

Towards a Trace-based Evaluation Model for Knowledge Acquisition and Training Resource Adaption

Soraya Chachoua, Nouredine Tamani, Jamal Malki and Pascal Estraillier
L3i Laboratory, University of La Rochelle, Avenue Michel Crépeau, La Rochelle, France

Keywords: e-Assessment, Scoring, Temporel Trace, Attempts Trace, Training Quality, Learner Behaviour, Resource Adaption.

Abstract: e-Assessment in an e-learning system is aimed at evaluating learners regarding their knowledge acquisition. Available assessment methods are usually used at the end of a training activity in order to state if a given learner has either passed or failed a training unit or level, based on the grading results obtained. Most of grading processes follow the SCORM norm in the matter (Scorm, 2006) and make use of duration and number of attempts to compute the scores. These information are valuable in grading but they can also be exploited to capture the learner behaviour during a training activity, and then assess both learner knowledge acquisition and training resource quality in terms of adequacy. Therefore, we consider in this paper duration and number of attempts as modeled traces, upon which we build a theoretical model for automated evaluation of learners' knowledge acquisition evolution as a training activity progresses. The values obtained can be used to adapt training strategies and resources to improve both learner's knowledge level and e-learning platform quality.

1 INTRODUCTION AND MOTIVATION

In recent years, the use of e-learning platforms has seen a tremendous increase in the training field. The more so with the large adoption of ICT in teaching processes (Martin, 2008), in which a learner is empowered with tools and materials to acquire new knowledge such as courses, exercises, quizzes, etc., which are organized in such a way that learners can manage and monitor by themselves the training activities. In order to be able to track their progress in terms of knowledge acquisition, learners have to take tests and exams in the form of multiple-choice questions and/or fill-in-the-blank exercises (Nicol, 2007; Crisp, 2009), which deliver a static evaluation based on learner's answers. These evaluation methods, in the sense of grading or scoring (Crooks, 1988) functions, are parts of a so-called e-Assessment component widely integrated into e-learning platforms.

As stated in (Wandall, 2011), testing and test results can be used in order to assess the learner's knowledge and competencies, as a pedagogical tool to assess the effect of the teaching, but also for admission, regulation, controlling, rewarding/punishment of individuals/schools. Early student evaluation is important in any pedagogical process. Its role is

twofold; first, it gives the possibility to quantify the level of the learner and thus optimally adapt learning materials and processes accordingly, and second, it can identify learner progress patterns after some training sessions. Therefore, more sophisticated student evaluation or scoring approaches have been proposed which harness some additional information, called traces, such as in (Kumar et al., 2015; Papamitsiou and Economides, 2015). In (Papamitsiou and Economides, 2015), temporal trace data visualization during the assessment has been proposed to help actors involved in the activity to intuitively interpret and quickly perceive a concealed feature. In (Kumar et al., 2015), an on-line platform called *Learnform* has been developed for problem solving in the domain of Electricity and Magnetism. Student progress evaluation is based on platform's *logs*, upon which Cohen's statistical approach (Cohen, 1977) has been applied to compare between student's post-test scores with his/her pre-test ones and state either the student has improved or not his/her knowledge.

Given the possibility to attach the duration and number of attempts in an interactive training activity, we can make use of these information as traces to evaluate both learner level and progress. Indeed, traces generated by a learner during his/her training activity are valuable in the process of evaluation

of his/her knowledge acquisition level (Lebis et al., 2015). Moreover, trace-based evaluation process allows to compute relevant indicators that permit the personalization of the analysis (Settoui et al., 2009c), since traces can be attached to a unique user, and by personalization we can obtain more accuracy in the behavior modeling and analysis. For instance, a student who took twice as much time as recommended to solve a kind of problem can reveal some difficulties faced by the learner to acquire the corresponding knowledge. The reason could be related to student deficiencies in prerequisite skills or the training strategy (training materials and processes) is not adequate, or both. Therefore, traces can disclose and reveal new knowledge about both learners and teaching strategies, which are actually difficult to reach with the traditional evaluation methods. Again, traces about the number of tries or attempts students carried out to solve a problem, for instance, can inform experts about both the level of knowledge the learners mastered and the adequacy of the teaching materials to learners' profiles.

