avoid the student getting the same version of a certain task more than once and get the task closest in its level to student's state on the current topic.

Also, the system supports HomeLabKits – portable lab equipment that gives students the possibility to complete the entire course via the Internet in the time and place of their own choosing. ISC takes the "authentic" student input, meaning that even if there are some multiple choice items, majority of the tasks need students to fill in the answer [2]. It uses an automatic evaluation for the tasks and lab exercises; therefore no teacher interference is needed between students' actions. Students get instant responses thanks to the algorithms used to process answers. These algorithms give relevant comments and calculate results as a feedback. Results vary from -1 to +1, where -1 represents totally wrong and +1 totally correct result.

In the 2010 fall semester, competence-based learning was introduced into the ISC system. Classic topic-based courses were replaced by skill-based competence driven learning to achieve a clearer and more detailed view of student abilities. The main learning goals of the courses remained the same with the switch to the competence-based learning but tasks and topics were divided into smaller units and new exercises were added. Since that time a lot of information has been gathered and conclusions can now be drawn.

B. *Competence-based learning*

The competence-based (or competence-driven) learning is a knowledge based methodology which concentrates on measuring what a person can actually do as a result of learning [3]. It contrasts to classical topic-based learning where certain topics (e.g. Ohm's Law as part of Physics course) are learned and then it is checked whether the student can solve exercises or knows the theory relevant to that topic. As a result of assessment, one grade is given to represent how well the whole material was acquired. Problem with one number representing the knowledge on some wider topic is that it does not measure elementary skill or competence rather than summation of elementary competences (summative grading) and those cannot be used as input in other courses.

For example, if grading the mathematical equation x=(2*5+4)/2, in the classical "topic-based" approach, we check if 'x' is correct and discard the intermediate states/skills as opposed to analysing student ability to add, multiply and divide in competence-based approach. Instead of evaluating one result for each task ('x'), every exercise is analysed to extract smaller parts of skills (e.g. multiplication, adding) that the task uses, checks or requires [4].

Therefore, if we wish to give a proper representation of students' abilities, instead of summating or averaging the grading of skill levels, we should look into each and every competence presented in a task and process them separately.

This work has been partly supported by the Estonian Science Foundation Project 9463

II. MOTIVATION

The main factors and motivations behind switching to the fully web-based competence-driven learning were:

- *Repetitio est mater studiorum*. The aim of learning should be acquiring new skills and knowledge. Cramming in the end of the course for the exam does not serve that purpose. The work should be evenly divided and repetition will make sure that the skill is reinforced.

- The majority of students work (at least half-time but a large portion of students work full-time) during their studies. That limits their ability to attend lectures on the campus. Old-fashioned specific times and dates for specific lab experiments were not suitable for them any more as they were unable to attend them.

- Every student is different and big classroom lecturers do not take that into account. Students tend to work with different speed and have different previous knowledge and interests. It is impossible to take every student's specifics into consideration when interacting with large community at the same time. Personal learning enables students to work on their own time on their own speed without the pressure of running out of time.

- Nowadays students are very capable of adapting to new technology and expect those technologies to be used in learning too. Universities and teaching seems to adapt to changes more slowly even though students are expecting it.

- In the majority of the courses students tend to cram for the exams, concentrating all the learning to the last minute. They go over large amount of information in a short period of time and are able to repeat and use it shortly afterwards to the best of their abilities. After the specific event (like an exam or a class test), the majority of them will not use that information again in close future and it is just forgotten, making the learning void.

- Competence-based learning enables to have personal picture of every student's learning behaviours and current abilities. That gives the teacher a better understanding of class level, what the students already know (even before they come in the first class if they have used that system before) and where the gaps in their knowledge are. It saves time, avoids repeating material student already knows (but what sadly happens quite often) and gives the teacher better idea how to plan course material between lectures.

III. LEARNING, TASKS & GRADES

Exercises in the system have been divided into small units that make analyzing easier for teachers. It also makes it easier for students to concentrate on the small portion of data at a time. Smaller tasks also make those exercises repeatable by students, which is required for new skill fortification. The workload needed to complete the course depends on the student and his/her behaviour and is determined by the system, varying widely. Tasks have 1-13 input competences connected to them. That shows us that even if we divide tasks into smaller units, they still contain more than one skill required to solve them.

Now, in competence-based learning, instead of a pool of exercises connected to certain topic, exercises have been assigned competences. There are two different kinds of competences: input competences, that are measured every time a task is solved (e.g. main skills needed for the exercise), and concealed competences, that may be graded if applied (e.g. if the student makes a specific mistake and it is certainly made due to the lack of knowledge in a certain area, then that competence can be graded but if the student does not make that certain mistake, we cannot tell anything about student's ability in that area). For a course, specific competences/skills are picked that are connected to the field and should be acquired during the learning process. Weights/points will be assigned to those competences and usually the total sum of credit units exceeds the nominal one by 12-15%.

Student does not have specific topics that he/she needs to complete in order to pass the course. Student is offered a set of competences connected to the course. Different competences have different weight to the final result in different courses. Student chooses the competences he/she wants to learn, having much more flexibility than with topic-based learning, giving the whole learning process a personal aspect.

After completing competences for certain required weight to pass the course, system offers a grade and the student has the freedom to pick this or not. The student can accept it and officially finish the course. The alternative is to continue mastering currently acquired competences or to learn new skills to raise the grade. The system offers the grades as soon as the student has completed enough work and it does not correlate with official semester dates. There are no time limits for completing the course from our side. The only limitation is from university regulation that the course declaration expires with 1.5 years. Abandoning deadlines gives students freedom to plan their own work and sense of personal responsibility.

IV. LABS &HOME LAB KIT

Lab experiments test more practical skills and require students to perform real experiments, measure results and to be able to interpret them. The benefit of real experiments is that they can give students unpredictable results and students have to deal with them, trying to find the reasons why they happen [5].

The system uses small lendable kits (figure 1) giving students the possibility to do experiments on their own time and place of their own choosing, still retaining actual physical experiment. If virtual laboratories were used, all the results would be simulated and therefore the experience would not be so accurate. Laboratories should be places where connection between theoretical knowledge and real world is made [6].

Figure 1. HomeLabKit for barchelor students

Lab experiments are mainly used in electrical engineering courses. Currently two different specifications of kits are in use – one for bachelor and other for master students. All the required devices (from multimeters and signal sources to resistors, capacitors and inductors) are in those kits, called HomeLabKit, available to be lent from the university. Experiments with those kits can be done any place where student has internet connection and computer with USB port (USB port is needed for some experiments). Every week students are offered time slots (decided on their suggestions) where supervised help is available in the real lab. Even in the real lab they use their HomeLabKits.

HomeLabKits are flexible as their content can be varied and these variations are reflected by software so experiments may be specific for any kit. Devices used in kits are all individual with component parameters stored in database.

Lab experiments are also divided into small portions, usually requiring one or two measurements. Repetition is important, again. Usually, freshmen are nowadays without any practical experience with electrical devices, measurement instruments etc. Most of the students start with a couple of hours in the lab but after that they will take the kits home declaring that the freedom of choosing the learning time is extremely important for them. The average lending time is about 2-3 days, except distant learners in other locations, who can take the kit for some months.

V. AUTOMATIC EVALUTAION

System uses automatic task evaluation. It takes "authentic" student input, meaning that even if there are some multiple choice items, the majority of tasks ask the student to fill in the answer. It is done by complex algorithms specifically designed for set of similar exercises. There is no need for teacher intervention. Over 300 distinct algorithms are currently in use.

Answer evaluation in competence learning differs from answer evaluation in topic-based learning. As every task is related to multiple competences, answer evaluation will not give one result for the whole task but different results for every competence that is present in the completed task. All the competences and results are independent and therefore some competence levels can rise and others fall at the same time.

The algorithm for that kind of evaluation is much more complex (figure 2). A perfect algorithm for answer (that may consist of several items) analysis would be able to find mistakes, understand the reasons why the mistakes were made and find the associated competences that were used to find the answer. It would also be able to alter given correct answers according to the mistakes made. Therefore, one mistake would not make the rest of the answer fully wrong, even if it depends on earlier calculations. The system has adopted the approach that answer evaluation should simulate the process of student solving the task. By simulating the process weight of repeated mistakes can be lowered, reasons of mistakes identified and therefore useful feedback generated, and only student view to data used.

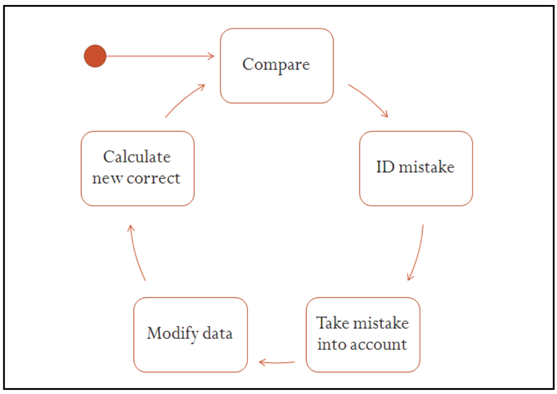


Figure 2. Answer evaluation algorithm

The problem with that kind of approach is that those algorithms are really complex as they have to take account all the possible types of mistakes and competence uses that can be identified. Because of that, those algorithms are usually suitable for only a small portion of exercises and they are not easily re-usable. Deep knowledge of programming is needed to write those algorithms and majority of teachers are not able to do that. So instead of complex simulation of answering process, much easier algorithms tend to be used in practice for many cases, losing the ability of generate proper feedback.

To be able to stimulate student answering process and identify majority of possible mistakes (and therefore give proper feedback), the best source is the log files. Usually when new algorithm is created, the programmer/teacher uses his/her imagination on the types of mistakes students could possibly make and programs algorithm to detect and identify those. After a certain time period (usually 6 months), when enough information has been gathered, the log files for those exercises are scanned and the algorithm is updated according to the mistakes students make in real learning process. We have found that this practice gives the best optimal algorithm efficiency and programmers/teachers never seem to be able to think of all the types of mistakes students make in reality (it also happens quite often in classroom: lecturer does not think about something that is very clear and easy for him but the students may be struggling).

Further work in this area consists of automatic synthesis of evaluation algorithms and possibilities to assign competences to tasks automatically.

VI. FORGETTING MODEL AND LEARNING CONTROL

Important part of the learning control is forgetting as one of the ultimate goals of learning is remembering [7]. The main goal of forgetting model is to evaluate the decreasing of abilities over the time to model student memory behaviour. Human forgetting has been studied carefully over long time and H. Ebbinghaus was the first to show with his experiments that retention is very fast in the beginning, quite slow later and repeating is important to memorization [8].

If the student only solves the task once, the skill is only superficial. Over the time student forgets it. In order to fully acquire the knowledge in long term, student has to repeat the tasks to move newly learned skills from the short term to the long term memory. That is the reason why repetition has an important part in the learning environment. As the number of times one type of task is taken rises, it is important to have many variations of the same task as students strongly oppose getting the task they have already solved.

The main rule in the design of the system has been 3dB rule, meaning that selection of tasks should produce probability of correct answer close to 0.707. To meet 3dB rule, complexity levels assigned to tasks have to be corrected periodically even though those updates level themselves out eventually getting the most accurate complex level found from real learning process.

As answer evaluation is done automatically, there is no new stress for teacher with the repeated taking of a task and the memory model can be used for further personalization. The level of forgetting depends on student's previous behaviour and the correctness of student's answers.

Forgetting model is used in the system to lower student results over time, stimulate process of forgetting and forcing them to retake exercises. Two well known forgetting functions have been used: exponential and power [9]:

$$L(t) = L(\infty) + \left(L(0) - L(\infty)\right)e^{-\frac{t}{\tau}} \qquad (1)$$

where t is time, L(0) is initial level at t=0, L(∞) is limit level and τ is time constant, and

$$L(t) = L(0)\left(1 + \frac{t}{\tau}\right)^{-p} \qquad (2)$$

where p is forgetting parameter. Equation (2) is considered by several authors the preferred option for having one clear advantage: only one parameter determines the forgetting curve as time constant can be set to constant [9]. Exponential form is determined by two parameters which must be varied that is substantially more complex, especially finding final level L(∞) . Therefore, after two years of testing (1) was replaced by (2).

In (2), time constant is set equal to 14 days. That is based on published data [9][10], recommendations and processing previous data from using (1).

VII. ANALYSIS

ISC system logs every action taken by the student and study behaviour with competence-based learning can be analyzed.

A. *Students' results*

One of the main things we wanted to achieve with competence-based learning was getting students to repeat the tasks to fortify the knowledge in areas where the need for fortification is identified with memory model. For that the tasks were divided into smaller units. As seen from the figure 3, the average number of tasks solved by students rose remarkably (as there are different modules for fall and spring semesters, they are given in different figures).

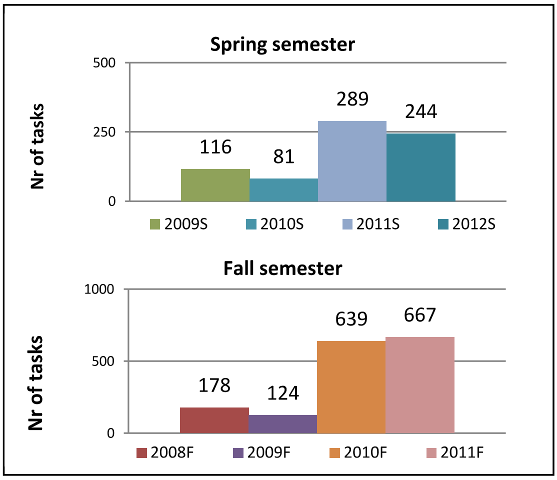


Figure 3. Average number of tasks solved in spring and fall semesters

On average student spends 1.5 minute on one exercise. 43% of students solve the task with 30 seconds. That, sadly, contains students inserting random answers but that behaviour does not help them to make progress. Moreover such behaviour can be detected and applied when evaluating comps. We have achieved that task do not take too long to solve and therefore it is not a problem for students to repeat them.

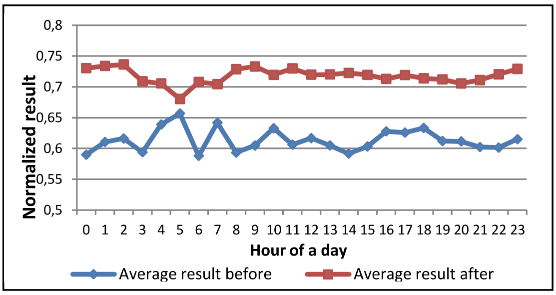


Figure 4. Average result by time of a day

As shown in figure 4, the average result after competence-based learning was introduced is better at all the time slots. It was expected that the results during the night would be lower, as students are more tired than during lab hours where onsite help is offered. However, the results show no clear signs of this. The results around midnight are actually best of the day, reaching the highest peak at 2 am. After this time the results reflect more of what would have been expected, as there is drop in the average results. Overall, the average result is more equal than before as standard deviation dropped.

One thing, that differs remarkably, is marking behaviour. With topic based learning, small set of grades was mainly used to grade exercises. With competence base learning, there are still some grades given more often (-1 and +1), but values between them are used more widely now.

B. Students' study behavior

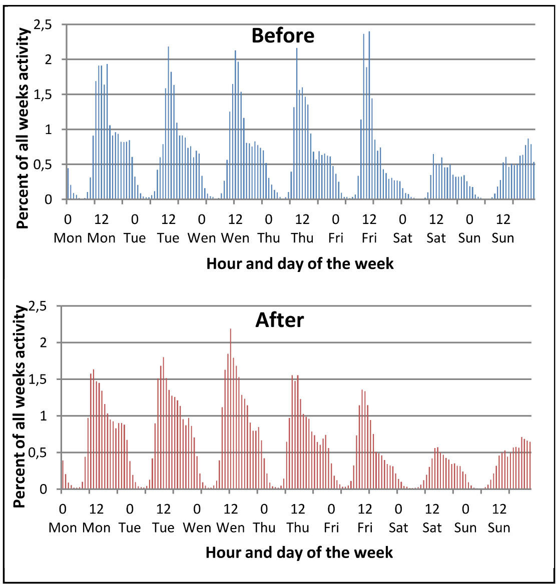


Figure 5. Student weekly activity before and after competence-based learning was inroduced

When comparing students daily study behaviour (as seen from figure 5), it can be seen that before there were larger fluctuations between minimum and maximum. When comparing histograms of days before the days were more equal. Standard deviation seems to prove that visual comparison.

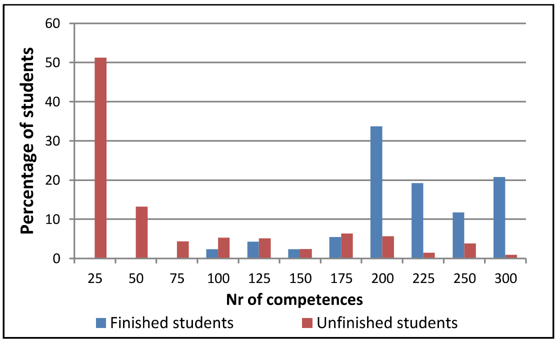


Figure 6. Number of competences students have activated

After competence-based learning, Wednesdays are more active and Thursdays and Fridays are slower days according to statistics. Weekends look to be quite similar. Also, before competence-based learning there seemed to be more peak activities correlating with the offered lab timeslot, while after, the daily activity seems to be more divided, and the drop of activity after usual lab hours is more gradual.

There are total of 387 competences currently (November 2012) in use in the system. On average over all the students registered in the system, student has been graded in 80 different competences. Difference between students who have finished the course and who haven't can be seen on figure 6.

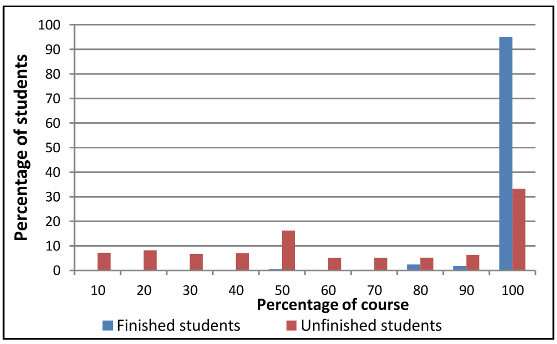


Figure 7. Percent of course connected competences students have tried

In course, to get the final grade, competences need to be confirmed. As courses have different number of competences and they have different weights, the percentage of maximum work possible in a course is taken to be able to compare results in different courses. When comparing percentage of course work without confirmation the results, as seen from the figure 7, a majority of finished students have activated at least 90% of course competences.

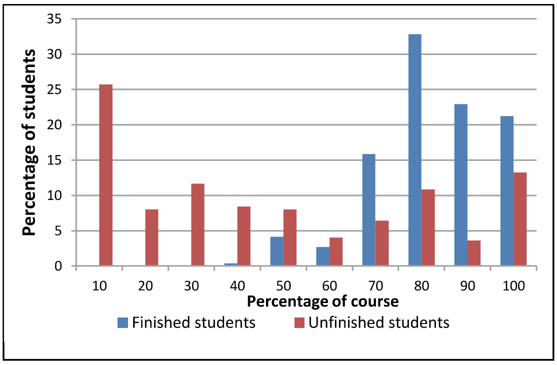


Figure 8. Percent of course connected competences students have been confirmed

When looking at confirmed competences, figure 8, they confirm only portion of them. When looking at the competences that students tend to confirm with their weights to the course, it is clear that the students tend to confirm mostly those with higher weights even if they are thought to be more complex. It is surprising that there are students who are in unfinished group with high number of confirmed competences but have not accepted the mark.

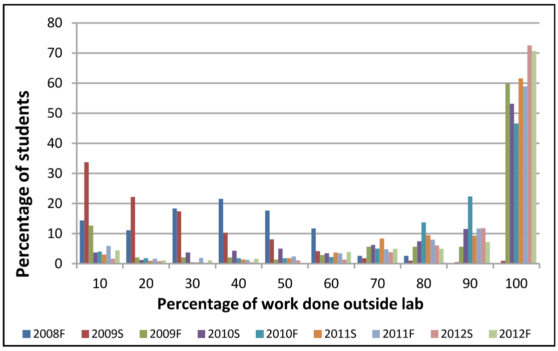


Figure 9. Work done outside laboratory

We can distinguish if student works in the lab or outside it. As shown in figure 9, over the years, the popularity of working outside of the lab has risen. We assume that part of it is due to the students' busy schedule and lendable kits give them the opportunity to work on their own time. Also, it seems that help materials, ways to ask questions remotely and newly introduced small hints give them enough ways to find help without them feeling the need to show up personally. There is a drop in work done only at home when competence-based learning was introduced, as it confused many students and therefore attending lab was more popular on that semester. It showed that it was at first hard to break out of the old accustomed approach to studying but when the students already know what is ahead of them in next semester, the popularity of working independently rose again.

As seen in figure 10, three different sections of piece-wise linear approximations with different slopes are shown. First one is very short reaction including most of the data. The second last from a couple of minutes to 1 day. The third one is from one day to a year (longer periods are not available reliably yet). There is an interesting area between 2 – 9 hours where the average result change is much highter than in its neighbours. That interval may include the night time between studying and may be caused by the effect of psychological time (no forgetting during sleeping) [11]. Also an important point is 4 minutes where curve crosses zero level. Time intervals less than 4 minutes average results increased but after it decreases – forgetting is happening.

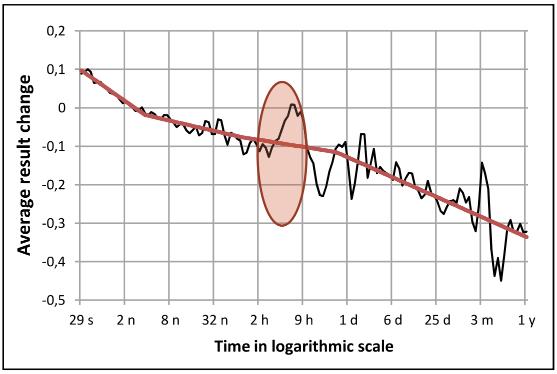


Figure 10. Change in average result over time period between tries

C. *Course completition*

System offers students marks from 1 to 5 ('1' is sufficient, '5' is excellent). '0' marks failing in the course. They can take offered mark or continue working.

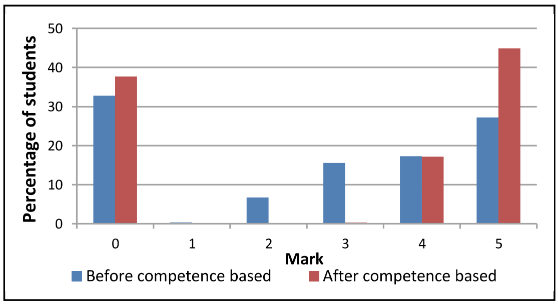


Figure 11. Course marks

After competence-based learning, was introduced, the percentage of students who managed to get the highest mark rose significantly as seen in figure 11. Also, it is quite visible that students preferred to continue working until they got a 4 or 5. (Over those 5 semesters with competence-based learning, only a few students have taken grade 3, with no students settling with mark 1 or 2). The rise in students failing does not accurately reflect the reality as students have three semesters before their declaration becomes void. So many students still have a change to get a mark but are currently in fail section. If we look on the data with the 2010 fall semester, currently the only semester where student declarations have expired, the failing percent has been higher than before.

One thing that competence-based approach seems to have changed is student behaviour towards course completion as seen from figure 12. Only 1% of students now tend to do all the work within 1 month (it's even more surprising as they have to compensate for the forgetting model lowering their grade for 16 week prediction). Majority of students complete the course with 16 weeks (with one semester, showing that they want to be finished before exam session). Exam session was the most popular time before to finish the course (outside factors had an impact: they wanted mark for the right semester). There seems to be constant small portion of students (<10%) who leave the course completion to the later semesters.

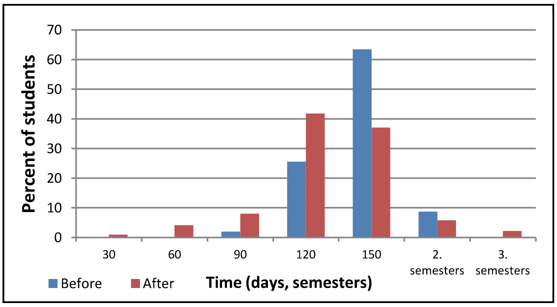


Figure 12. Time to complete course

Student with best marks finished usually within 4 months. The percentage of students doing that has remained the same. Students who take extra semesters to finish the course usually do not get higher mark than 4 (only 4% of 5-s are received after end of first semester). Ironically, the reason why students leave their mark hanging and take extra time is usually that they wish to get better result (next to the excuse that they did not had enough time during the semester to work on the module). Overall, marks correlated to time have remained the same after the switch to competence-based learning but students have started to finish their work earlier.

D. HomeLabKit lending

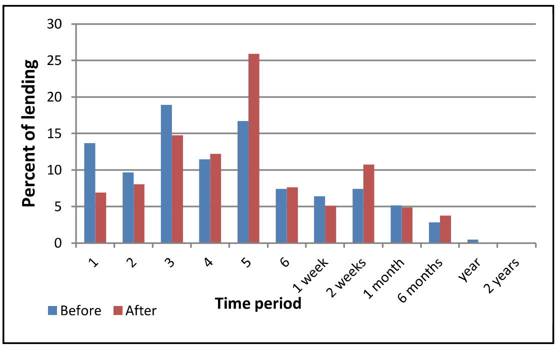


Figure 13. Lending time

Average planned lending of home lab kit after competence-based learning is 4 days for full-time students. Distance learners hold the kit on average for 26 days. Before competence-based learning, the lending tended to be shorter as seen from the figure 13.

In reality, students tend to hold the kits litter longer, meaning they do not return them for deadline. The delay is usually a day caused by not coming to campus that day or simply forgetting. Getting students to return the kits has not been a problem. In only few cases have we been forced to track down the student after longer time period and ask them to return the kit.

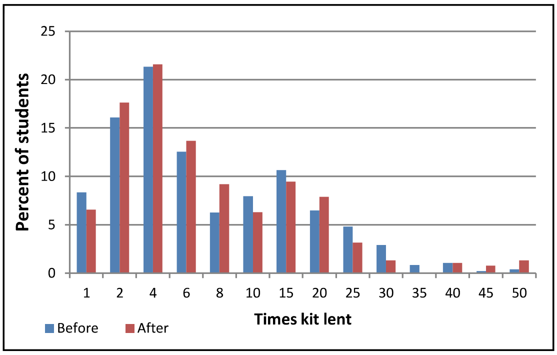


Figure 14. Number of times kit lended

During the semester, student lends kit on average 8 times of 67 days in total. Those numbers have remained the same compared to student behaviour before competence-based learning as can be seen on the figure 14. So, even if they do more experiments now when tasks have been divided into smaller units, it has not made them to borrow kit more frequently.

On average students do 140 experiments per lending (in contrast to 71 before competence-based learning), 60% of those outside the lab as seen from figure 15.

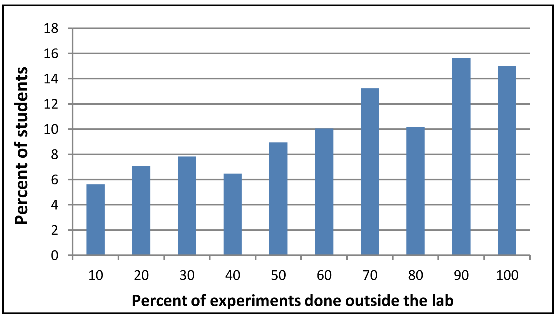


Figure 15. Lab experiments done outside lab turning kit lending period

VIII. CONCLUSION

After two years of practical usage of the competence-based learning, it can be said that switching from more conventional topic-based learning to novel competence-based learning process is not an easy process and leads to confusion among students, but the gain is worth it. The average result rose with smaller tasks concentrating in few skills at the time and repetition forced by the forgetting model. Also, competence-based learning gave students the change to personalize their learning further giving them the change to choose their working place (majority tends to work outside campus), choose what they want to study (by choosing specific competences from the wider pool connected to the module) and decide their time to complete the course and their own final grade. It came out that many students prefer to finish as much work as soon as possible, concentrating on one module at the time and not leave everything to for the exam session. Automatic evaluation stimulating student answering process is used for undisrupted learning process. It has been found that teachers and developers can only guess what the best is, but real mistakes, complexity levels etc will be found over the time with log files and periodical adjustments.

All our motivations for switching to competence-based learning were filled.

REFERENCES

[1] V. Kukk, M. Jaanus, K. Umbleja and O. Shvets “Developing integrated learning environment – analysis, ” Proceedings of International Conference on Engineering Education, ICEE-2010, paper no.166 ,Gliwice, Poland, 2010.

[2] K. Scalise et al, “Assessment for e-Learning: case studies of an emerging field”, International Objective Measurement Workshop, Berkeley, CA , 2006.

[3] M. Jaanus, “The Interactive Learning Environment for Mobile Laboratories”, Tallinna Tehnikaülikooli Kirjastus, Tallinn, 2011,pp 59.

[4] C. Hockemeyer, O. Conlan, V. Wade and D. Albert, “Applying Competence Prerequisite Structures for eLearning and Skill Management”, Journal of Universal Computer Science, 2003 vol. 9, issue 12, pp. 1428-1436.

[5] D.R. Cruickshank, “Teaching methods: designs for learning“, Theory into Practice, 1980, vol 19, issue 1.

[6] L. D. Feisel and A. J. Rossa, “The Role of the Laboratory in Undergraduate Engineering Education”, Journal of Engineering Education, 2005, vol 94, issue 1.

[7] P.A.Wozniak, “Two components of long-term memory”, Acta Neurobiologiae Experimentalis, vol.55, pp.301-305.

[8] C. Ostyn, “Competency Data for Training Automation,” unpublished.

[9] J.T.Wixted and S.K.Carpenter, "The Wickelgren Power Law and the Ebbinghaus Savings Function”, Psychological Science, vol.18, issue 2,pp. 133-134.

[10] K.G. White, “Characteristics of forgetting functions in delayed matching to sample”, Journal of the Experimental Analysis of Behavior, 1985, vol 44, pp.15–34.

[11] A. Tilley, S. Brown, M. Donald, S. Ferguson, J. Piccone et al. “Human sleep and memory processes” in “Sleep, arousal, and performance” Birkhauser, Boston, MA, United States,1992, pp 117

Publication 4

Reference

K. Umbleja, V. Kukk, M. Jaanus, and A. Udal, "New concepts of automatic answer evaluation in competence based learning," in 2014 IEEE Global Engineering Education Conference (EDUCON), 2014, pp. 922-925.

Abstract

This paper introduces current work with automatic evaluation that has been done to enable algorithms generating proper feedback according to the mistakes students have made. Learning environment using novel competence based approach has been used to implement those concepts and over the years data from different development steps has been collected that can be used to verify the benefits to students results of offered algorithm that mimics step-by-step student answering process.

New Concepts of Automatic Answer Evaluation in Competence Based Learning

Kadri Umbleja, Vello Kukk, Martin Jaanus, Andres Udal
Department of Computer Control
Tallinn University of Technology
Tallinn, Estonia
kadri.umbleja@ttu.ee

Abstract—**This paper introduces current work with automatic evaluation that has been done to enable algorithms generating proper feedback according to the mistakes students have made. Learning environment using novel competence based approach has been used to implement those concepts and over the years data from different development steps has been collected that can be used to verify the benefits to students results of offered algorithm that mimics step-by-step student answering process.**

Keywords—feedback; automatic evaluation, assesment, competence, competence based learning.

I. Backround

A. Competence based learning

There are many different definitions for the term competence. Klarus has defined it as mix of skills, attitudes and knowledge that makes the employee or graduate successful in society and his/her profession [1]. Other definition states that by competence, we understand good performance in diverse, authentic contexts based on the integration and activation of knowledge, rules and standards, techniques, procedures, abilities and skills, attitudes and values [2].

We have defined competence-based (or competence-driven) learning as a knowledge based methodology which concentrates on measuring what a person can actually do as a result of learning [3]. It contrasts to classical topic-based learning where certain topics (e.g. Ohm's Law as part of Physics course) are learned and then it is checked if student can solve exercises or knows theory relevant to that topic. As a result of assessment, one grade is given to represent how well whole material connected to the wider topic was acquired. Problem with one number representing the knowledge is that it does not measure elementary skills/competences rather than summation of those elementary competences (summative grading).

The competence-based approach divides those wider topics into atomic portions of theory (elementary skills of knowledge that cannot be divided into smaller units). It checks all the elementary skills/competences separately and also grades them independently.

For example, if grading the mathematical equation x=(4*8+10)/2 in the classical "topic-based" approach, we check if 'x' is correct (21 in current case) and discard the intermediate states/skills as opposed to analysing student ability/skill to add, multiply and divide in competence based approach. If student's answer would be 11, in "topic-based" approach, it is considered to be a wrong answer. Using competence based approach, answer 11 offers possibility that student mistook multiplication as addition (as (4+8+10)/2 equals 11). So, instead of saying that answer was totally wrong, we might say that in this case, that student can add and divide but lacks competence or skill in multiplication. In competence based approach, instead of evaluating one result for each task ('x'), every exercise is analysed to extract smaller parts of competences (e.g. multiplication, adding) that the task uses, checks or requires [4].

Therefore, if we wish to give a proper representation of students' abilities, instead of summating or averaging the grading of the wider topic, we should look into each and every competence presented inside a task and process them separately.

B. E-learning environment ISC

ISC is an e-learning environment used in the Department of Computer Control, Tallinn University of Technology (TUT) from the end of 1990s [5]. The system started with the classic "topic-based" approach, where the set of tasks were divided into discrete topics. For every topic in a course, student's state of knowledge and level of efficiency was measured. System also implements memory forgetting model to mimic forgetting over time [6].The system is fully web-based supported by HomeLabKits [7] – portable lab equipment, giving students the ability to solve tasks when and where they wish and not being bound to the campus. In the 2010 fall semester, competence-based learning was introduced into the ISC system. Classic topic-based courses were replaced by skill-based competence driven learning to achieve a clearer and more detailed view of student abilities.

The courses in the system contain of set of elementary competences/skills. Those competences have different weights according to their importance in the course. There is overlap in the points that student can achieve in the course so that he has an ability to choose what he studies. Tasks in the system have competences connected to them according to the skills needed

This work has been partly supported by the Estonian Science Foundation Project 9463

to solve them. System offers student a field of competences where they have full control over the learning process by choosing which competence they want to learn and excel. System displays them an exercise according to the chosen competence, current level of knowledge in that competence and forgetting parameters.

II. MOTIVATION

As the learning system is fully autonomic and does not require tutor interception for fully continues learning process, it is system's responsibility to guide the student in the learning process and give him/her proper feedback according to the mistake made. Making mistakes and overcoming them in natural part of learning process [8].

The given feedback has to enable students to improve, understand their mistakes and what is expected on them [9]. Therefore, the aim of the feedback generated by the algorithm is not to show the student correct answer (what would require the student to extract the part of knowledge they faulted against by themselves) but to offer elaborated feedback by pointing them directly to the missing competence and (if possible) specifically explain the reason why they made the error in cases when the type of mistake is common [10].

III. AUTOMATIC EVALUATION

ISC system uses automatic task evaluation. It takes “authentic“ student input, meaning that even if there are some multiple choice items, the majority of tasks ask the student to fill in the answer. Automatic evaluation uses complex algorithms designed specifically for set of similar exercises. Those algorithms find reasons of mistakes generate feedback according to them and try to mineralise the impact of repeated mistakes. There is no need for teacher intervention. Also, that kind approach assures that assessment is fair and consistent.