Subsequently, we introduce in this paper a theoretical model for both student level and progress rhythm evaluation based on two types of trace, namely, (i) time spent to solve a problem, (ii) the number of attempts or tries to solve it. Their combination is also taken into account. In practical terms, we propose three mathematical functions which compute learner's performance indicators (we refer the reader to (Wilhelm, 2009) for more details about performance indicators in computer-based assessment) and refined scores for the level of knowledge the learner has acquired, and the way he/she managed to handle training activities. The proposed approach is a first step towards a trace-based learner behavior model for knowledge acquisition and training resource adaption. The significance of our approach is twofold:

- computing personalized e-learning performance indicators for training materials and process adaption. For instance, if the system notices that a learner frequently fails to solve a given kind of problem, then it recommends some actions to be performed to adapt the contents and/or in the training processes to knowledge level of the learner,
- computing a personalized and refined assessment of a learner regarding his/her knowledge acquisition; thus the approach proposed can be seen as a scoring or a grading method. In effect, the approach introduced can distinguish between students having answered correctly to a given test activity but in different duration and number of attempts.

The remainder of the paper is organized as follows. Section 3 summarizes some definitions about traces and learner evaluation; then describes the positioning of our approach within a modeled trace-based assessment process. Section 4 details our trace-based assessment model. Section 2 discusses some related work in the domain of e-Assessment. Finally, Section 5 concludes the paper and gives some perspectives for further research.

2 RELATED WORK

Automatic assessment or computer-based assessment is considered as an important tool in educational system as detailed in the large report of the European Union (Scheuermann and Björnsson, 2009). This latter is a collection of articles and studies about the assessment topic from diverse standpoints encompassing assessment needs in European countries, Computer Adaptive Testing (CAT), and methodologies of computer-based testing and benchmarking, to name a few. Amongst this collection, the authors of study (Kozma, 2009) pointed out the limitation of traditional assessment methods as tools for measuring the competencies and skills that are needed to be mastered by the learners. Our proposed approach can be seen as an improvement of such methods and also as a step towards a computer adaptive testing based on learners' traces.

In (Crisp, 2009) three kinds of assessment and their relationships have been introduced, namely: (i) *diagnostic* which is performed before the learning to estimate the level of students, (ii) *formative* which is applied during the learning to test the knowledge evolution and (iii) *summative* which is considered after the learning to assess the knowledge acquired. These assessment steps allow learners to follow their progress and track their objectives targeted during a given activity in order to know if they passed or failed a given training level. In (Amelung et al., 2011), the authors designed and implemented a generic support of assessment for assignments in programming languages, which could be integrated into on-line laboratory (Guo, 2013). They also developed an eduComponents (<http://wdok.cs.ovgu.de/eduComponents/>) module for student evaluation, which only provides a semi-automated and static assessment which is performed by the tutor under the guidance of the tool. Thus, the formative and summative assessments are carried out by the instructor. However, eduComponents offers only a semi-automated and static assessment which is done by the tutor and assisted by the tool

during student evaluation process. In (Nicol and Macfarlane-Dick, 2006; Nicol, 2007), a large study of formative assessment and feedback has been carried out in order to identify how assessment processes can help the enhancement and the development of learner's self-direction and regulation. Our approach is actually a technical solution which covers the three assessment types, since time and number of attempts are useful traces for diagnostic processes to evaluate learners knowledge level. The proposed approach can also be used during the training process, providing formative evaluation or after training session implementing a summative evaluation.

Finally, it is noteworthy that SCORM norm (Scorm, 2006) considers both duration and attempts in test activities, such as those implemented in language e-learning platforms, IT certification courses and exams, hiring tests, etc. In this case, duration and number if attempts are not considered as traces, as in our model, but as parameters to set for a test or an exam, in order to put learners under a temporal stress by imposing on them to answer some questions or solve puzzles in a given time, usually short, eventually combined with a limited number of authorized tries or attempts, beyond which the system considers that the candidate failed to answer the question or solve the problem.

3 CONTEXT AND DEFINITIONS

In this section we briefly define some concepts useful for the description of our approach. Section 3.1 introduces definitions of trace and their related concepts. Section 3.2 gives our definition of evaluation function in an assessment process.

3.1 Trace Definition

In literature, the notion of trace often refers to log files describing events happened in a given system (Andrews, 1998). In (Djouad et al., 2010) a trace is a series of things left by any action related to a being or an object; a series of imprints or marks left by the passage of an object or a being, which reveals that something has existed. In our context, a digital trace is any piece of information captured by observation processes within an e-Learning activity. It represents a residual footprint of interactions between learners and training processes. A digital trace about an activity consists of objects which are chronologically observed, captured and saved on a support. Such traces are collected and analyzed by Trace-Based Systems (TBS) (Settouti et al., 2009a; Laflaquiere et al., 2006).

A TBS considers three components as described in (Mille et al., 2013).

- Observer: it is a system agent called *obsel*, which is attached to event occurred during an execution of an application. Each Obsel is characterized by a name, a timestamp, and a set of properties.
- Trace model: is an abstract description of trace elements. In other terms, it represents a trace vocabulary and each trace can be associated to a model.
- M-Trace (for Modeled Trace): it is a trace attached with its model, and managed by an m-Trace-Based System (m-TBS). Figure 1 illustrates two examples of Obsels: User and Training situation, and their recorded traces modeled respectively by $\{user\ identifier\ (UserID),\ Date\}$ and $\{Name,\ Duration,\ Score,\ Attempts\}$.

The aforementioned modeled traces can be exploited within an e-Assessment component in both training and learner evaluation.

3.2 Assessment Definition

To the best of our knowledge, grading or scoring functions have been only considered in few work. For instance, the authors in (Burstein et al., 2001) defined the scoring function in a context of essay as a linear equation, and in (Yang et al., 2002), authors consider automated scores are consistent with the scores from expert human graders, fair and have been validated against external measures. As stated in (Kozma, 2009), there is a need for new theories and models of scoring the students' processes and strategies during assessments, as well as outcomes. As stated in (Williamson et al., 2012), the field of study of automated scoring remains new enough that the guidelines of best practice are still evolving.

In our case, we combine both aspects by defining a quantitative evaluation function, which computes a value in \mathbb{R} in contrast with symbolic evaluation function which returns a qualitative grading value such as $\{A^+, A, B, \dots, F\}$.

Definition 1 (Quantitative Scoring Function). A quantitative scoring function is any function f defined from $2^{\mathcal{A}}$, such as \mathcal{A} is the set of all correct answers, to $[0, M]$, such as $M \in \mathbb{N}$ is the maximum mark, and satisfies the following properties:

- f is fair: in the sense that f is a non-decreasing function over the set $2^{\mathcal{A}}$,
- f is bounded: $f(\emptyset) = 0$ and $f(\mathcal{A}) = M$,
- minimality in the sense that any set of answers \mathcal{B} containing \mathcal{A} , $f(\mathcal{B})$ should be less than $f(\mathcal{A})$.

We extend the definition of quantitative scoring function to traces to define trace-based quantitative scoring function.

Definition 2 (Trace-Based Quantitative Evaluation Function). Let f be a quantitative scoring function defined from $2^{\mathcal{A}}$ to $[0, M]$, and T a trace defined over a domain \mathbb{D} . A trace-based quantitative scoring function is a binary function g defined from $2^{\mathcal{A}} \times \mathbb{D}$ to $[0, M]$ as follows.

$$\begin{aligned} 2^{\mathcal{A}} \times \mathbb{D} &\longrightarrow [0, M] \\ (A, d) &\longmapsto g(f(A), d) \end{aligned}$$

such that g is bounded and fair.

3.3 Trace-based Automated Scoring Process

Figure 2 describes the components of the trace-based automated scoring process considered as follows:

- e-Learning system, equipped with a user interface to interact with users,
- e-Assessment module which implements the scoring and evaluation functions, and delivers the results to both learners and teachers/experts,
- m-trace database which is put between the e-learning system and the e-Assessment module. It is fed by the former with users' traces, and it feeds the latter with saved traces.

Its steps are also described as follows. Users interact with the system (e-Learning platform, on-line laboratory, etc.) in order to do a training activity. Their interaction actions are tracked and the traces are saved in trace database. The traces are then modeled based on the three phases of (i) data collection, (ii) data transformation, (iii) data analysis; this process is out of the scope of this paper. We refer the reader to (Settouti et al., 2009b) for more details. Then a testing phase (see (Patelis, 2000; Thompson and Wiess, 2009; Sireci and Luecht, 2012)) is launched to check whether the answer is correct or false. Other traces are also generated from this process which are useful to our evaluation function. After that, we apply one of our assessment models: (i) time trace-based model, (ii) attempt trace-based model and (iii) combined time and attempt trace-based model for both learner progression and training activities. Finally, a performance step can be carried out to compute indicators about learners progress and training resources and/or strategies adequacy, in the evolution phase of the e-Assessment process. The performance indicators are delivered to experts to act accordingly by adapting the training situations.