Answer evaluation in competence learning differs from answer evaluation in topic-based learning. As every task is related to multiple competences (even if we tried to divide exercises into small units, in majority of cases, they require more than on elementary skill to solve them), answer evaluation will not give one result for the whole task but different results for every competence that is present in the completed task. All the competences and results are independent and therefore some competence levels can rise and others fall at the same time.

A. Algorithm Mimicing Student Answering Process

If we want to achieve best feedback generation for the student according to mistakes made, the most logical way would be to mimic student's behaviour step-by-step when solving the excises. Usually, finding the final answer contains multiply middle-steps with their own results. If one of those results is wrong, the mistake is carried over and the final answer will also be wrong. When following the steps students take and using only data generated by the algorithm during the process, only actual mistakes are discovered. By simulating the process, the weight of repeated mistakes can be lowered; reasons of mistakes identified more accurately and therefore useful feedback generated, and only student view to data used. The algorithm for that kind of evaluation is much more complex (figure 1).

We are particularly interested in a student's incorrect answers as there is much to learn from them – what went wrong, why it went wrong, where the answer went wrong, what competence should be more focused on when teaching etc. Best source of finding most common and all possible types of mistakes are old log files. Usually when algorithm is created, set of few most common mistake types identified by the professor or algorithm creator are used. After a certain time period (usually 6 months), when enough information has been gathered, the log files for those exercises are scanned and the algorithm is updated according to the mistakes students make in real learning process.

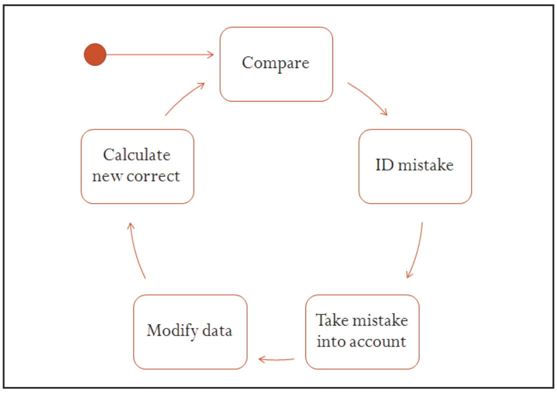


Fig. 1. Answer evaluation algorithm

The problem with that approach is that those algorithms are really complex as they have to take account all the possible types of mistakes and competence used that can be identified. Because of that, those algorithms are usually suitable for only a small portion of exercises and they are not easily re-usable. Analysis shows that currently there are over 1000 algorithm in use. Some of those algorithms are really similar with only minor differences. When analysing only their structure, 300 distinct types can be identified. That shows that a lot of work goes into writing those complex algorithms even if there is only need for a minor difference. Therefore, it would be reasonable to find some better system for producing automatic evaluation scripts

B. Automatic Synthesis Of Answer Evaluation

As maintaining and writing algorithms mimicking student answer process is very time consuming, mistakes are hard to find and identify. Also, those algorithms tend to be not re-usable. Therefore, a better solution has to be achieved in order to facilitate huge number of exercises that are essential for diverse learning environment with many different kind of tasks. As all those exercises in the system use some set of rules that are applied to input data to derive the answer, the idea is to come up with common “world rules” that can be used with all of them and then apply algorithm that is able to understand, taken input data, which rules to apply. The result would be one system, so called expert system, that can be used to derive correct answer and also for mimic step-by-step student answering process. That system would also contain another set of rules for mistakes that are applied if inaccuracy is detected.

That kind of solution would not require to have many different algorithms with only minimal differences. Only one copy of the system is maintained. Rete algorithm is used as a motor for pattern matching between world rules and current student input.

Rete algorithm is efficient algorithm for a large collection of patterns connected with a large collection of objects [11]. It has many possible applications and should also be suitable for storing set of "world rules" for answer evaluation and matching given information with rules [12]. A basic production system checks each if-then statement to see which ones should be executed based on the facts in the current dataset, looping back to the first rule when it has finished. The computational complexity is of the order $O(RF^P)$, where R is the number of rules, P is the average number of patterns in the condition part of the rules, and F is the number of facts in the knowledge base [11]. Algorithm tries to derive all possible new knowledge from rules, not only looking for specific knowledge.

The algorithm is efficient even when it processes large sets of patterns and objects, because it does not iterate over the whole set [12]. As a result, the computational complexity per iteration drops to O(RFP), or linear in the size of the fact base [11]. Patterns are compiled into a program to perform the matching process. The program does not have to iterate over all the patterns because it contains a tree-structured sorting network or index. It also does not have to iterate over the data because it maintains state information: the program computes the matches and partial matches for each object when it enters the data to the memory, and it stores the information as long as the object remains in the memory [12].

Rules given for Rete Algorithm are usually given in following form: if <condition> then <action>. For example, resistors in parallel connection can be written down as following rule: If there are two elements, that are connected in parallel connection and both of those elements have resistance available/assigned, then the resistance of element created when combining those two in parallel connection, equals $1/(1/R_1+1/R_2)$.

Currently, simple well known truths in electrical engineering field have been implemented as rules usable for Rete algorithm in Drools system and prototype system is able to solve correctly simple exercises where those rules are used.

IV. ANALYSIS

New algorithms mimicking student behaviour step-by-step were first introduced at Operation System course. That course was chosen as the number of students taking it every year is large enough but it is not one of the biggest thought with the system. Also, that course contained many complex exercises were students felt that if they get the wrong answer, they do not understand what they did wrong.

So, those exercises in that Operation System course that were later transformed using our novel approach to give student better feedback and grade them more accurately, form "testset". In the same time, there are other courses that ran in the same system that contain exercises outside test pool. Those exercises form a group "others" and that data is used to verify over the same time period that the change in student result is not only caused by other factors (change in teaching approaches, hard working students etc)

A. Students' result before competence based approach was introduced

Figure 2 shows that before competence based approach was introduced, that exercises in the test set had almost half a change of getting correct answer compared to others. Notably, possibility of getting totally wrong answer was in the same level for both exercise sets. The difference comes from middle values that were much often given to test set exercises. Figure 2 proves that statistically that students' feedback that they had hard time with those exercises was a verifiable from the data.

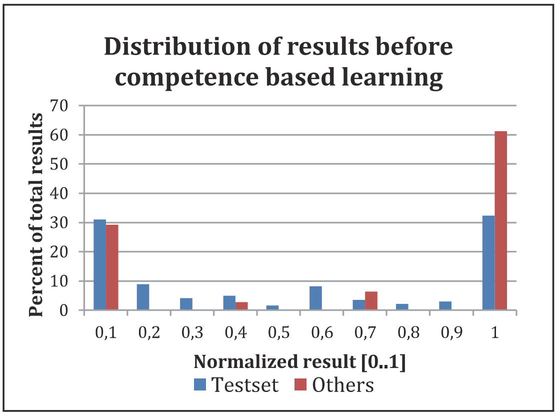


Fig. 2. Distribution of results before competence based learning

B. Studens't results with competence based learning

When competence based learning was introduced, students got detailed feedback by every competence used in the exercise that they solved. That gave them a better overview to understand what might have gone wrong even though written feedback identifying exact mistakes were not yet returned to students.

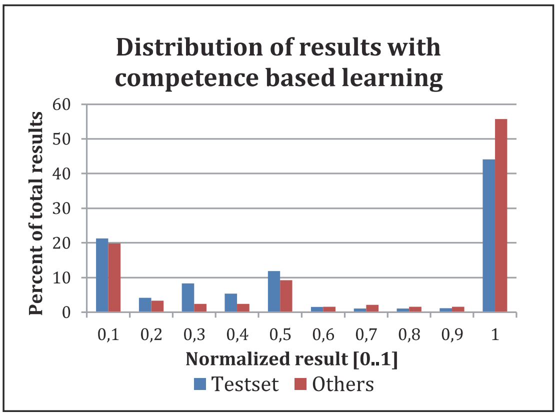


Fig. 3. Distribution of results with competence based approach to learning

Figure 3 shows that percentage of wrong answers has fallen for both sets. Test set exercises are now having much higher change of getting totally right answer showing that competence based learning had an effect of student learning. Also, much more results between wrong and right are used to the other set instead only using those two grades.

C. Students' result with novel algorithm mimicing student answering process in competence based learning

Novel approach to algorithm mimicking student answering process seems to achieve the goal expected of it as can be seen on figure 4. The probability of getting right answer rose remarkably, even past the set containing other exercises. So, we would say that granting student detailed feedback about their mistakes helps them to learn from their errors and give them better probability to gain better result next time solving exercise with similar skills needed.

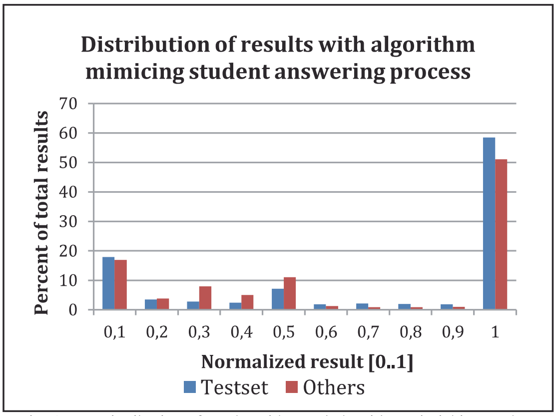


Fig. 4. Distribution of results with novel algorithm mimicking student answering process

In the mean time, the change of getting right answer for the other set has dropped almost 5%. That might have been caused by a many different factors – different students with different kind of learning behaviours, internal changes in the courses.

V. CONCLUSION

Feedback is crucial for students to learn from their mistakes. In order to be able to facilitate huge number of students, automatic evaluation is only plausible solution. The problem in current case has been that it is not easy to give detailed feedback with automatic evaluation. First step has been to introduce competence based learning where every elementary skill or competence is graded separately giving more detailed overview of students' abilities. Also, that approach enables student to see in which competence they made a mistake after submitting result. Taking that one step further, algorithms mimicking student answering process were developed that were able to specify exact point where the mistake was made and therefore generate detailed feedback specifically for that error. Also, mistakes are not carried over as only current state of data that student uses while working is considered. Those algorithms are really complex and there is work in process to develop one expert system with knowledge specific rules that would be able to reduce the complexity of the algorithms.

REFERENCES

[1] H. Rozeboom, "Competence based learning", Bulletin of University of Agricultural Sciences and Veterinary Medicine Cluj-Napoca. Horticulture 2006 vol63, issue 1-2, pp. 365-367.

[2] A.V. Sánchez, M. P. Ruiz, "Competence based learning - A proposal for the assessment of generic competences", University of Deusto, Bilbao 2008, pp. 29.

[3] M. Jaanus, "The Interactive Learning Environment for Mobile Laboratories", Tallinn Univeristy of Technology Press, Tallinn, 2011,pp 59.

[4] C. Hockemeyer, O. Conlan, V. Wade and D. Albert, "Applying Competence Prerequisite Structures for eLearning and Skill Management", Journal of Universal Computer Science, 2003 vol. 9, issue 12, pp. 1428-1436.

[5] V. Kukk, M. Jaanus, K. Umbleja and O. Shvets "Developing Integrated Learning Environment – Analysis, " Proceedings of International Conference on Engineering Education, ICEE-2010, paper no.166 ,Gliwice, Poland, 2010.

[6] V. Kukk, K. Umbleja "Analysis of Forgetting in a Learning Environment", Proceedings of 13th biennial Baltic Electronics Conference, BEC 2012, pp. 335 – 338, Tallinn, Estonia, 2012.

[7] K. Umbleja, V. Kukk, M. Jaanus, O.Shvets "Laboratory Experience with Competence Based Learning" Proceedings of International Conference on Engineering Education & Research 2013, ICEER-2013, paper no.94, Marrakesh, Morocco, 2013.

[8] M. Tulis, "Error Management Behavior In Classrooms: Teachers' Responses to Student Mistakes", Teaching and Teacher Education 2013, vol 33.

[9] D. R. Sadler, "Formative Assessment and the Design of Instructional Systems", Instructional science 1989, vol 18, issue 2.

[10] R. Stark, V. Kopp, M. Fischer "Case-based Learning with Worked Examples in Complex Domains: Two Experimental Studies in Undergraduate Medical Education", Learning and instruction 2011, vol 21, issue 1

[11] M. L. Lee, W. L. Tok, L. L. Wai "IntelliClean: a Knowledge-Based Intelligent Data Cleaner" Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2000.

[12] P. Nayak, A. G. Pandurang, S. Paul "Comparison of the Rete and Treat Production Matchers for Soar", AAAI 1988.

Publication 5

Reference

K. Umbleja, "Students' grading control and visualisation in competence-based learning approach," in 2015 IEEE Global Engineering Education Conference (EDUCON), 2015, pp. 287-296.

Abstract

This paper analyses how students familiarize themselves and cope with novel competence-based approach to learning in online elearning environment where focus is on personal learning. Learners have freedom to mark their own path through the courses without deadlines, choose themselves what, when and how much they learn. That methodology concentrates on measuring what a person can actually do as a result of learning. In order to have that kind of detailed picture of everyone's abilities, different kind of approach for assessment had to be developed using automatic algorithms that mimic student answering process with small repeatable exercises. As every competence is being graded separately with high level of granularity, one number final grade requires a more complex aggregation function to be designed. Grades are suggested for students when enough work is done and on their own discretion they can accept that result or keep working to achieve higher grade. As this kind of personalized approach to learning leaves all the responsibility of planning to learner, they have to have proper overview of their own process and possibility to access sufficient data in order to make educated learning decisions. To facilitate that problem four different student's progress visualization tools were offered for students. Their usage, benefits and drawbacks according to students' feedback are analyzed. The conclusions drawn are that visual tools have helped majority of students to follow their process better than before and every student prefers different kind of tools depending on their personality. When some students have benefited a lot from freedoms offered by the system, some students have hard time motivating themselves to work and therefore have troubles completing the course on time.

Students' grading control and visualisation in competence-based learning approach

Kadri Umbleja
Department of Computer Control
Tallinn University of Technology
Tallinn, Estonia
kadri.umbleja@ttu.ee

Abstract— **This paper analyses how students familiarize themselves and cope with novel competence-based approach to learning in online e-learning environment where the focus is on personal learning. Learners have freedom to mark their own path through the courses without deadlines by choosing themselves what, when and how much they learn. That methodology concentrates on measuring what a person can actually do as a result of learning. In order to have that kind of detailed picture of everyone's abilities, different kind of approach for assessment was developed using automatic algorithms that mimic student's answering process with small repeatable exercises. As every competence is being graded separately with high level of granularity, one definitive final grade requires a more complex aggregation function to be designed. Grades are suggested for students when enough work has been done and on their own discretion they can either accept that result or keep working to achieve a higher grade. As this kind of personalized approach to learning leaves all responsibility of planning to a learner, they have to have a proper overview of their own progress and an access to sufficient data in order to make educated learning decisions. To facilitate that problem five different student's progress visualization tools were offered to students. Their usage, benefits and drawbacks according to students' feedback have been analyzed. The conclusions drawn are that visual tools have helped majority of students to follow their process better than before and every student prefers different kinds of tools depending on their personality. While some students have benefited a lot from the freedoms offered by the system, others had hard time motivating themselves to work and therefore found it to be difficult to complete the course on time.**

Keywords— competence, assessment, visualisation, grade suggestion, personalised learning

I. Introduction

Learning, as in any other field, is a constant progress of innovation and improvement. Some side effects of those reforms cannot be foreseen or are simply missed in analyses phase. Therefore, they only manifest themselves after new features have been launched. When switching from *classical topic-based approach* to novel *competence-based approach* that concentrates on small atomic portions of learning material called *competence*, it was expected that students will have many more ways to personalize their learning. That, for example, involves students studying without specific timetable for every week, working without a deadline and having freedom to complete the course before the end of the semester. It also gives students the freedom to choose what they learn, when they learn and how they learn it. Those changes were welcomed by the students as nowadays learners need much more flexibility than they used to as majority of students also work while studying. Work commitments tend to overshadow their learning responsibilities – students say they are unable to attend laboratories in specific times, want to have freedom to create their own timetable that would allow them to combine both working and studying in contrast to strict rules and schedule that university wants to force on them.

Competence-based approach was expected to give students the flexibility they complained they lacked before. The side effect of those changes was that responsibility of mapping his/her own path through the course fell to the student. When some of the students were really good at planning and used new features and freedoms to their fullest, some students had hard time being masters of their own personal approach to learning. It merged that when competence based approach tries to offer very detailed view of knowledge acquiring, it also desegregates the general overview of student's progress. When learning is led by competences, the overall progress in course as a whole becomes dimmed. Students do not grasp how much time and effort they need to complete the course. Without deadlines they get the false sense of security and leave everything until the end of semester. Some of them gram at the end and somehow manage to finish the course. Others, sadly, fail to do that.

In order to address that problem, it was clear that some sort of tool for following the progress was required. That tool should be easy to use and enable students to dissect their own behavior. After some analysis, it was decided that the best solution would be using visual tools. As every learner is unique, people's abilities to comprehend information via visual tools is different. Therefore, in the end, five different tools are offered. Some of the functionalities are the same, some are different. After introducing those changes to the system, questionnaires and interviews were used to collect feedback and draw conclusions.

II. Competence-Based Approach

Competence-based or competence-driven learning has become more and more popular of late. It has been adopted by many researchers and applied in many institutions [1,2,3 e.g.]. There are many different definitions for the term *competence*. Klarus has defined it as a mix of skills, attitudes and knowledge that makes the employee or graduate successful in society in his/her profession [4]. Other definition states that by competence, we understand good performance in diverse, authentic contexts based on the integration and activation of knowledge, rules and standards, techniques, procedures, abilities and skills, attitudes and values [5].