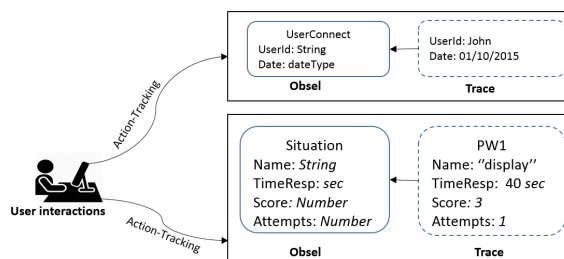


Figure 1: Examples of m-traces.

The following section details the trace-based model we propose.

4 TRACE-BASED ASSESSMENT AND EVALUATION MODEL

In this section we detail our evaluation models based on traces left by a learner during his/her training activity. In Subsection 4.1 a time trace model considering durations spent to solve a problem is introduced. In subsection 4.2, our second model based on number of tries is detailed. The combination of both traces is introduced in Subsection 4.3.

4.1 Time Trace-based Evaluation Model

We suppose that traces are considered regarding a single learner. A training activity can be attached with two timestamps corresponding respectively to its start instant and finish instant. These timestamps returns the duration spent by the learner to perform the considered activity. As mentioned in the introduction the time spent by the learner is useful for his/her evaluation. The intuition behind our time trace-based model is as follows.

- let a training objective be modeled as a set of m testing activities $O = \{A_1, \dots, A_m\}$, a testing activity can be any fill in blank exercise, reordering exercise, value to compute problem, single/multiple choice question, etc.
- each exercise or activity A_i with $i = 1, \dots, m$, is characterized by a full maximum mark, denoted by \mathcal{M}_{A_i} , which can be set by an expert and/or a trainer in the e-Learning system, or computed by a quantitative scoring function f (see Definition 1), and an optimal duration, denoted \mathcal{D}_{A_i} ,
- from temporal traces, it is possible to compute the duration a learner took to answer the problem; let us denote this duration by d ,
- if the answer of the learner is wrong, then his/her score is 0,

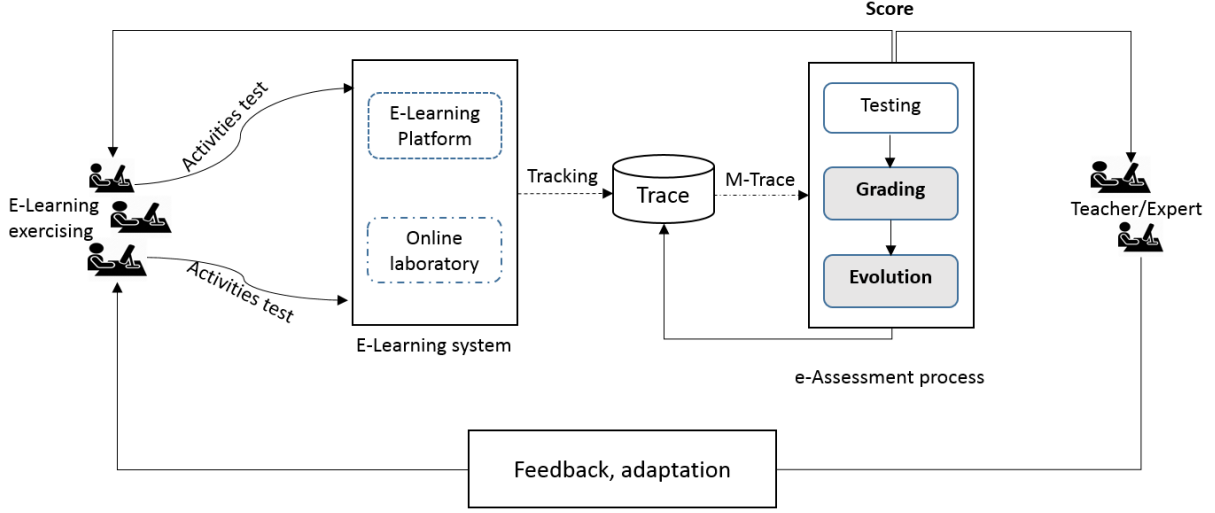


Figure 2: Traced-Based assessment process.

- if the learner answers well to the exercise in a duration less than or equal to the optimal duration \mathcal{D}_{A_i} , then he/she obtains the full mark \mathcal{M}_{A_i} ,
- if the learner took more than the optimal duration, then the longer the extra time he took to correctly answer the question, the less the mark will be.

The mark obtained by the learner for the testing activity A_i , is computed by the trace-based binary function $M_{A_i}(\mathcal{M}_{A_i}, d)$ (as introduced in Definition 2), and defined by the following formula (1) based on the duration taken to achieve it.

$$M_{A_i}(\mathcal{M}_{A_i}, d) = \begin{cases} \mathcal{M}_{A_i} & \text{if } (d \leq \mathcal{D}_{A_i}), \\ \mathcal{M}_{A_i} e^{-\alpha \left(\frac{d - \mathcal{D}_{A_i}}{\mathcal{D}_{A_i}} \right)} & \text{otherwise.} \end{cases} \quad (1)$$

such that:

- $\alpha \in [0, 1]$ is an attenuation constant. It allows to compute a score in $[0, \mathcal{M}_{A_i}]$, which indicates that $M_{A_i}(\mathcal{M}_{A_i}, d)$ is bounded. If α is close to 0, then $M_{A_i}(d)$ approaches \mathcal{M}_{A_i} . Otherwise if α approaches 1 then $M_{A_i}(\mathcal{M}_{A_i}, d)$ approaches 0. The closer to 1 α is, the harsher the attenuation will be. Moreover, for a fixed value for parameter α , the longer the duration d is, the lower the mark computed is, which shows the fairness of the binary function $M_{A_i}(\mathcal{M}_{A_i}, d)$,
- the fraction $\left(\frac{d - \mathcal{D}_{A_i}}{\mathcal{D}_{A_i}} \right)$ aims at normalizing the duration to eliminate the effect of its unit of measurement, and so for the mark computed.

It is worth noticing that the parameterized exponential term in Formula (1) permits to capture the natural fading effect of the mark when the extra time increases. Moreover, it well models the ability to express the importance of the skill the learner should

Table 1: Example of time trace-based model instantiation.

Learner	$M_{A_i}(\mathcal{M}_{A_i}, d)$			
	$d(s)$	$\alpha = 0.2$	$\alpha = 0.5$	$\alpha = 0.8$
l_1	200	5	5	5
l_2	300	5	5	5
l_3	530	4.29	3.41	2.71
l_4	620	4.04	2.93	2.13

acquire in a given training activity by configuring the constant parameter α . Indeed, the more important the skill is, the closer to 1 α should be. This flexibility is not easy to implement with a linear model. Besides, when $\alpha = 0$ then formula (1) reduces to a traditional scoring function, which returns the full mark if the answer is correct, independently of temporal traces.

Example 1. Let us suppose an e-Learning system for SQL training. Let us suppose an exercise about an SQL query for which we set the following parameters: (i) $\mathcal{M}_{A_i} = 5$, (ii) $\mathcal{D}_{A_i} = 300$ seconds. To illustrate the behavior of the model (formula (1)), we ran it thrice with three different values of α such that $\alpha = 0.2$ (low importance level activity), $\alpha = 0.5$ (medium importance level activity), and $\alpha = 0.8$ (high importance level activity). Table 2 summarizes the results obtained for 4 learners $\{l_1, l_2, l_3, l_4\}$ with their diverse duration taken to solve the exercise.

We notice that the marks decrease according to the extra-time took by the learner to do the activity. With $\alpha = 0.2$ the mark of learner l_4 has been diminished by about 20% compared with the full mark, even though he/she took more than twice as much time as recommended, and by more than 57% when $\alpha = 0.8$.

4.2 Attempt Trace-based Model

In this subsection we define our second model which is based on number of attempts (t). As mentioned in the introduction the number of attempts performed by the learner is also useful for his/her evaluation. The intuition behind the model is as follows.

- in this case, each exercise or activity A_i is characterized by a full maximum mark \mathcal{M}_{A_i} , which can be set by an expert and/or a trainer in the e-Learning system, or computed by a quantitative scoring function f (see Definition 1), and a number of allowed attempts, denoted \mathcal{T}_{A_i} ,
- from attempt traces, it is possible to compute the number of tries a given learner did to solve the problem; let us denote this attempt number by t ,
- if the answer of the learner is wrong, then his/her score is 0,
- if the learner answers well to the exercise in a number of tries less than or equal to \mathcal{T}_{A_i} , then he/she obtains the full mark \mathcal{M}_{A_i} ,
- if the learner made more attempts than \mathcal{T}_{A_i} , then the greater number of attempts he/she makes to answer the question, the less the mark will be.