In our approach, competence-based learning is defined as a knowledge based methodology which concentrates on measuring what a person can actually do as a result of learning [6]. It contrasts classical topic-based learning where learning is focused on certain topics (e.g. Ohm's Law as part of Physics course) and then it is checked if student, in general, can solve exercises or knows theory relevant to that topic. In competence based learning, those topics are divided into smaller units, called competences, which should be acquired throughout the course. In our case, we have extracted atomic portions of theory (elementary skills of knowledge atoms that cannot be divided into smaller units) from previously used wider topics that is the foundation of our methodology.

To clarify that approach, the following example can be used. In classical "topic-based" approach, if simple mathematical equation $x = (4\times8+10)/2$ is analyzed, it is only checked if the x is correct. Intermediate steps like addition, multiplication and division used to getting the final answer are ignored. In competence-based approach, student's skills of addition, multiplication and division are also analyzed and graded. If final answer given by student is 11 (instead on correct answer 21), it can be deducted that the student made a mistake with multiplication part of the task (adding 4 and 8 instead of multiplying). Division and addition skills can be graded with high marks, but multiplication skill with low a mark as that was the actual competence where student's knowledge was inadequate.

Using competence-based approach enables us to give students (and teachers) a detailed and proper representation of their abilities. Instead of summating or averaging the grade over a wider topic, we should look into each and every competence presented inside an exercise and process/grade them separately. That kind of methodology rises many challenges for grading exercises but gives student proper representation of their current state in the course. It also raises many exciting research questions.

In competence-based learning environment, it is recommended that exercises would be divided into smaller units that concentrate on only a few competences simultaneously. It would be ideal if an exercise focuses only on one competence at a time but in practice it has been revealed that, even the simplest tasks usually require usage of more than one elementary knowledge or skill. Having small exercises makes assessment easier. Also, it does not take long to complete the tasks and therefore exercises become repeatable.

Repetition is a vital concept for competence-based approach as the methodology concentrates to achieve firmly acquired knowledge as a result of learning. In order to fortify the knowledge it has to move from short term memory to long term memory. Human forgetting has been studied thoroughly and H. Ebbinghaus was the first to show with his experiments that retention is very fast at the beginning, quite slow later and therefore repeating is important to memorization [7,8]. Therefore, trough repetition it is made sure that newly acquired knowledge will move from short term memory to long term memory and is reinforced.

Courses do not consist of specific topics rather than list of competences that should be acquired by the students when they finish the course. Every competence in a course has a weight connecting to it illustrating its prominence in that specific course. Weights can also be seen as a points towards course competition when summed up equaling to the specific number of European credit units assigned for the course from official curriculum. In order to personalize the learning [9], courses usually have 10-20% overlap when summing up competences' weights. It does not force students to acquire every one of the competences listed in course but to make a choice according to his/her personal needs and interests. The weights are chosen so that it is not possible to complete the course without acquiring the main topics and fundamental ideas from the courses. Also, in some courses the weights are distributed so that it would not be possible to finish to course without acquiring competences requiring the use of lab equipment.

Therefore, to complete the course, students do not have a specific number of tasks or assignments they have to complete in order to pass. Students solve tasks (small exercises about theory taking no longer than 2-3 minutes) or lab experiments (practical, hand-on exercises that can be done remotely via HomeLabKit [10]).The number depends on every student's personal learning behavior – how much time is between learning sessions and using specific competence, how correct are the answers, and how many times certain mistakes are made etc. Data has shown that the amount of work done by students to complete the course may vary even as widely as 6-7 times (only time spent in e-learning system can be measured).

III. MOTIVATION

Using classical "topic-based" approach, it is relatively easy for a student to keep track on his/her progress. There are only a certain number of limited tests/tasks that have to be completed to finish the course. Every task in the course usually gets its own mark represented by a number or passed/not passed. It is usually known how much weight every subtask has over final mark. With competence-based approach that kind of "familiar" system for the student could not be used in order to achieve maximum learning result (competence acquisition).

As the main goal of learning should be acquiring new knowledge, we give students the chance to work as many times as they wish on the tasks until they (and the system) is happy with their competences acquisition. We achieve this by using online e-learning system enabling student to work when and where they wish. E-learning system has intelligent algorithm guiding the automatic answer evaluation process [7]. This way learning does not depend on human resource and an increase in number of tasks done by students does not elicit increase of workload for teachers. Student selects competence they want to learn, algorithm chooses an exercise most suitable for learners current state of knowledge and needs using *forgetting model* to mimic memory model over time [8]. After submitting the answer (filling the blanks, choosing correct answers from the list e.g.), algorithm automatically and without a significant time delay (few seconds), identifies correctness of student's answer and competences used to solve given task. Then, algorithm reports changes of competence levels and, if necessary, gives feedback according to mistakes made.

Therefore using that kind of methodology for grading student's submissions requires specific approach how to derive final mark and how to inform student of his/her current state as the number of tasks required to complete the course is not constant and depends on every student's personal learning speed and progress.

Our methodology has been in use since 2010 fall in courses connected to electrical engineering. Majority of the students use the system to take undergraduate courses dealing with foundational principles. Some graduate courses are also offered but they tend to have fewer participants (6% of all students using the system). Every semester around 400 to 600 students use our system that has been in constant evolution according to the analysis of data collected by the e-learning system, feedback from the students and teachers experience in using the system. Since switching to competence based approach, 3950 declarations have been made to courses taught with the system from more than 2000 distinct learners.

IV. Grading

In topic-based methodology, one grade is given as a result of an assessment to represent acquisition of a larger portion of material. Using one number to represent knowledge of some wider topic does not measure elementary skills or competences but rather offers summation of students' knowledge of elementary competences (summative grading). Also, that kind of summative result cannot be used as input for other courses and it lacks granularity required for a clear and detailed picture of student specific abilities.

Therefore, competence-based approach requires different kind of strategy for grading than topic-based methodology. In order to have detailed overview of student progress, grading has to be in-depth and extract detailed results.

To achieve desired granularity and fine-detail results, every competence in the system has its own score/ability level that represents student's progress in acquiring a specific atomic competence. When student solves exercises, every competence that is used by the student to solve the task is graded separately and independently, meaning that some competences can go up (if they were used correctly in the task) in the same time when others can go down (when those competences were required in places where mistakes were made).

In order to facilitate that kind of fine detail grading without extra workload for the teaching-staff, automatic answer evaluation is used. It is achieved by using complex algorithms mimicking student answering process. If the final answer is wrong, algorithm tries, step-by-step, to find the place where mistake was made and identify reasons behind it. Algorithm also generates proper feedback according to the mistake made. That kind of algorithm is a fair evaluator lacking subjective bias and is always guaranteed to grade similar answers the same way.

Every exercise has two sets of competences connected to it – first are the competences, called *input competences*. Those competences are always present in the exercise and they are always graded, even if mistake is made. The second set contains competences that do not manifest every time exercise is solved. Competences from the second set may appear only when student submits totally correct answer or when some specific kind of mistake is made.

Furthermore, to achieve even higher granularity and in order to minimize effects of discretization, 128 different ability levels are used instead of commonly used 5 or 6 grades. Level 0 means that student has no knowledge of that competence and level 127 represents full and solid knowledge. There is another important level, 77, that is called *acquisition level*. It is used as a benchmark for student being able to pass the course with minimal result. It is expected that student is able to apply (Bloom's taxonomy level 3 [11]) acquired competence and use it in further basic exercises correctly. Also, it is expected that student continues to fortify the knowledge about that specific competence and acquire higher competences' and Bloom's taxonomy levels by continuing working in the system.

After student has reached acquisition level with some competence, he/she has to confirm the knowledge in class test on campus (all the other time, learning, including lab experiments, can be done outside classroom. For lab experiments student can lend small kits with required equipment). Class test uses the same exercises that student has solved before and contains only competences that has state higher than acquisition level. So, in theory, student should be able to answer given task correctly. Data analysis shows that it is true in 63% of cases of theoretical tasks and 59% of cases of laboratory experiments. Two-level system is used in order to verify that actual student has done the work (with e-learning, there is always a possibility that someone else has been solving the exercises).

A. Grade-suggestion motor

Even though e-learning system uses its own methodology for grading and assessment, in the end, official system requires one final grade for the course. We would prefer solution that would be most flexible for the student and would leave our detailed approach relatively intact.

As every competence has been graded separately using 128-level, some kind of aggregation function has to be used to extract the final mark. One possibility would be to transfer every competence from 128-level to system using 5 grades. Then competence's weight/importance for the course could be considered and final mark could be extracted. That kind of approach would result loss of detail as grades will be sampled from high detail scale to lower detailed scale.

In order to keep the detailed results and avoid loss of granularity, more complex aggregation function, grade-suggestion motor, was developed. It takes into account the official number of credit units assigned for the course. Current curriculum guidelines for grading were taken into account when developing this motor. Those regulations state that student should be able to pass a course with 51% of work done/knowledge acquired. Therefore, it is set that if student achieves highest levels for competences worth half the points, he or she should be able to pass. Also, if student confirms competences equaling 100% of required points to minimal level, he or she should also be able to pass. Those two points are used to derive a line. Below that line – student cannot pass the course. Above the line – grade-suggestion motor is able to suggest a grade. There are exactly 50 levels between acquisition level (77) and maximum (127). Therefore steps of 10 levels are used to define line-functions for other grades in a system where 5 grades are used.

The formula that defines the line for pass on the course takes the following shape:

$$y = \frac{100 \times (x-A)}{-A} + 77 \quad (1)$$

Where y is ability level needed, x is credit units earned and A is credit units assigned for the course by curriculum. Formula 1 clearly shows that if x equals A (all credits done), y equals 77 – acquisition level. And if x equals A/2, then (x-A) is receded to –A/2 and the first part of the formula (division) yields 50. 50+77 equals maximum level for competences that is 127.

Formula 1 can be evolved to take into account all possible grade lines:

$$y = \frac{100 \times (x-A)}{-A} + 77 + 10 * (G-1) \quad (2)$$

where G represents the desired grade. It can be seen that if desired grade is 1, then last part of the formula disappears and the same form as formula 1 is achieved.

When student completes a task during the class test confirming competences, grade-suggestion motor is called by the system. Student's current confirmed competences and their levels are used as input for the function. If the formula returns a grade, it is suggested for a student. Student has a possibility to take the grade or keep working on the course to achieve a better final grade. If students does not take offered grade, he or she can always resume and take it later by specific functionality on the system – as long as confirmed competences' weights and their levels remain high enough for a grade-suggestion motor to be eligible for a grade.

The other possibility for a student is to continue working and acquiring more competences or achieving higher levels with currently acquired competences. If new grade becomes eligible, it is again, suggested for a learner. Each grade is promoted for a student once, first when it becomes available, therefore minimizing noise and extra actions required by a student.

B. *Impact*

Data analysis shows that majority of students do not take first grade offered by the system but continue working until they get a higher grade (in 5 grade system, where 1 is the lowest and 5 is the highest grade, 5 is the most popular choice as seen in Fig. 1).

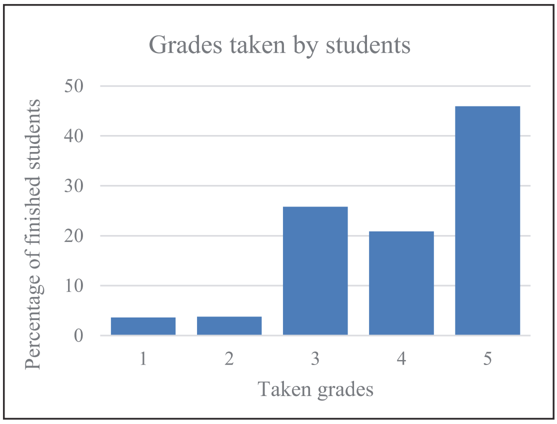


Fig. 1. Grades taken by the students with grade-suggestion motor.

Using that kind of grade-suggesting motor, students have the ability to finish the course as soon as they have acquired enough knowledge and are not bound by exam session dates. Only limitation to our "deadlineless" approach is the one limitation enforced by the official system - end of the semester date assigned by the university. It is a last date when grades can be carried from our e-learning system to an official registry and therefore students have to take a grade by that date or they achieve "fail to appear".

This kind of approach enables student to plan his/her workload personally – something that is not common for students. It happens quite often that all the different courses have their big tests (for example, mid-semester tests) on the same week. Therefore, student may have weeks were the study load is very high and then weeks where there are not too many things needed to be done. Using personal approach to workload, at least with courses we offer, student can choose by him/herself when to work on the course more actively and when to take a break in order to concentrate on other things. Also, students have stated that this approach suits well for those who work half or full-time besides university studies.

Also, as all the courses in the system are given via e-learning form, means that there are only few lectures at the start of the semesters or no lectures at all. Therefore, student does not have to wait until some material is introduced by the lecturer. Student can start working on the material on his/her discretion. Also, there are no deadlines or specific test dates – class test dates are offered every week, usually every day. Furthermore, students have a tool in the system to indicate best time slots for them that are bases for assigning places for class tests and on-campus sessions (where they can solve exercises in the system with supervised help). Student has full freedom of choosing his/her own personal timetable.

That kind of approach has created interesting phenomena – some students complete course within first few weeks. Some of them even manage to finish before the official university date for course declarations (week 2) as seen on Fig 2. Feedback from those students has shown us that at the start of the course, the general workload is very light and therefore they have plenty of time to concentrate on the course. Students have stated that it makes their life easier even if they can get one course finished early and then deal with the rest. Automatic evaluation, the fact that all materials are accessible through the system and that there are no deadlines for in-semester submissions enables them to work extensively on the course for a short time period instead of spreading it over the whole semester.

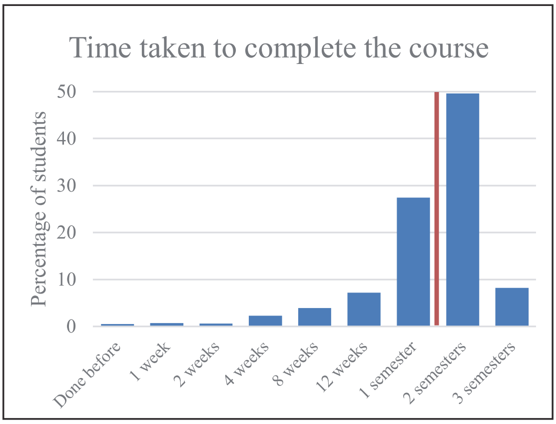


Fig. 2. Time taken to complete the course. Red line shows formal deadline.

Also, some students work on the course in the system, get the unofficial grade and then, officially, declare the course next semester (Fig. 2, column "Done before").

Results show that students are generally happy with so much control over their own grade and they love freedoms this system offers them. Students learning style analysis shows that this kind of approach seems to work better with certain group of students depending on their personality. Some students have stated that it is very hard for them to motivate themselves to work when there are no required deadlines burdening them during the semester. Those students tend to leave everything to the last minute and therefore have a hard time completing the course on time. They prefer to rush before deadlines and statistics have shown that the busiest time for on-campus class tests are weeks before official university end of the semester deadline. Statistics also show that those students who start working early, usually finish solidly before official deadline and on average have higher grades than their comrades.

Also, this approach for grading gives student's full control over their grade that also benefits personalized learning as everyone has their own speed and style of learning and they can set their own goals for the course.

V. Process Visualization

Before competence-based approach was used, there was no need for an elaborate visualization. Courses usually consisted of 10-15 topics. Every topic was graded with one number. Students were able to keep track on things they still needed to do by their selves.

All that changed with competence-based approach. It is vital, with described system for "grade suggestion," that student is able to understand his/her current progress in the course and therefore be able to plan his/her time accordingly. As the learning process is controlled by the student it is vital that the learner has adequate data to be able to make the best decisions.

System offers multiple views of the data for the student. Visualization for the grade has evolved with the system.

A. My Field

MyField (Fig. 3) consists of small color-coded current competence state indicator boxes that are used by the students in the learning process to choose competence they want to learn. So, the field also doubles as a workbench for learners, not being only visual tool to follow progress. It was the first progress following tool offered in the e-learning system when switching to competence-based approach even if process visualization problem was not acknowledged at that time.

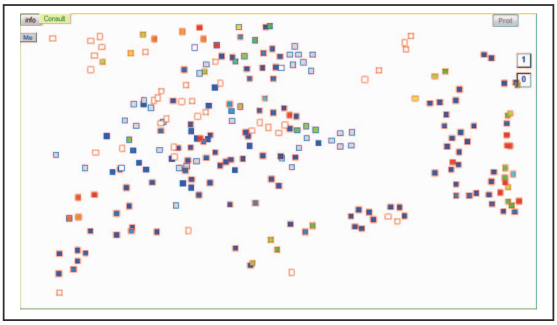


Fig. 3. MyField – screenshot of central workbench that students use to choose competences during learning process. There is 10-20% overlap in the field in order to personalize the learning.

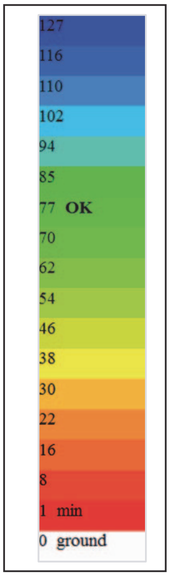


Fig. 4. Color-coding for system with 128 ability levels.

Student chooses a competence he/she wants to learn and indicates his/her wish by clicking on the corresponding box on the field. System then uses an algorithm to find the most suitable exercise with desired competence. Algorithm takes into account student's current state, time cap from the last time a desired competence was graded, forgetting parameters and previous learning history.

Color-coding is used to represent 128 different levels of competence acquisition (Fig. 4). Red and orange are used at lower level that instinctively make student feel like they are in "danger". When student reaches competence's acquisition level, green tones are used to indicate that current level is "safe". Blue is used to indicate higher levels after acquisition level is used.

Competences on the field are positioned so that similar competences would be close by and therefore areas of wider topics are formed. Thus, MyField also offers higher level views merging together similar competences to groups in order to facilitate students who have hard time switching from topic-based thinking to competence-based approach. Statistics have shown that using that higher level has decreased with years. One reason may be that new students who do not have previous experience with the system, may not notice that functionality at the start and get used with detail view of competences.