The mark obtained by the learner for the testing activity A_i , is computed by the trace-based binary function $M_{A_i}(\mathcal{M}_{A_i}, t)$ (as introduced in Definition 2) defined by the following formula (2) based on the number of attempts t to achieve it.

$$M_{A_i}(\mathcal{M}_{A_i}, t) = \begin{cases} \mathcal{M}_{A_i} & \text{if } (t \leq \mathcal{T}_{A_i}), \\ \mathcal{M}_{A_i} e^{-\beta \left(\frac{t - \mathcal{T}_{A_i}}{\mathcal{T}_{A_i}} \right)} & \text{otherwise.} \end{cases} \quad (2)$$

such that:

- $\beta \in [0, 1]$: is an attenuation constant. It allows to compute a score in $[0, \mathcal{M}_{A_i}]$, therefore $M_{A_i}(\mathcal{M}_{A_i}, t)$ is bounded. If β is close to 0, then $M_{A_i}(\mathcal{M}_{A_i}, t)$ approaches \mathcal{M}_{A_i} . Otherwise, if β approaches 1 then $M_{A_i}(\mathcal{M}_{A_i}, t)$ approaches 0. The closer to 1 β is, the harsher the attenuation will be. Moreover, for a fixed parameter β , the greater the number of attempts t is, the lower the mark computed is. Therefore, function $M_{A_i}(\mathcal{M}_{A_i}, t)$ is fair,
- The fraction $\left(\frac{t - \mathcal{T}_{A_i}}{\mathcal{T}_{A_i}} \right)$ normalizes the number of tries, and so for the mark computed by the function $M_{A_i}(\mathcal{M}_{A_i}, t)$.

We note that properties of formula (1) are still valid for formula (2). Formula (2) is flexible enough (by the exponential term) to catch the fading effect, to express the importance of the training activity A_i , and behaves like a traditional scoring function when $\beta = 0$.

Table 2: Example of attempt trace-based model instantiation.

Learner	$M_{A_i}(\mathcal{M}_{A_i}, t)$			
	Tries t	$\beta = 0.2$	$\beta = 0.5$	$\beta = 0.8$
l_1	1	5	5	5
l_2	4	4.68	4.23	3.83
l_3	2	5	5	5
l_4	6	4.09	3.03	2,25

Example 2. Let us consider the same exercise of Example 1 for which the model parameters are set as follows: (i) $\mathcal{M}_{A_i} = 5$, and (ii) the number of allowed tries $\mathcal{T}_{A_i} = 3$. To illustrate the behavior of the model (formula (2)), we consider 4 learners $\{l_1, l_2, l_3, l_4\}$ and their respective number of attempts to resolve the exercise. Table 2 summarizes the results obtained for the learners, under the following values of β : 0.2 (low importance level activity), 0.5 (medium importance level activity), 0.8 (high importance level activity).

We notice the same effect of attenuation in the results and its severity when β increases (i.e becomes close to 1).

4.3 Combination of Time and Attempt Traces

In this subsection we define a combined model based on both duration and number of attempts traces. We recall that an activity A_i has a full mark \mathcal{M}_{A_i} , which can be set by an expert and/or a trainer in the e-Learning system, or computed by a quantitative scoring function f (see Definition 1), an optimal duration \mathcal{D}_{A_i} , and an authorized number of tries \mathcal{T}_{A_i} . Let us denote the duration taken by a learner and his/her number of attempts by d and t respectively. Therefore, the evaluation of the learner, denoted by $M_{A_i}(\mathcal{M}_{A_i}, d, t)$, according to these parameters is as follows.

- if $(d \leq \mathcal{D}_{A_i})$ and $(t \leq \mathcal{T}_{A_i})$ then $M_{A_i}(\mathcal{M}_{A_i}, d, t) = \mathcal{M}_{A_i}$ (the full mark),
- if $(d > \mathcal{D}_{A_i})$ and $(t \leq \mathcal{T}_{A_i})$ then $M_{A_i}(\mathcal{M}_{A_i}, d, t) = M_{A_i}(\mathcal{M}_{A_i}, d)$ (by using Formula (1))
- if $(d \leq \mathcal{D}_{A_i})$ and $(t > \mathcal{T}_{A_i})$ then $M_{A_i}(\mathcal{M}_{A_i}, d, t) = M_{A_i}(\mathcal{M}_{A_i}, t)$ (by using Formula (2))
- if $(d > \mathcal{D}_{A_i})$ and $(t > \mathcal{T}_{A_i})$ then we aggregate both traces intuitively by summing the extra-time and the extra number of attempts, then we apply the same reasoning as for Formulas (1) and (2) to define the evaluation function of three parameters $M_{A_i}(\mathcal{M}_{A_i}, d, t)$. Therefore, the combined model can be expressed by Formula (3) as follows.

Table 3: Example of combined duration and attempts traces instantiation.

Learner	$M_{A_i}(\mathcal{M}_{A_i}, d, t)$				
	d	t	$\gamma = 0.2$	$\gamma = 0.5$	$\gamma = 0.8$
l_1	200	1	5.00	5.00	5.00
l_2	300	4	4.68	4.23	3.83
l_3	530	2	4.29	3.41	2.71
l_4	620	6	3.31	1.78	0.96

$$M_{A_i}(\mathcal{M}_{A_i}, d, t) = \begin{cases} \mathcal{M}_{A_i} & \text{if } (d \leq \mathcal{D}_{A_i}) \wedge (t \leq \mathcal{T}_{A_i}) \\ \mathcal{M}_{A_i}(\mathcal{M}_{A_i}, d) & \text{if } (d > \mathcal{D}_{A_i}) \wedge (t \leq \mathcal{T}_{A_i}) \\ \mathcal{M}_{A_i}(\mathcal{M}_{A_i}, t) & \text{if } (d \leq \mathcal{D}_{A_i}) \wedge (t > \mathcal{T}_{A_i}) \\ \mathcal{M}_{A_i} e^{-\gamma \left(\frac{t - \mathcal{T}_{A_i}}{\mathcal{D}_{A_i}} + \frac{d - \mathcal{D}_{A_i}}{\mathcal{D}_{A_i}} \right)} & \text{otherwise.} \end{cases} \quad (3)$$

such that:

- $\gamma \in [0, 1]$ is an attenuation constant. It allows to compute a score in $[0, \mathcal{M}_{A_i}]$, therefore the function $M_{A_i}(\mathcal{M}_{A_i}, d, t)$ is bounded. If γ is close to 0, then $M_{A_i}(\mathcal{M}_{A_i}, d, t)$ approaches \mathcal{M}_{A_i} . Otherwise if γ approaches 1 then $M_{A_i}(\mathcal{M}_{A_i}, d, t)$ approaches 0. The closer to 1 γ is, the harsher the attenuation will be. Moreover, for a fixed parameter γ , the longer the duration d and the greater the number of attempts t are, the lower the mark computed is. Therefore, function $M_{A_i}(\mathcal{M}_{A_i}, d, t)$ is fair.

Formula (3) has the same properties as Formulas (1) and (2) in terms of fading effect, flexibility and regular scoring.

Example 3. Let us consider the same learners of both Examples 1 and 2. We recall the training activity parameters are as follows: the full mark $\mathcal{M}_{A_i} = 5$ points, the optimal duration $\mathcal{D}_{A_i} = 300$ seconds, and the authorized number of tries is $\mathcal{T}_{A_i} = 3$. To illustrate the behavior of the model (formula (3)), we consider the same traces obtained for the learners $\{l_1, l_2, l_3, l_4\}$. Table 3 summarizes the results obtained for the four learners under the following values for γ : 0.2 (low importance level activity), 0.5 (medium importance level activity), 0.8 (high importance level activity).

We notice in Table 3 the double effect of penalties for learner l_4 due to the extra time taken to solve the problem and the number of tries which exceeds by twice the authorized number.

5 CONCLUSION

We have developed in this paper a generic model for evaluation of both learners and training strategies and

resources through performance indicators, based on duration, number of attempts, and their both combination, which are considered here as modeled traces. We have also illustrated the behavior of the model through some simple examples. The proposed model assesses in a flexible way the learner's performance and knowledge acquisition during a training activity. It can also capture some patterns of behavior of the learner during training sessions. In addition, the experts can configure in the model the importance of a training activity from skill and knowledge acquisition standpoint, or limit the model to perform a regular scoring by setting the attenuation constants to 0. This kind of evaluation allows, on the one hand, personalizing and adapting learner training strategy and resources, and personalizing assessment of the knowledge acquisition which helps learner's categorization and ranking, on the other hand.