MyField also has a number representation of how much points student has gained by passing acquisition level and how many of them have been confirmed in class test. It also offers students' possibilities to sign up for class tests and on-campus labs. In addition, there is also a feature that enables to use the system in so called "protected"-mode where system tries to offer learner exercises where confirmed competences are not included. Those competences are visually represented with bold border.

Therefore, it can be said that for the students, Myfield is effectively the nerve center of the system. Statistics show that students spend 93% of time when logged into the system on solving exercises, originating from MyField.

B. Grade-suggestion motor

MyField may use color-codes to indicate student progress with competences, but it does not give visual representation of process in the course (how much is done, how much work is still needed). Grade suggestion function does that instead. Data from the function can be visualized so that the student could follow precisely how much work he/she has to do before desired grade becomes available as seen on Fig. 5. On x axis, credit units are shown. On y axis competence states that are equal or higher than acquisition level are represented. Possible grade areas on 2D are shown using function that takes into account the current university guideline according to which student should be able to pass after acquiring 51% of course material. To construct the lines that are used to plot different grade areas, this guideline and the point where student completes 100% of course with minimal level are used. Steps of 10 levels are used to define lines for higher grades. If student's results are above any of the visualized lines, it becomes the highest eligible grade.

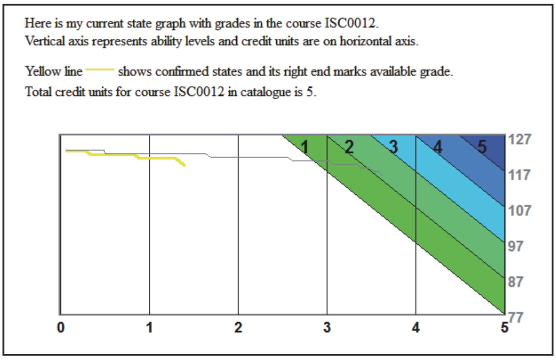


Fig. 5. Grade-suggestion motor visualization tool.

When yellow line reaches any of the grade areas, grade-suggestion motor is able to suggest that grade for a student.

C. Radar chart

Lately, it has also become popular to represent competence levels with radar style graphs [3 e.g.]. Thus, radar chart with all the competences required for the course with their current and acquired level has been offered (Fig. 6)

.

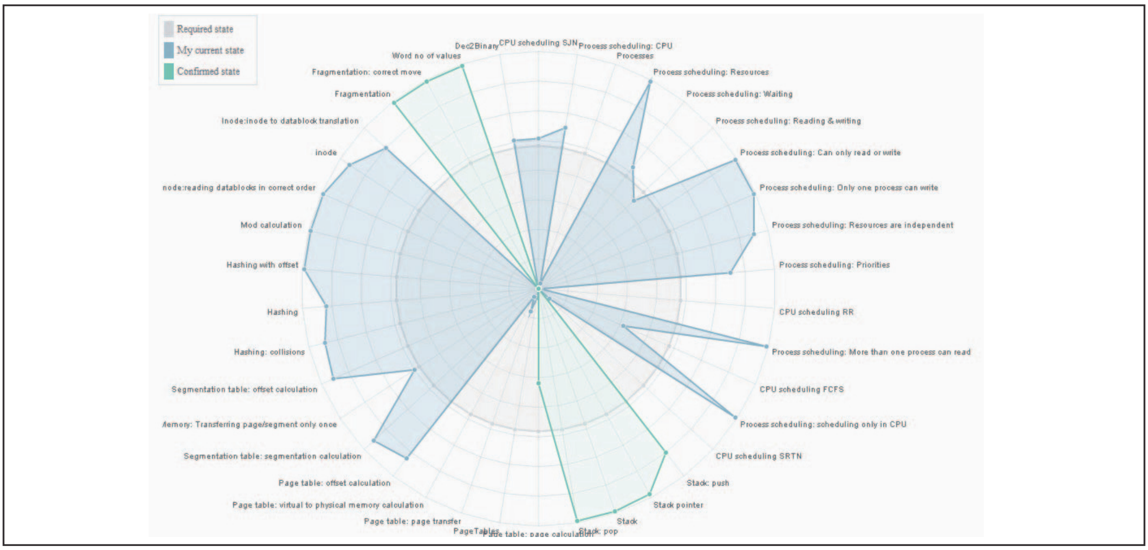


Fig. 6. Radar chart vizualisation tool is showing student's current level of confirmed and unconfirmed competences. Acquistion level is shown as a grey circle. Similar competences appearing frequently together are positioned in adjecant sectors using Karadar algorithm

Competence positioning on the radar can be thought as a line that is connected from the end to the start like a circle. Dimension of that kind of representation is 1d therefore information used to plot competences for MyField cannot be used straight forwardly. Transformation function from 2d plot area to 1d line is required.

Desired position for competences on the radar chart would be the following: it is expected that competence is surrounded from both right and left by two of the competences it appears most often together with in the exercises. Therefore, instead of using the same data that is used for MyField, new closeness methodology was derived called Karadar.

First, for Karadar algorithm, list of all competences for the course is collected. Then, for all the competences, from exercises where it is presented as input competence, other competences are counted. As a result, list of pairs of competences and how many times they appear together is attained. Then, using ideas of maximal spanning tree and travelling salesman methods, line according to similarities is formed of all the competences that will be used for the radar chart.

D. Interactive data viewer

Dashboard style interactive data viewer (Fig. 7) has also been offered. Similar visualization tool had previously offered for teachers to follow sole student's progress. For student view, some of the graphs were replaced or modified to hide system sensitive information (task id-s, correct answers etc). Also, some new graphs were added according to specific needs of the students.

Many different graphs are offered for student in interactive data viewer. Also, multiple type of graphs are used, trying to find most suitable visual representation for specific data. General layout remains the same, but as all the graphs are interlinked selection in one of the graphs results in redrawing all the others according to the filter. For example, a student can see the time he/she spends solving the task or which hours of day he/she works on a bar chart. Number of exercises done every day is offered as a timeline. Competences, student has solved, and if learning actions have been part of class test or not are shown as a pie chart. Also, data table with 30 rows is offered.

Interactive data viewer was designed using d3 javascript library [12] and data driven approach for documents. Different graphs were linked together using crossfilter javascript library [13].

This tool enables student to follow his/her overall process or use the interactive tools to limit the data in order to draw more specific conclusions.

The interactivity of data viewer gives students free reign to "play" with the data. At first students may not realize the benefits of using filters to limit the information they are looking at, but it has been observed that more they use this tool, the more they start to comprehend.

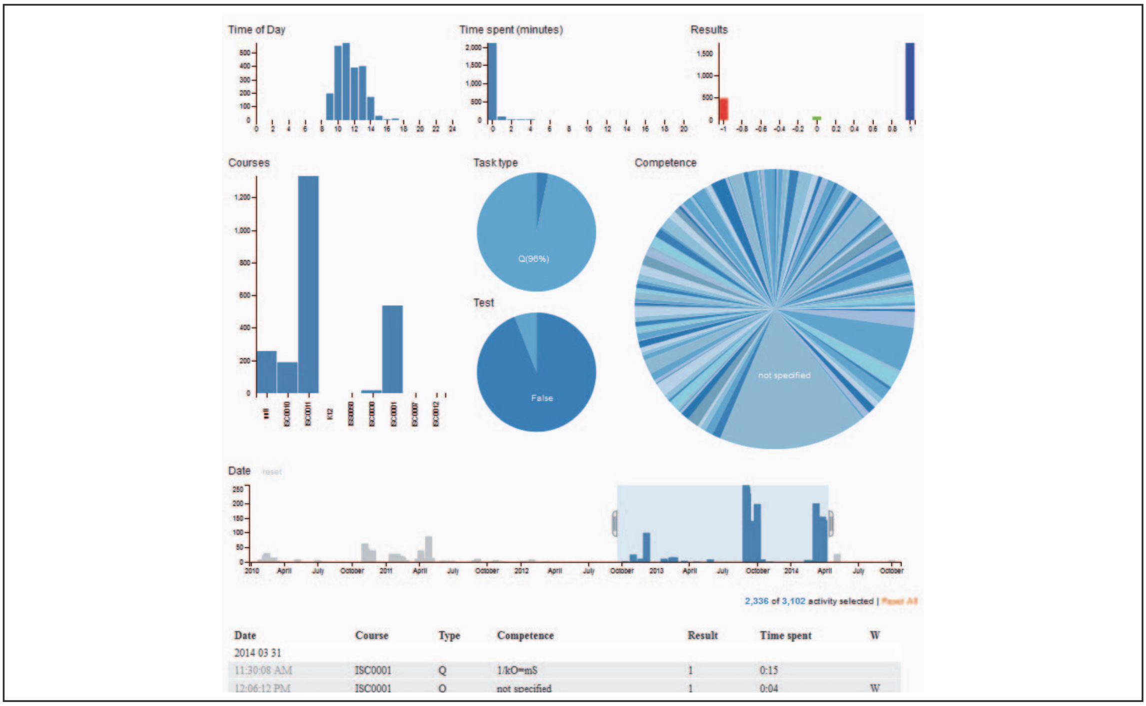


Fig. 7. Interactive data-viewer with multiply interlinked graphs enabling students selection to redraw all the other view.

E. History map

Taking into account the initial feedback from the launch of visualization tools, another view was designed. History map (Fig. 8) has been designed to enable students to playback their competence state changes over the last four months. They can select any course they have declared and their original state from four months ago is shown. They are given information on how many competence changes have been recorded on that timeframe. In case, they have not been active on the course for four months, the timeline is shown from their first action and therefore also information about the resolution of the timeline is given.

History map loads student's state in the course at the start of the timeframe. Chosen course MyField layout is used as a base for the view. Then student has the option to start and pause the playback. Playback consists of changing competence box colors according to the state changes. Student can hover on the colorful boxes representing competences to see what elementary skill is behind specific box.

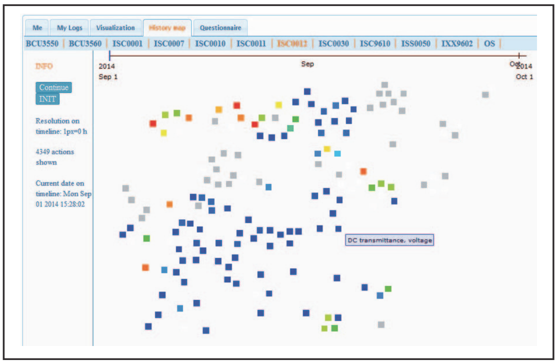


Fig. 8. History map that enables students to replay their actions over last 4 months for selected course.

Four months for the default timeframe was chosen as it equals one semester. If all the data would have shown, for some students it would have meant that ten thousands of actions would have been displayed. It would make the playback too long, and it was thought that earlier data might not be relevant anymore for learners.

VI. Results

Questionnaires in both Estonian and English were added to the e-learning environment with newly introduced features. Students had time to try the features and were encouraged to fill out the form. Data from the system following learners path and behavior (time spending doing tasks, time spent reading materials, time spent using visual tools, how fast student made process, average rate of answering correct etc) through the course and final result was used to verify their statements. Feedback originating from the questionnaire and from verbal conversations with students was used to draw following conclusions.

MyField is a tool that students have used for learning since switching to competence based learning. Students have said that they familiarize themselves with Myfield quickly at the start of the course. They state that they can check their process between exercises without having to click or go some other part of the system. Also, some have thought that current process might not be so easily grasped with MyField.

Simplicity, familiarity and a constant automatic access as a side product for choosing a new task seems to be the major benefit for students. Alas even if more detailed and specific views are offered, some students still prefer to stick with the old and familiar concepts.

Those students who chose MyField as their preferred tool for process following, had more likely completed 2 or more courses already in the system. They had usually worked more than average hours in the system and had been in university for few years already.

Interactive dashboard catches students with its interactivity. It has been stated by the students that they first start just to play with different views to familiarize themselves with the environment. When they have understood how the views are offered and linked with each other, they start to look for more specific information. This view enables them to check their results during class tests – they can see what competences they got correct and which resulted in mistakes. According to students it is a major benefit for them as they were not able to access class test results before. No results/feedback is generated for the students during class test as it might discourage and disturb them.

Also, this view offers them access to additional data they had no previous access to – for example, detailed personal logs of their activities. In addition, it enables them to get statistic what competence they are most likely to answer wrongly. What competence they have solved most often, which one the least. It opens up many possibilities and gives students a chance to analyse and modify their learning behaviors accordingly. This is the new visualization tool that students have given most positive feedback on and stated that it is highly useful for them.

Students who preferred interactive dashboard were mostly taking their first course in the system. They were more likely to finish the course on time with high grade. Also, interactive dashboard was overwhelmingly most popular choice by those students who filled out the form in English. Majority of those students finished the course before rest of their peers.

On the other hand, some students have stated that interactive dashboard is hard to understand. As this tool expects student to take active lead on what he/she wants to know, some are modest to do so. Those students have said that it is hard for them to understand what those different views offer even if they have titles. Also, the crossfilter functionality seems to confuse them.

That group consisted mostly of first year students but they stood out that they spent very little time reading materials provided by teachers in the system, their rate of getting correct answer was below average (that might be the side-effect of not familiarizing themselves with material) and they tended to leave using the system and making process in the course to the end of semester.

Radar chart assembles all the competences in the course together in the same chart to give a simple, fast visual view how much work is done and which of the competences have been confirmed. Even though radar chart seems to accomplish this goal, the view does not take into account that different competences have different weights for the courses. Students have pointed out that in general the chart gives them overview but it does not make the planning accurate because of the competence's weights. It has been suggested that this factor should also be represented somehow on the chart.

The group of students who preferred this view was the most varying consisting both new and old learners without any specific characteristic in their behavior standing out. Biggest fluctuation in this group was with final grades. Almost half of them were achieving high grades. On the other hand, around 30% of those students had trouble finishing the course.

Grade suggestion motor has been praised by students for showing progress on the course as a line that makes it really easy to grasp and leaves no room for misunderstandings. Also, representing grades as areas on 2D plot has been well received and this chart offers them easy way to see where they are in the course and how far are their goals. Also, they have stated that grey little line giving the result if all current competences have be confirmed has been stated to have positive motivating force. They are more eager to enroll for on-site tests. On the other hand, the problem that competence weights are not easily visible on this type of chart have been pointed out. Also, the fact that there is nothing interactive, seems to bother students even thought when asked what they would like to see they are not able to suggest anything specific.

This view was mostly preferred by students who have used the system for years and majority of graduate students tended to favor this approach. This group stood out that even thought they had used visual tools in the system a lot, they had clicked mostly (around 82% of visual tools visits) on grade suggestion motor .

History map was launched later in the system than other visualization tools therefore the amount of feedback for that tool has been smaller than for others. The feedback shows that students enjoy replaying their process – sometimes only to see the progress they had made with weeks of learning. They have brought out that this tool motivates to work harder as they see, like a movie, where they have come from and how far in the course they have made. On the other hand, students have stated that this tool has entertaining value, but they prefer not to follow their process only with this tool. It does give them overview but they state that tool lacks details they can access with other tools. Students have also said that it is easy to understand as it uses the same "MyField" layout that is very familiar to them.

Students who filled out the feedback form stated that radar chart was the tool easiest to grasp and understand. Grade suggestion motor was brought out as a tool that had most of the information that they required and myField was voted to be the tool that has helped most to follow the process. When asked for their favorite tool, myField and interactive dashboard were two options chosen with equal share. Also, majority of the students stated that they used visual aids only few times. Only around 20% of answers stated that they used tools at least once a week or more frequently.

Apriori frequency set algorithm was applied to questionnaire data. 78% from those who answered stated that those tools have helped them to grasp their process better than before. Those who said that they used visualization tools frequently were also the ones who stated that it was very useful for them. The main benefit brought out by the students was access to the data. That was mentioned very frequently by older students who had experience with e-learning system with many courses before visual tools.

On the other hand, the students who said visual tools had not helped them, had different favorite tools and also their frequency of using those tools varied to almost never to every day. That group seemed to show two distinct behaviors. First group of students had spent more than average hours using the e-learning system and number of tasks they had solved was very high. Their average result varied. Some of the students had very inconsistent pattern of answering correctly. Sometimes they got totally right answer, then next one was totally wrong. Other students hardly ever got totally right answer but they never got totally wrong answer either. Their result was constantly around 0 (if -1 is totally wrong and 1 is totally right).

The other distinct group was learners who had spent minimal hours in the system. They almost lacked time spent using help materials and the number of tasks they had competed at the time of filling the feedback was well below their peers who started at the same semester. As expected, they had not advanced enough to attend class tests and majority of them had no confirmed competences. They had used visual tools around 10 minutes when they submitted their feedback form.

Around 32% of those who said that visual tools did not help them had no trouble of finishing the course – it took them longer than average with few "early-birds" in the mix.

Students, who voted MyField to be the best visual tool, also gave it the best scores. On the other hand, students who liked interactive dashboard best did not score it higher than all the other tools. As expected, the opinions of students vary a lot as everyone as their own personal learning style.

Overall, the feedback has shown that students appraise visual tools (3/4 said it helped them), but it took them some time to get used to using the new features (first year students taking their first course were more eager to try new features than those who had already used the system before – behavior we have observed before when releasing new features to the e-learning system). Students who had worked hours in the system tended to prefer older features, for example prefer MyField over interactive dashboard.

In conclusion, analysis shows that when student passes the course, their understanding of learned competences is firm and memory model determines that they do not forget what they have learned as easily when using classical “topic-based” learning. Therefore, it can be concluded that even though described system may not be a perfect fit for everyone, the firmer knowledge, possibilities to personalize learning, motivation to achieve better grades and clearer picture for both students and teacher about learners’ current state are highly valued and make this kind of methodology beneficial.

REFERENCES

[1] I. Baumgartner and V. Shankararaman, "Actively linking learning outcomes and competencies to course design and delivery: experiences from an undergraduate Information Systems program in Singapore," Proceedings of Global Engineering Education Conference EDUCON-2013 (Mar 13-15, 2013, Berlin, Germany), IEEE, 2013.

[2] M. Berges et al. "Developing a competency model for teaching computer science in schools," Proceedings of the 18th ACM conference on Innovation and technology in computer science education (Jul 1-3, 2013, Canterbury, United Kingdom), ACM, 2013.

[3] D. Zehetmeier, M. Kuhrmann, A. Bottcher, K. Schlierkamp, and V. Thurner. "Self-Assessment of Freshmen Students’ Base Competencies," Proceedings of Global Engineering Education Conference EDUCON-2014 (Apr 3-5, 2014, Istanbul, Turkey), IEEE, 2014.

[4] H. Rozeboom, “Competence based learning,” Bulletin of University of Agricultural Sciences and Veterinary Medicine Cluj-Napoca. Horticulture, vol 63, issue 1-2, 2006, pp. 365-367.

[5] A.V. Sánchez and M. P. Ruiz, “Competence based learning - A proposal for the assessment of generic competences,” University of Deusto, Bilbao, 2008, pp. 29.

[6] M. Jaanus, “The Interactive Learning Environment for Mobile Laboratories,” Tallinn Univeristy of Technology Press, Tallinn, 2011, ch.4, sec. 3, pp 59.

[7] K. Umbleja, V. Kukk, M. Jaanus and A. Udal, "New concepts of automatic answer evaluation in competence based learning," Proceedings of Global Engineering Education Conference EDUCON-2014 (Apr 3-5, 2014, Istanbul, Turkey), IEEE, 2014.

[8] V. Kukk and K. Umbleja. "Analysis of forgetting in a learning environment," Proceedings of Electronics Conference BEC 2012 (Oct. 3-5, 2012, Tallinn, Estonia), IEEE, 2012

[9] C.M. Chen, “Intelligent web-based learning system with personalized learning path guidance,” Computers & Education, vol 51, issue 2, 2008 pp. 787-814.

[10] M. Jaanus, V.Kukk, K.Umbleja, B.Gordon and M.Pikkov, “HomeLabKits -implementation and usage”, Proceedings of 13th biennial Baltic Electronics Conference BEC-2012 (Oct 3-5, 2012, Tallinn, Estonia), IEEE, 2012.

[11] M. Forehand, “Bloom’s taxonomy” in Emerging perspectives on learning, teaching, and technology, 2010, pp.41-47.

[12] M. Bostock, “Data-Driven Documents” [Online]. Available: http://d3js.org/

[13] “Crossfilter” [Online]. Available: http://square.github.io/crossfilter

Publication 6

Reference

K. Umbleja and M. Ichino, "Predicting Students' Behavior During an E-Learning Course Using Data Mining," in Interactive Collaborative Learning: Proceedings of the 19th ICL Conference - Volume 2, Cham: Springer International Publishing, 2017, pp. 175-189.

Abstract

This paper introduces a process of building a prediction model for student's final grade and time of finishing, based on students' previous behavior. Prediction model was developed using data mining with regression analysis, principle component analysis and hierarchical clustering of symbolic histogram valued data. 35 different features of students' activates was considered but only the 9 most important, so called principle components, were used in the model. Then, using histogram valued data - a type of symbolic data that allows learning processes to be described in a more natural form, and a hierarchical clustering, previous students' behaviors were grouped. For an accurate prediction, a closest cluster to student's current progress was found. To verify the model's correctness, predictions were tested on a largest course in e-learning system in 2015 fall semester. The model was found to work sufficiently.

Predicting Students' Behavior During an E-Learning Course Using Data Mining

Kadri Umbleja[1](✉) and Manabu Ichino[2]

[1] Tallinn University of Technology, Tallinn, Estonia
kadri.umbleja@ttu.ee
[2] Tokyo Denki University, Saitama, Japan

Abstract. This paper introduces a process of building a prediction model for student's final grade and time of finishing, based on students' previous behavior. Prediction model was developed using data mining with regression analysis, principle component analysis and hierarchical clustering of symbolic histogram valued data. 35 different features of students' activates was considered but only the 9 most important, so called principle components, were used in the model. Then, using histogram valued data - a type of symbolic data that allows learning processes to be described in a more natural form, and a hierarchical clustering, previous students' behaviors were grouped. For an accurate prediction, a closest cluster to student's current progress was found. To verify the model's correctness, predictions were tested on a largest course in e-learning system in 2015 fall semester. The model was found to work sufficiently.

Keywords: Educational data mining · Study behavior · Symbolic data analysis · E-learning · Self-regulated learning

1 Introduction

E-learning, like any other field, is in a constant process of innovation and improvement. All changes have side-effects that cannot be foreseen or predicted beforehand, as they only become visible when those new innovations and improvements have been applied. Switching from classical topic-based learning to competence-based or competence-driven approach, which has become more and more widely used, the problem with self-regulating has arisen.

2 Competence-Based Approach

Development of ISC e-learning environment started at the end of last century [1]. At first, the system only supported classroom learning as a place to store materials and offer information. After a few years of development in web technologies, ISC became a central hub for the learning. The classroom learning supported independent, self-regulated learning that students were conducting in the system. To overcome the students' attitude that they are interested in only passing the course with minimal effort, classical topic based learning was replaced with novel competence-based approach.

M.E. Auer et al. (eds.), *Interactive Collaborative Learning*, Advances in Intelligent Systems and Computing 545, DOI 10.1007/978-3-319-50340-0_14

Competence-based or competence-driven learning has been widely adopted lately by different researchers and institutions [2–6]. They all have a similar framework and focus but they do not have a common definition of a term competence. For example, Klarus has defined competence as a mix of skills, attitudes and knowledge that makes the employee or graduate successful in society in his/her profession [7]. Other definition states that by competence, we understand good performance in diverse, authentic contexts based on the integration and activation of knowledge, rules and standards, techniques, procedures, abilities and skills, attitudes and values [8]. For current work, the competence-based learning is defined as a knowledge based methodology which concentrates on measuring what a person can actually do as a result of learning [9].

Competence-based learning contrasts classical topic-based learning. For example, a topic-based learning is focused on certain topics (e.g. Ohm's Law as part of Physics course) and then it is checked if a student, in general, can solve exercises or knows the theory relevant to that topic. In competence-based learning, those classical topics have been divided into smaller units, competences. In current study, atomic portions of theory (elementary skills of knowledge, atoms that cannot be divided into smaller units) from previously used wider topics have been extracted.

To clarify that approach, the following example can be used. In classical topic-based approach, if simple mathematical equation $\mathrm{x} = (4 \times 8 + 10)/2$ is analyzed, it is only checked if the x is correct. Intermediate steps like addition, multiplication and division used to getting the final answer are ignored. In competence-based approach, student's skills of addition, multiplication and division are also analyzed and graded.

This kind of approach requires more complex grading. Instead of summating or averaging the grade over a wider topic, each and every competence presented inside an exercise should be considered and processed/graded separately. That kind of methodology raises many challenges for grading exercises but gives students proper representation of their current state in the course. Currently, grading is achieved using complex algorithms that try to mimic student answering process [10].

In competence-based learning environment, it is recommended that exercises would be divided into smaller units that concentrate only on a few competences simultaneously. Repetition is a vital concept for competence-based approach - in order to fortify the knowledge it has to move from short term memory to long term memory. Human forgetting has been studied thoroughly and H. Ebbinghaus was the first to show with his experiments that retention is very fast at the beginning, but quite slow later, thus repeating is important to memorization [11, 12].

In this kind of approach, courses do not contain specific topics but rather consist of a list of competences that should be acquired by the student during the course. Every competence in a course has a weight connecting to it, illustrating its prominence in that specific course. In order to personalize the learning [13], courses usually have 10–20% overlap when summing up competences' weights. It does not force students to acquire every one of the competences listed in course but to make a choice according to his/her personal needs and interests. The weights are chosen so that it is not possible to complete the course without acquiring the main topics and fundamental ideas from the courses.

3 Motivation

Using classical topic-based approach, it is relatively easy for a student to keep track on his/her progress. There are only a certain number of tests/tasks that have to be completed once in order to finish the course. Even with self-regulated learning, students were quite adequate at planning their own time and finishing before the deadline.

That kind of straight forward approach is not possible in competence-based learning. Competence-based approach requires repeating similar tasks over and over again to make sure the knowledge moves from short term memory to long term memory. Therefore the time is not linearly predictable. Some students work very fast, others need much more time. It also depends on time between solving similar tasks, on forgetting model parameters, on how correctly student answers, on how often certain mistakes are made etc. Data has shown that the amount of work done by the students to complete the course may vary even as widely as 6–7 times (only time spent in e-learning system can be measured). The second problem is that when students are masters of their own time and the only deadline is at the end of the semester, they tend to leave everything to the last minute. As the system also embeds forgetting model, rushing at the end usually does not lead to pass.

It became obvious that when self-regulated learning worked sufficiently in classical topic-based approach, it became a source of many problems in the competence-based learning. Students became unable to predict how much time they require to complete the course. For example, students hardly showed any activity in the middle of the course and rushed weeks leading to the final deadline. This led to a rise in students failing the courses and to students who thought they got a lower grade than they deserved.

Throughout the years, different approaches have been used to tackle the problem of "late-wakers" and prediction of time needed to complete course. For example, different tools have been developed [10]. Despite initial positive feedback from the students about those tools, the data shows that students have not been very active in using them. Also, those approaches have not yet delivered any significant change in students' behaviour.

The proposed solution to those two problems is to predict students' final result throughout the semester using all the data that has been collected by the system from previous students.

4 Methodology

Data mining will be used to develop previous students' behavior models. It is a large field consisting of different disciplines from machine learning and statistics to database technologies and visualization. There is a specific sub-field called educational data mining that is concerned with developing, researching, and applying computerized methods to detect patterns in large collections of educational data – patterns that would be otherwise hard or impossible to analyze due to the enormous volume of data they exist within [14]. Data mining has been used in many different educational research projects [15–18].

Regression analysis is a statistical technique for investigating and modelling relationships between variables [19]. It is widely used in prediction and in forecasting [15, 20, 21]. Regression analysis allows us to estimate how set of variable is connected to the unknown variable. In current study, we are interested in how known student behavior (set of known variables) will result in finishing time and final grade (unknown variables).

As e-learning system logs all actions that a learner does in the system, there will be a lot of information (variables). Instead of using all of the possible information collected in the system, a set of linearly uncorrelated variables, so called principle components, can be found to describe the relationships without any significant loss of details. Therefore, instead of using all possible variables from the system, a set of most important variables could be found.

Principal component analysis (PCA) uses a vector space transformation to reduce the dimensionality of large data sets [22]. It is a statistical procedure using an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables [23]. This transformation is defined in such a way that the first principal component has the largest possible variance and each succeeding component has the highest variance possible under the constraint that it is orthogonal to the preceding components [23].

Furthermore, variables are usually given by a single value (for example: number "5", modal variable "woman"). In learning, it is not so important how student solved a specific task in a specific moment rather than how he or she is progressing through the course. So a single value for learning or action in the e-learning system may not be the best way how to present the data.

Symbolic data can be used to compress big data into analyzable size or present data that is more natural in symbolic form. Symbolic data values are defined as hypercubes in p-dimensional space opposite to classical data values that are defined as points in p-dimensional space [24]. Symbolic data comes in more complex forms than classical data – it does not only containing values or categories but also has internal structure and variation [25].

Good example of data that is more natural in symbolic form is weight as it is constantly changing. Therefore, instead of using single value to give persons weight, it would be more natural to give is as an interval. Interval valued data [25] can be derived even more precisely when probabilities are added to the intervals – that kind of symbolic data is called histogram data. For current model histogram valued data is used.

For grouping similar students' behaviors hierarchical clustering algorithm was used based on Cartesian system model [26]. The generalized Minkowski metrics was modified to work with histogram data [27]. Cluster compactness was used as a criteria of merging clusters. Cluster compactness was defined as the arithmetic mean of distances between cluster average observation and every other observation used to form the cluster. Defined cluster compactness is also a metric. This kind of approach guarantees most compact clusters at the end of the clustering. The drawback is that, instead of common hierarchical clustering, this method is algorithmically more complex.

Clustering algorithm is used to find groups among previous students. Current students' histograms are compared with those groups using similar metric as in clustering. A group with whom the current histogram has lowest dissimilarity is taken as a base for describing students' predicted process.

5 Implementation

First step in building the prediction model is to determine the set of variables that are principle components.

At first, results for every competence during a time period of a week were considered. There are currently more than 300 competences connected to electrical engineering and computer science in the system and PCA was applied to them using R software [28]. The regression model was developed based on those principle competences but it became evident that using competences as variables did not lead to a good model. Competences only reflected one side of student activity in the system – solving tasks.

Therefore, wider range of information was considered and different kind of variables were deducted from the data. In the end, 35 variables were considered. After PCA, 9 most important indicators were found. Those included number of tasks done, active days, correctness of submissions, difficulty level of submissions, number of times help material was accessed, time of the day of submissions, current progress on the course, times of HomeLabKit (portable kit with everything needed to do basic electrical laboratory experiments outside laboratory) borrowed and times of class test done. The results of PCA are plausible as they cover all important factors of the system, not only the task submissions.

All previous study records were analyzed and histograms for all the students calculated. Week was used as a time period for one symbolic object. Every semester consists of 16 weeks of teaching and additional 3 weeks for exam session. Then, another week serves as an extra time for those who did not manage to finish on time. Therefore, for every student 20 weeks were analyzed.

The e-learning system contains information from more than 15 years. Students' study behaviors and course context have changed a lot during those years. Therefore, at first implementation, only records since competence based approach was introduced in 2010 fall semester, were considered. The analysis showed that throughout the years learning patterns have changed.

Despite one course considered had been taught during those 5 years in similar manner, students' behavior during the course has changed a lot. For example, at 2010 they worked more and frequently. Getting a good grade was the main goal. Lab experiments were done in regular intervals and learners borrowed HomeLabKits from the start of the semester once in two weeks. In 2014 fall semester, at the same course, students waited around half semester before majority of them started to show any progress in the course. Students accept low grades as soon as they become available. Lab experiments were avoided as often as possible, being the last things done in the course. Also, borrowing kits has become very irregular and less frequent.

Similar changes were found in other courses. Therefore, for building the analysis, only last 2 years (4 semesters) were considered. It would make sure that significant pattern changes in students' attitudes do not affect the model too much. The downside of that constraint is, that every year, the models have to be re-evaluated meaning extra work.

Therefore, for implementing the model, 5 courses with largest number of participants in the last 2 years containing 1100 students in total were considered. Those courses were Circuit theory, Basic Measurements, Microprocessors and two courses on

Operating Systems. For all the students on those courses, histograms over the 20 weeks of their studies were found for all 9 most important variables. Then, for all the students, clusters were found using hierarchical clustering algorithm. Students were clustered based on the course and all together.

Clusters were then analyzed using dendrograms. Dendrograms are graphical representations for the arrangement of the clusters produced by hierarchical clustering [29].

To extract meaningful groups or clusters from dendrogram, it has to be cut. Conditions for cutting were the shape of the dendrogram and the properties of the would-be-clusters. If the properties were too specific, merging it with another cluster was considered. If the properties were too wide, cluster was cut from the furthest connection point and both of the cut clusters were considered further. This continued until groups with well-defined properties had been achieved and cutting any further would just fragmentize the result.

Clustering students based on the course worked very well – clear clusters with different characteristics could be detected. Clustering all the students together did not yield satisfactory results despite high hopes that there could be a possibility of developing a general model for all students independent of the courses. Apparently courses have a distinct characteristics that affect the patterns. For example, Operating System courses do not use HomeLabKit. Therefore pattern for borrowing kits in that course is constant zero.

Currently, models were developed for 5 largest courses in the system. Courses had 7 to 14 clusters. As an overall rule, the clusters contained at least 20 students. There were a few exceptions with clusters with clear (very different) characteristics that had less members. Despite courses having different learning volumes and credit points, there were groups with specific characteristics which were common for all courses:

- “Early rushers” – students who started working at the beginning of the semester and finished within the first month. They got very good marks (4–5, in scale 0 to 5 where 0 is fail and 5 is the best mark).
- “Late rushers” – do not work at all during the semester. Start studying during the 16th week or during exam session. Do a lot of submissions in the small timeframe. Their submission times are well below the average (they hardly have time to read the task). Some of them manage to finish with lowest grade, some of them not.
- “Constant workers” – they work regularly with low number of submissions during the whole semester
- “Drop outs” – do not work at all or very little. Do not pass.
- “Mid-semester sleepers” – Work little at the start of the semester. Then activity slows down and they start working again on last few weeks of the semester and during exam session. They pass the course with varying grades.
- “Second half-ers” – They start working after the mid-terms. Do not rush and have average number of submissions. Usually finish on time and with grade 3.

Also, all courses had a common characteristic that in last two years around 20–30% of students did not finish the course with a positive grade.

For every cluster in every course, a descriptive observation was deducted from the clustering step, meaning that every cluster had “average” or “common” histogram for all 9 variables for 20 weeks. Therefore, for every course a model was achieved containing descriptive observations about each cluster in the course.

6 Prediction

To verify the correctness of the achieved model, during a fall semester of 2015 the course with 110 officially declared students studying Basic Measurements was used as a test group. Basic Measurement group was the biggest one containing 14 clusters, as can be seen from Fig. 1, and had around 450 students who had studied the course in previous two years.

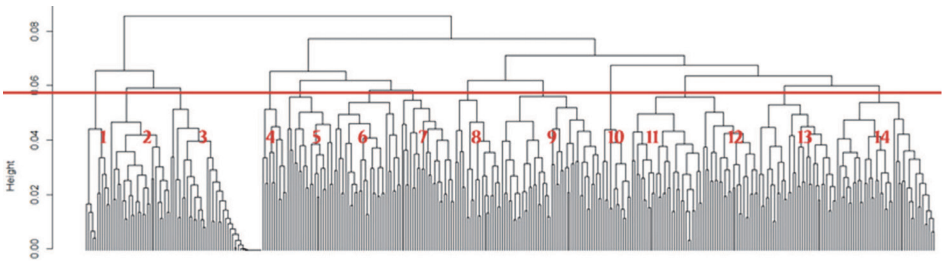


Fig. 1. Dendrogram for students of previous two years with 14 clusters in course Basic Measurements. Red line shows cutting point of clusters

During previous two years, some organizational changes have taken place. For example, the course was taught mostly for second and third year bachelor students in Informatics department (containing computer science, electronics and communication curriculums). From 2014 fall, computer science students are not required to take this course. Also, from that time onwards, the course was moved to first semester.

Cluster 1 contained students who have completed most of the work in previous semester before officially declaring the course or those students who do not work at all. This group is characterized by almost non-existing activity and very bad results for submissions. Students in this group usually fail or, if they have done work in previous semester, pass with grade 3.

Cluster 2 contains "late-rushers" who usually start working at the end of semester, from week 14 onwards. They have very fast submission times and do a lot of submissions on the short time period. They tend to have bad results and in the end, during exam session, they start using help materials as they realize they need some kind of theory. Majority of them pass with lowest grade.

Cluster 3, so called "drop outs", contains students who work very little, mostly on week 14 and 15. They try the course, get bad results for their submission and they give up. Students in this group do not pass the course.

Cluster 4, has students who try the course at the beginning of the semester, do not work at all during the semester and then try to complete it during exam session. Around half of them manage to finish with low grades, another half fail. They stand out by having good results for submissions – so they know what they are doing but they just leave everything to the last minute. They are so called “lazy but smart”.

Cluster 5, “late-wakers” tend to work from week 14 onwards and usually continue up to the additional extra exam week. They have fast submission times that reflects that they are not spending enough time with tasks. Therefore, they also have bad results. Usually, students in this group do not pass on right semester as they just run out of time.

Cluster 6, is quite similar to cluster 5. Those students also start working from week 13–14 and continue to exam session. They differ from previous cluster that they are able to finish within three weeks during exams. They also have a lot of submissions and mostly bad results but they have more reasonable submission times meaning that they actually have time to read and think what they have been asked. Surprisingly, they mostly finish with grade 3 or even with 4.

Students in cluster 7 are so called “early-rushers”. They start working at the beginning of the course and usually finish by week 10. They have high number of submissions and they get very good results. They tend to use lot of help materials, participate in class tests and borrow home kits. Despite having large number of submissions, their solving times are quite slow meaning that they spend a lot of time thinking and calculating while solving the exercises that probably contributes to their high results. Grades in this group are varying from 3 up to 5.

Cluster 8 work mostly from 12 weeks forward. They have good results but their solving times are from average to slow. They do average number of submissions. Usually finish with grade 3. They also have quite high help material usage and class tests due to their working pattern – work is only done at the second half of the semester and during exam session.

Cluster 9, “constant workers”, start working from the start of the course and they usually finish from week 12 onwards. Students in this group use help materials more than other students. They have average to high submission numbers, average solving times and average results. They tend to start doing class tests at the second half of the semester that does not allow them to finish earlier than mid-semester. They finish with 3 s.

Cluster 10 is similar to cluster 7. They also have very good results and they know what they are doing. They manage to complete the course with lowest number of submissions that are submitted very fast. They are so called “geniuses”. They almost never use help materials. Despite good knowledge they do not care to work for highest grades. Usually they finish with 4.

Cluster 11 contains “second half-ers”. They have almost no activity at the first 8-9 weeks of the semester. Then, when the mid-terms are over, they tend to remember they have registered in the course and start working. They have high number of submissions with very good results. Their results for laboratory experiments are much worse than theoretical tasks and they do not like using HomeLabKits. So they avoid lab tasks and finish with 3 or 4.

Cluster 12 is so called “mid-semester sleepers”. They start working at the first half of the course, then they stop and start again from week 14 up to the end of exam session. They do a lot of submissions on those peak working times but they tend to have bad results. They are especially slow when solving laboratory experiments. Usually they pass with grade 3. Few in this group finish with grade 2.

Cluster 13 is similar to cluster 11. They also start working at the middle of the semester, usually earlier than students in cluster 11. They have bad results and they work very slowly with laboratory experiments. They are uneven learners with few weeks they have more submissions and then some weeks they have significantly less. The unevenness is not week-based but active weeks differ from student to student.

Students in cluster 14 are similar to students in cluster 11 but they work more slowly during the whole course. They tend to have average to low submission times. They do not rush and have almost constant number of tasks per week during most of the semester. It is surprising that they usually get good results for theoretical tasks but lab tasks are hard for them and they get bad results for those. Usually pass with 3 or 4.

Using descriptive histograms from those 14 clusters, students' progress during the course was predicted. The first prediction based on the model was done after week 4. It was assumed that before week 4, there is very little activity and very few histograms can be generated for the students. For predicting student progress, first, 9 variables for four weeks were calculated. Then, using same kind of dissimilarity measure as in hierarchical clustering, a cluster, from those 14 described, with smallest dissimilarity was found. That cluster was considered to be student's predicted pattern.

Afterwards, this process was then repeated at the end of every week to keep the prediction up to date. The prediction is done up to the point when student accepts the grade that is varying. The week when students finished can be seen in Table 1.

Table 1. Number of students who finished on specific week. The course is 16 weeks long and students should finish by the end of the teaching period. They do have an extra 3 weeks to get a grade during exam session. Additionally, there is extra week between semesters where grade acquisition is also possible (so called extra exam week).

Week	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Nr of students	1	0	1	0	2	7	2	0	9	2	8	4	4	8	12	12	0

7 Analysis

At the end of the semester, 72 students (65.45%) passed the course. This is slightly less than has been average for last two years. It could be explained that this year no student managed to complete the course on week 20 that has always been quite popular among "late-rushers". The other possible explanation could be that the course is taught on the first semester to freshmen.

As a result in 89 cases (80.91%) the final grade was predicted correctly using developed prediction model. In 17 cases (15.45%), the actual final grade was one step higher than the predicted one. This occurrence was further analyzed and it was concluded that during semester in question, it was the change in students' study behavior compared to previous students – they continued to work further to achieve better grades. In some of those cases the grade prediction was spot on until week 17. All those students got grade during exam session.

In 4 cases (3.63%), student was predicted to pass with 3 but in reality, the student failed. All those 4 students belonged to cluster 11. This case was also further analyzed and it looks like again the prediction goes off at week 17 (start of exam session) – before that (week 16 and before) students were clustered into cluster 3 that has characteristics of having almost no activity at all. This case needs further analysis as those last few weeks should not have had such a huge impact on clustering as all features are normalized.

The change in clusters during the course can be seen in Fig. 2. The changes in groups are quite frequent at the beginning of the prediction process. Further into the semester, especially at the end, the changes become less frequent, almost stopping. It was expected that the changes stop earlier during the semester. On the other hand, most of the changes occurring are with adjacent clusters and majority of the time, the final grade prediction remains the same despite cluster change.

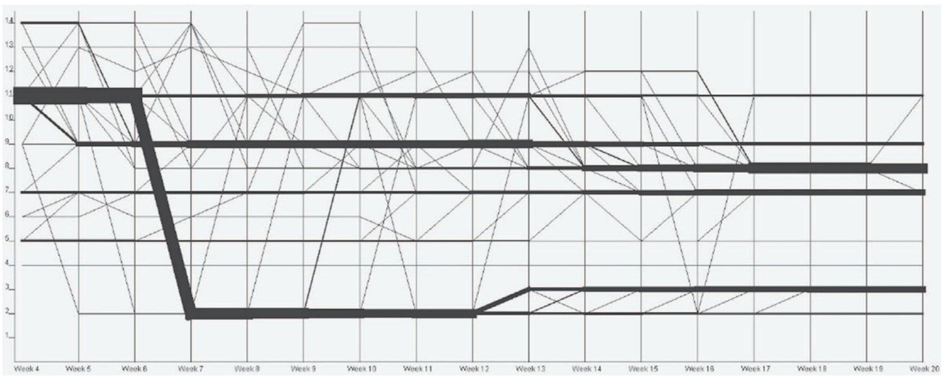


Fig. 2. Cluster prediction and its changes for students in Basic Measurement course from week 4 till the extra exam week. Bolder the line, more students are in the cluster that week

Major changes are happening after week 7. Students, who were predicted to be "mid-semester sleepers", those who try little-bit at the first half of the semester and then start again after midterms, are expected to have shown some kind of activity but they remain inactive and are therefore clustered to cluster 2 – "late rushers". Their final grade prediction actually remains the same – what changes is the pattern that is expected of them.

From week 7 onwards, it can be followed, that as students start to work, they are moved away from cluster 2, mainly to cluster 11. This trend actually continued till the exam session.

Another major change happens after week 12 when "late-rushers" who do a lot of submissions in a short time are expected to start working. Those who remain inactive are moved to cluster 3, "drop-outs". It is problematic that "drop-outs" cannot be predicted earlier during the semester but that is due to the characteristics of students' study behavior. Some "later-wakers" manage to finish as they work a lot in the end. Others do not work enough. Therefore the difference between those two groups become visible only at the end of the semester making it impossible to distinguish them before.

Changes between week 16 and week 17 draw attention. When changes slow down to week 16, they are more frequent again at week 17. Also, it is the point where for 4 cases the prediction becomes "incorrect" – prediction for students who have been evaluated to fail are updated to cluster with pass. This gap between the end of studies and the start of exam session requires further analysis.

Also, it was observed that at the start, the predictions are usually more positive (when considering the final grade) than the predictions at the end of the semester.

Furthermore, for the semester in hand, the percentages of students in clusters is very different from other semesters from last two years. Clusters 3, 7 and 8 are much more present in current semester than in last two years. On the other hand, clusters 6, 13 and 14 are not present in current semester but they made 14%, 9% and 12% of all predictions previously.

When analyzing those clusters that were more likely predicted on current semester, then those clusters are the most common "fail" cluster with students with little activity and clusters that actually have good results. As said before, during analyzed semester students tended to work more to get better grades than during last two years. Cluster 7, "early-rushers", is actually the only cluster where highest grade prediction is possible. Therefore it is expected that more students got clustered into group 7 during this semester. Cluster 8 also has students with good results.

Popularity of fail cluster, cluster 3, can be explained by changes in curriculum. The course is now taught for the freshmen on the first semester and they have high percentage of drop-outs from university during the first year, especially during the first semester.

When looking further into the clusters that were more common in model than during the actual test semester, changes are harder to explain. Cluster 14 contains constant workers during the whole semester who are doing well in theoretical tasks but not in lab experiments. Another cluster, 9, that also has constant workers, was also less frequent during fall 2015. Therefore, it could be argued that students are becoming less likely to work through all semester. If they are good at planning, they try to complete the course quickly and move on to other courses. If they are lazier, they do not start working before second half of the course. That can be verified by other statistics from the courses.

Cluster 6, students who work at the second half of the semester, who do a lot of submissions and usually get bad results but manage to finish with grades 3 or 4, has always been little "unusual". Cluster 13 contains students with very uneven study patterns. Seems that none of those quite extreme cases was present with students on 2015 fall semester.

When looking at the finishing time prediction, the results are less favorable. Only in 64 cases (58.18%) the actual finishing is within three weeks of predicted finishing week. Three weeks were chosen for prediction quality measure due to the fact that for some groups the finishing time is predicted to be "within exam session" and exam session is 3 weeks long. There were some extreme cases when student actually passed on week 9 but the group he/she was clustered expected the student to finish within exam session. The four cases that were problematic for the final grade prediction, were also among "extremely-off" cases.

8 Discussion

Build model seems to working satisfactorily for final grade prediction with more than 80% students' being predicted the right grade. Also, in majority of other cases the fact that their final grade was better than predicted can be seen as good result because trying harder was a characteristic for the semester considered. As mentioned previously, the model should be re-evaluated regularly so that newest students' trend would be taken into account. It will be interesting to follow chances from year to year when model is going to be re-evaluated.

The changes in clusters are more rapid than expected – the changes continued through the teaching period, slowing down by the end of it. Even during the exam session, there were slight changes. It is also interesting to follow that some of the changes have a pattern that continues throughout the weeks. For example, from week 12 onwards many students from group 2 are moved to group 3 when they are inactive. Also changes from cluster 5 to 7 or 9 to 8 happens often during the second half of the semester. It would be preferred if the changes in clusters could be lowered down even further by fine-tuning the model. Maybe some of the features considered before PCA should be reintroduced into clustering if they have the desired effect on the cluster prediction.

Using the model for predicting the students' finishing time does not yield such good results. The finishing time, which can be anything from week 1 to week 20, was averaged over the cluster with extreme cases ignored. Also, as there are more possible values than for final grade, it could be expected that averaged result will have some variety. The fact that there were quite extreme cases shows that there is room for improvement for the model for final grade prediction. Maybe separate model that only concentrates on the finishing time should be developed. The other option could be that when cluster is found, student's histogram is compared with all the previous students within the cluster and that student's finishing time could be used as a prediction value.

The first prediction was tested only with the largest course and students weren't shown the prediction – it was due to the fact that we had no guarantee how well the model works. Now, when the models effectiveness has been proven, it could be also shown to students. The key concerns are how students react to the prediction and how can we measure if changes in students' behavior are due to the characteristics of that semester or due to seeing the prediction model. Showing predictions to students will be incorporated into the main page when students have logged into the system so that they do not have to specifically go and look for that information.

Another unknown factor is how students are going to react to the predictions. In psychological point of view, it has been advised, that we have to be careful how we phrase the descriptions of clusters and recommendations as we could achieve the opposite effect of to our goal – students get scared off as they are predicted to fail. On the other hand, predicting that student is going to fail usually happens at the latter half of the semester when no activity has been shown. Therefore, the prediction's description for students' in danger of later being clustered into the "drop-out" cluster, could remind them to start working earlier instead of stating that they are probably going to fail.

Also, many of the courses taught on the system have enough students for reasonable model. There are also some courses that may have only 10 or less students per semester. For those kind of courses, developing a prediction model based on current methodology is not possible. Therefore, probably, we are only going to be able to offer prediction for the largest courses in the system.

9 Conclusion

Prediction model for final grade and finishing time for competence-based e-learning system based on last two years students' activates was developed in this paper. Last two years were chosen due to the fact that during the analysis, it was found that students' behaviors are changing from year to year and, to keep the model up-to-date, only portion of available data should be used.

Prediction model was developed using data mining with regression analysis, principle component analysis and hierarchical clustering of symbolic histogram valued data. 35 different features of students' activates was considered but only 9 most important, so called principle components, were used in the model. Those featured were number of tasks done, active days, correctness of submissions, difficulty level of submissions, number of times help material accessed, time of the day of submissions, current process on the course, times of HomeLabKit borrowed and times of class test done.

Using hierarchical clustering that used cluster compactness as a dissimilarity measure, it was found that it is not possible to develop a single model for all the courses but a specific course-centered models had to be developed. Clusters with similar study behaviors were found. Their number and size varied from course to course but there were clusters with similar behaviors common for all 5 models currently developed. Those similar behaviors are students who start early and finish early; students who start working at the end with lot of submissions; constant workers; students who do very little; students who try the course at the beginning and start working again later half of the semester; and those student who start working at the second half of the semester.

To verify the prediction model, the largest course with 14 distinct clusters during 2015 fall semester was chosen. From week 4 onwards, prediction for every student's learning was done on weekly basis. It was found, that on 80.91% of cases the final grade was predicted correctly. Majority of other cases, the prediction was one grade lower than the final result. It was concluded to be due to that semester's characteristic – students worked more to get better grades than in previous years. In 3.63% of cases, the prediction failed: student was predicted to pass with 3 but instead they failed the course, not finishing on time. That occurrence have to be further analyzed and model corrected accordingly. It was also found that during the semesters, predicted clusters are changing up to the end of the semester but final grade prediction remains more constant. Changes in predicted groups slowed down during the semester as more data becomes available. It was also found that majority of students, who fail the course, are at first clustered into group who start working very late into the semester but pass the course. Only at the end of the semester, actual fail becomes predictable from lack of activity.

Predicting finishing time did not yield as good results as predicting the final grade. Therefore, it would be beneficial, if cluster has been found, find previous students within the cluster with most similar study behavior and use his/her finishing time as a prediction instead cluster's average finishing time.

Overall, the model worked satisfactorily and it will be introduced into student view in the e-learning system.

Acknowledgments. This research was supported by European Social Fund's Doctoral Studies and Internationalization Programme DoRa and DoRa+.

References

1. Kukk V, Jaanus M, Umbleja K, Shvets O Developing integrated learning environment–analysis. In: International Conference on Engineering Education, ICEE 2010, pp 18–22 (2010)
2. Nussbaumer, A., Hillemann, E.-C., Gütl, C., Albert, D.: A competence-based service for supporting self-regulated learning in virtual environments. J. Learn. Analytics **2**(1), 101–133 (2015)
3. Baumgartner, I., Shankararaman, V.: Actively linking learning outcomes and competencies to course design and delivery: experiences from an undergraduate Information Systems program in Singapore. In: 2013 IEEE Global Engineering Education Conference (EDUCON), pp. 238–246. IEEE (2013)
4. Berges, M., Hubwieser, P., Magenheim, J., Bender, E., Bröker, K., Margaritis-Kopecki, M., Neugebauer, J., Schaper, N., Schubert, S., Ohrndorf, L.: Developing a competency model for teaching computer science in schools. In: Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education, p. 327. ACM (2013)
5. Zehetmeier, D., Kuhrmann, M., Bottcher, A., Schlierkamp, K., Thurner, V.: Self-assessment of freshmen students' base competencies. In: 2014 IEEE Global Engineering Education Conference (EDUCON), pp. 429–438. IEEE (2014)
6. Marienau, C.: An Integrated Approach to Competency-Based Learning & Assessment: Two Models at the School for New Learning, DePaul University (2015)
7. Rozeboom, H.: Competence based learning. Bull. Univ. Agric. Sci. Vet. Med. Cluj-Napoca Hortic. **63**(1–2), 365–367 (2006)
8. Sánchez, A.V., Ruiz, M.P.: Competence-based learning: a proposal for the assessment of generic competences. Universidad de Deusto (2008)
9. Jaanus, M.: The interactive learning environment for mobile laboratories. Tallinn University of Technology (2011)
10. Umbleja, K.: Students' grading control and visualisation in competence-based learning approach. In: 2015 IEEE Global Engineering Education Conference (EDUCON), pp. 287–296. IEEE (2015)
11. Umbleja, K., Kukk, V., Jaanus, M., Udal, A.: New concepts of automatic answer evaluation in competence based learning. In: 2014 IEEE Global Engineering Education Conference (EDUCON), pp. 922–925. IEEE (2014)
12. Kukk, V., Umbleja, K.: Analysis of forgetting in a learning environment. In: 2012 13th Biennial Baltic Electronics Conference (BEC), pp 335–338. IEEE (2012)
13. Chen, C.-M.: Intelligent web-based learning system with personalized learning path guidance. Comput. Educ. **51**(2), 787–814 (2008)

14. Romero, C., Ventura, S., Pechenizkiy, M., Baker, R.S.: Handbook of Educational Data Mining. CRC Press, Boca Raton (2010)
15. Baradwaj, B.K., Pal S.: Mining educational data to analyze students' performance. arXiv preprint arXiv:1201.3417
16. Dutt, A., Aghabozrgi, S., Ismail, M.A.B., Mahroeian, H.: Clustering algorithms applied in educational data mining. Int. J. Inf. Electron. Eng. **5**(2), 112 (2015)
17. Gobert, J.D., Sao Pedro, M., Raziuddin, J., Baker, R.S.: From log files to assessment metrics: Measuring students' science inquiry skills using educational data mining. J. Learn. Sci. **22** (4), 521–563 (2013)
18. Tair, M.M.A., El-Halees, A.M.: Mining educational data to improve students' performance: a case study. Int. J. Inf. **2**(2), 140–146 (2012)
19. Montgomery, D.C., Peck, E.A., Vining, G.G.: Introduction to Linear Regression Analysis. Wiley (2015)
20. Kaytez, F., Taplamacioglu, M.C., Cam, E., Hardalac, F.: Forecasting electricity consumption: a comparison of regression analysis, neural networks and least squares support vector machines. Int. J. Electr. Power Energy Syst. **67**, 431–438 (2015)
21. Kotsiantis, S.B.: Use of machine learning techniques for educational proposes: a decision support system for forecasting students' grades. Artif. Intell. Rev. **37**(4), 331–344 (2012)
22. Richardson, M.: Principal component analysis (2009). http://www.sdss.jhu.edu/~szalay/class/2015/SignalProcPCA.pdf. Accessed 03 Jun 2016
23. Jolliffe, I.: Principal Component Analysis. Wiley Online Library (2002)
24. Billard, L.: Symbolic data analysis: what is it? In: Rizzi, A., Vichi, M. (eds.) Compstat 2006 - Proceedings in Computational Statistics, pp. 261–269. Springer, Heidelberg (2006)
25. Billard, L., Diday, E.: Symbolic Data Analysis: Conceptual Statistics and Data Mining. Wiley (2006)
26. Ichino, M., Yaguchi, H.: Generalized Minkowski metrics for mixed feature-type data analysis. IEEE Trans. Syst. Man Cybern. **24**(4), 698–708 (1994)
27. Umbleja, K., Ichino, M., Liiv, I.: A dissimilarity measure for general histogram data and its application to hierarchical clustering (2016, in preparation)
28. Foundation TR. The R Project for Statistical Computing (2016). https://www.r-project.org/. Accessed 03 Jun 2016
29. Everitt, B.: Dictionary of Statistics. Cambridge University Press, Cambridge (1998)

DISSERTATIONS DEFENDED AT TALLINN UNIVERSITY OF TECHNOLOGY ON *INFORMATICS AND SYSTEM ENGINEERING*

1. **Lea Elmik**. Informational Modelling of a Communication Office. 1992.
2. **Kalle Tammemäe**. Control Intensive Digital System Synthesis. 1997.
3. **Eerik Lossmann**. Complex Signal Classification Algorithms, Based on the Third-Order Statistical Models. 1999.
4. **Kaido Kikkas**. Using the Internet in Rehabilitation of People with Mobility Impairments – Case Studies and Views from Estonia. 1999.
5. **Nazmun Nahar**. Global Electronic Commerce Process: Business-to-Business. 1999.
6. **Jevgeni Riipulk**. Microwave Radiometry for Medical Applications. 2000.
7. **Alar Kuusik**. Compact Smart Home Systems: Design and Verification of Cost Effective Hardware Solutions. 2001.
8. **Jaan Raik**. Hierarchical Test Generation for Digital Circuits Represented by Decision Diagrams. 2001.
9. **Andri Riid**. Transparent Fuzzy Systems: Model and Control. 2002.
10. **Marina Brik**. Investigation and Development of Test Generation Methods for Control Part of Digital Systems. 2002.
11. **Raul Land**. Synchronous Approximation and Processing of Sampled Data Signals. 2002.
12. **Ants Ronk**. An Extended Block-Adaptive Fourier Analyser for Analysis and Reproduction of Periodic Components of Band-Limited Discrete-Time Signals. 2002.
13. **Toivo Paavle**. System Level Modeling of the Phase Locked Loops: Behavioral Analysis and Parameterization. 2003.
14. **Irina Astrova**. On Integration of Object-Oriented Applications with Relational Databases. 2003.
15. **Kuldar Taveter**. A Multi-Perspective Methodology for Agent-Oriented Business Modelling and Simulation. 2004.
16. **Taivo Kangilaski**. Eesti Energia käiduhaldussüsteem. 2004.
17. **Artur Jutman**. Selected Issues of Modeling, Verification and Testing of Digital Systems. 2004.
18. **Ander Tenno**. Simulation and Estimation of Electro-Chemical Processes in Maintenance-Free Batteries with Fixed Electrolyte. 2004.

19. **Oleg Korolkov**. Formation of Diffusion Welded Al Contacts to Semiconductor Silicon. 2004.

20. **Risto Vaarandi**. Tools and Techniques for Event Log Analysis. 2005.

21. **Marko Koort**. Transmitter Power Control in Wireless Communication Systems. 2005.

22. **Raul Savimaa**. Modelling Emergent Behaviour of Organizations. Time-Aware, UML and Agent Based Approach. 2005.

23. **Raido Kurel**. Investigation of Electrical Characteristics of SiC Based Complementary JBS Structures. 2005.

24. **Rainer Taniloo**. Ökonoomsete negatiivse diferentsiaaltakistusega astmete ja elementide disainimine ja optimeerimine. 2005.

25. **Pauli Lallo**. Adaptive Secure Data Transmission Method for OSI Level I. 2005.

26. **Deniss Kumlander**. Some Practical Algorithms to Solve the Maximum Clique Problem. 2005.

27. **Tarmo Veskioja**. Stable Marriage Problem and College Admission. 2005.

28. **Elena Fomina**. Low Power Finite State Machine Synthesis. 2005.

29. **Eero Ivask**. Digital Test in WEB-Based Environment 2006.

30. **Виктор Войтович**. Разработка технологий выращивания из жидкой фазы эпитаксиальных структур арсенида галлия с высоковольтным p-n переходом и изготовления диодов на их основе. 2006.

31. **Tanel Alumäe**. Methods for Estonian Large Vocabulary Speech Recognition. 2006.

32. **Erki Eessaar**. Relational and Object-Relational Database Management Systems as Platforms for Managing Softwareengineering Artefacts. 2006.

33. **Rauno Gordon**. Modelling of Cardiac Dynamics and Intracardiac Bio-impedance. 2007.

34. **Madis Listak**. A Task-Oriented Design of a Biologically Inspired Underwater Robot. 2007.

35. **Elmet Orasson**. Hybrid Built-in Self-Test. Methods and Tools for Analysis and Optimization of BIST. 2007.

36. **Eduard Petlenkov**. Neural Networks Based Identification and Control of Nonlinear Systems: ANARX Model Based Approach. 2007.

37. **Toomas Kirt**. Concept Formation in Exploratory Data Analysis: Case Studies of Linguistic and Banking Data. 2007.

38. **Juhan-Peep Ernits**. Two State Space Reduction Techniques for Explicit State Model Checking. 2007.

39. **Innar Liiv**. Pattern Discovery Using Seriation and Matrix Reordering: A Unified View, Extensions and an Application to Inventory Management. 2008.

40. **Andrei Pokatilov**. Development of National Standard for Voltage Unit Based on Solid-State References. 2008.

41. **Karin Lindroos**. Mapping Social Structures by Formal Non-Linear Information Processing Methods: Case Studies of Estonian Islands Environments. 2008.

42. **Maksim Jenihhin**. Simulation-Based Hardware Verification with High-Level Decision Diagrams. 2008.

43. **Ando Saabas**. Logics for Low-Level Code and Proof-Preserving Program Transformations. 2008.

44. **Ilja Tšahhirov**. Security Protocols Analysis in the Computational Model – Dependency Flow Graphs-Based Approach. 2008.

45. **Toomas Ruuben**. Wideband Digital Beamforming in Sonar Systems. 2009.

46. **Sergei Devadze**. Fault Simulation of Digital Systems. 2009.

47. **Andrei Krivošei**. Model Based Method for Adaptive Decomposition of the Thoracic Bio-Impedance Variations into Cardiac and Respiratory Components. 2009.

48. **Vineeth Govind**. DfT-Based External Test and Diagnosis of Mesh-like Networks on Chips. 2009.

49. **Andres Kull**. Model-Based Testing of Reactive Systems. 2009.

50. **Ants Torim**. Formal Concepts in the Theory of Monotone Systems. 2009.

51. **Erika Matsak**. Discovering Logical Constructs from Estonian Children Language. 2009.

52. **Paul Annus**. Multichannel Bioimpedance Spectroscopy: Instrumentation Methods and Design Principles. 2009.

53. **Maris Tõnso**. Computer Algebra Tools for Modelling, Analysis and Synthesis for Nonlinear Control Systems. 2010.

54. **Aivo Jürgenson**. Efficient Semantics of Parallel and Serial Models of Attack Trees. 2010.

55. **Erkki Joasoon**. The Tactile Feedback Device for Multi-Touch User Interfaces. 2010.

56. **Jürgo-Sören Preden**. Enhancing Situation – Awareness Cognition and Reasoning of Ad-Hoc Network Agents. 2010.

57. **Pavel Grigorenko**. Higher-Order Attribute Semantics of Flat Languages. 2010.

58. **Anna Rannaste**. Hierarcical Test Pattern Generation and Untestability Identification Techniques for Synchronous Sequential Circuits. 2010.

59. **Sergei Strik**. Battery Charging and Full-Featured Battery Charger Integrated Circuit for Portable Applications. 2011.

60. **Rain Ottis**. A Systematic Approach to Offensive Volunteer Cyber Militia. 2011.

61. **Natalja Sleptšuk**. Investigation of the Intermediate Layer in the Metal-Silicon Carbide Contact Obtained by Diffusion Welding. 2011.

62. **Martin Jaanus**. The Interactive Learning Environment for Mobile Laboratories. 2011.

63. **Argo Kasemaa**. Analog Front End Components for Bio-Impedance Measurement: Current Source Design and Implementation. 2011.

64. **Kenneth Geers**. Strategic Cyber Security: Evaluating Nation-State Cyber Attack Mitigation Strategies. 2011.

65. **Riina Maigre**. Composition of Web Services on Large Service Models. 2011.

66. **Helena Kruus**. Optimization of Built-in Self-Test in Digital Systems. 2011.

67. **Gunnar Piho**. Archetypes Based Techniques for Development of Domains, Requirements and Sofware. 2011.

68. **Juri Gavšin**. Intrinsic Robot Safety Through Reversibility of Actions. 2011.

69. **Dmitri Mihhailov**. Hardware Implementation of Recursive Sorting Algorithms Using Tree-like Structures and HFSM Models. 2012.

70. **Anton Tšertov**. System Modeling for Processor-Centric Test Automation. 2012.

71. **Sergei Kostin**. Self-Diagnosis in Digital Systems. 2012.

72. **Mihkel Tagel**. System-Level Design of Timing-Sensitive Network-on-Chip Based Dependable Systems. 2012.

73. **Juri Belikov**. Polynomial Methods for Nonlinear Control Systems. 2012.

74. **Kristina Vassiljeva**. Restricted Connectivity Neural Networks based Identification for Control. 2012.

75. **Tarmo Robal**. Towards Adaptive Web – Analysing and Recommending Web Users` Behaviour. 2012.

76. **Anton Karputkin**. Formal Verification and Error Correction on High-Level Decision Diagrams. 2012.

77. **Vadim Kimlaychuk**. Simulations in Multi-Agent Communication System. 2012.

78. **Taavi Viilukas**. Constraints Solving Based Hierarchical Test Generation for Synchronous Sequential Circuits. 2012.

79. **Marko Kääramees**. A Symbolic Approach to Model-based Online Testing. 2012.

80. **Enar Reilent**. Whiteboard Architecture for the Multi-agent Sensor Systems. 2012.

81. **Jaan Ojarand**. Wideband Excitation Signals for Fast Impedance Spectroscopy of Biological Objects. 2012.

82. **Igor Aleksejev**. FPGA-based Embedded Virtual Instrumentation. 2013.

83. **Juri Mihhailov**. Accurate Flexible Current Measurement Method and its Realization in Power and Battery Management Integrated Circuits for Portable Applications. 2013.

84. **Tõnis Saar**. The Piezo-Electric Impedance Spectroscopy: Solutions and Applications. 2013.

85. **Ermo Täks**. An Automated Legal Content Capture and Visualisation Method. 2013.

86. **Uljana Reinsalu**. Fault Simulation and Code Coverage Analysis of RTL Designs Using High-Level Decision Diagrams. 2013.

87. **Anton Tšepurov**. Hardware Modeling for Design Verification and Debug. 2013.

88. **Ivo Müürsepp**. Robust Detectors for Cognitive Radio. 2013.

89. **Jaas Ježov**. Pressure sensitive lateral line for underwater robot. 2013.

90. **Vadim Kaparin**. Transformation of Nonlinear State Equations into Observer Form. 2013.

92. **Reeno Reeder**. Development and Optimisation of Modelling Methods and Algorithms for Terahertz Range Radiation Sources Based on Quantum Well Heterostructures. 2014.

93. **Ants Koel**. GaAs and SiC Semiconductor Materials Based Power Structures: Static and Dynamic Behavior Analysis. 2014.

94. **Jaan Übi**. Methods for Coopetition and Retention Analysis: An Application to University Management. 2014.

95. **Innokenti Sobolev**. Hyperspectral Data Processing and Interpretation in Remote Sensing Based on Laser-Induced Fluorescence Method. 2014.

96. **Jana Toompuu**. Investigation of the Specific Deep Levels in *p*-, *i*- and *n*-Regions of GaAs p^+-pin-n^+ Structures. 2014.

97. **Taavi Salumäe**. Flow-Sensitive Robotic Fish: From Concept to Experiments. 2015.

98. **Yar Muhammad**. A Parametric Framework for Modelling of Bioelectrical Signals. 2015.

99. **Ago Mõlder**. Image Processing Solutions for Precise Road Profile Measurement Systems. 2015.

100. **Kairit Sirts**. Non-Parametric Bayesian Models for Computational Morphology. 2015.

101. **Alina Gavrijaševa**. Coin Validation by Electromagnetic, Acoustic and Visual Features. 2015.

102. **Emiliano Pastorelli**. Analysis and 3D Visualisation of Microstructured Materials on Custom-Built Virtual Reality Environment. 2015.

103. **Asko Ristolainen**. Phantom Organs and their Applications in Robotic Surgery and Radiology Training. 2015.

104. **Aleksei Tepljakov**. Fractional-order Modeling and Control of Dynamic Systems. 2015.

105. **Ahti Lohk**. A System of Test Patterns to Check and Validate the Semantic Hierarchies of Wordnet-type Dictionaries. 2015.

106. **Hanno Hantson**. Mutation-Based Verification and Error Correction in High-Level Designs. 2015.

107. **Lin Li**. Statistical Methods for Ultrasound Image Segmentation. 2015.

108. **Aleksandr Lenin**. Reliable and Efficient Determination of the Likelihood of Rational Attacks. 2015.

109. **Maksim Gorev**. At-Speed Testing and Test Quality Evaluation for High-Performance Pipelined Systems. 2016.

110. **Mari-Anne Meister**. Electromagnetic Environment and Propagation Factors of Short-Wave Range in Estonia. 2016.

111. **Syed Saif Abrar**. Comprehensive Abstraction of VHDL RTL Cores to ESL SystemC. 2016.

112. **Arvo Kaldmäe**. Advanced Design of Nonlinear Discrete-time and Delayed Systems. 2016.

113. **Mairo Leier**. Scalable Open Platform for Reliable Medical Sensorics. 2016.

114. **Georgios Giannoukos**. Mathematical and Physical Modelling of Dynamic Electrical Impedance. 2016.

115. **Aivo Anier**. Model Based Framework for Distributed Control and Testing of Cyber-Physical Systems. 2016.

116. **Denis Firsov**. Certification of Context-Free Grammar Algorithms. 2016.

117. **Sergei Astatpov**. Distributed Signal Processing for Situation Assessment in Cyber-Physical Systems. 2016.

118. **Erkki Moorits**. Embedded Software Solutions for Development of Marine Navigation Light Systems. 2016.

119. **Andres Ojamaa**. Software Technology for Cyber Security Simulations. 2016.

120. **Gert Toming**. Fluid Body Interaction of Biomimetic Underwater Robots. 2016.