The next step in our work is the implementation and test of the model we proposed in an online laboratory for SQL training, which is already developed in our research team. We need particularly to empirically compute the appropriate attenuation constants used in the model. We also plan to develop the adaption process combining the performance indicator proposed and semantics rules describing the actions and tasks to undertake to adapt learning strategies and resources to cope with pedagogical objectives the learner has to achieve in his/her training activities. An implementation of the extended approach is also planned for its validation through real world use cases.

REFERENCES

- Amelung, M., Krieger, K., and Rosner, D. (2011). E-assessment as a service. *Learning Technologies, IEEE Transactions on*, 4(2):162–174.
- Andrews, J. H. (1998). Testing using log file analysis: tools, methods, and issues. In *Automated Software Engineering, 1998. Proceedings. 13th IEEE International Conference on*, pages 157–166. IEEE.
- Burstein, J., Leacock, C., and Swartz, R. (2001). Automated evaluation of essays and short answers.
- Cohen, J. (1977). *Statistical power analysis for the behavioral sciences (rev.* Lawrence Erlbaum Associates, Inc.
- Crisp, G. (2009). Interactive e-assessment: moving beyond multiple-choice questions. *Centre for Learning and Professional Development. Adelaide: University of Adelaide*, 3:12–31.
- Crooks, T. J. (1988). The impact of classroom evaluation practices on students. *Review of educational research*, 58(4):438–481.
- Djouad, T., Settouti, L. S., Prié, Y., Reffay, C., and Mille,

- A. (2010). Un système à base de traces pour la modélisation et l'élaboration d'indicateurs d'activités éducatives individuelles et collectives. mise à l'épreuve sur moodle. *Mise à l'épreuve sur Moodle. Technique et Science Informatiques*.
- Guo, P. J. (2013). Online Python Tutor: Embeddable web-based program visualization for CS education. In *Proceedings of the 44th ACM Technical Symposium on Computer Science Education, SIGCSE '13*, pages 579–584, New York, NY, USA. ACM.
- Kozma, R. (2009). Transforming education: Assessing and teaching 21st century skills. *The transition to computer-based assessment*, 13.
- Kumar, R., Chung, G. K., Madni, A., and Roberts, B. (2015). First evaluation of the physics instantiation of a problem-solving-based online learning platform. In *Artificial Intelligence in Education*, pages 686–689. Springer.
- Laflaquiere, J., Settouti, L. S., Prié, Y., and Mille, A. (2006). Trace-based framework for experience management and engineering. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 1171–1178. Springer.
- Lebis, A., Lefevre, M., Guin, N., and Luengo, V. (2015). Capitaliser les processus d'analyses de traces d'apprentissage indépendamment des plates-formes d'analyses de traces. Technical report, LIG-LIRIS. ANR Project HUBBLELEARN.
- Martin, R. (2008). New possibilities and challenges for assessment through the use of technology. *Towards a research agenda on computer-based assessment*, page 6.
- Mille, A., Champin, P.-A., Cordier, A., Georgeon, O., and Lefevre, M. (2013). Trace-based reasoning-modeling interaction traces for reasoning on experiences. In *The 26th International FLAIRS Conference*, pages 1–15.
- Nicol, D. (2007). E-assessment by design: using multiple-choice tests to good effect. *Journal of Further and Higher Education*, 31(1):53–64.
- Nicol, D. J. and Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in higher education*, 31(2):199–218.
- Papamitsiou, Z. and Economides, A. A. (2015). Temporal learning analytics visualizations for increasing awareness during assessment. *RUSC. Universities and Knowledge Society Journal*, 12(3):129–147.
- Patelis, T. (2000). An overview of computer-based testing.
- Scheuermann, F. and Björnsson, J. (2009). The transition to computer-based assessment. *Luxembourg: Office for Official Publications of the European Communities*.
- Scorm (2006). *SCORM 2004 Handbook*. The e-Learning Consortium, Japan. Version 1.04.
- Settouti, L. S., Prié, Y., Champin, P.-A., Marty, J.-C., and Mille, A. (2009a). A trace-based systems framework: Models, languages and semantics.
- Settouti, L. S., Prié, Y., Cram, D., Champin, P.-A., and Mille, A. (2009b). A trace-based framework for supporting digital object memories. In *Workshops Proceedings of the 5th International Conference on Intelligent Environments, Barcelona, Spain, 19th of July, 2009*, pages 39–44.
- Settouti, L. S., Prié, Y., Marty, J., and Mille, A. (2009c). A trace-based system for technology-enhanced learning systems personalisation. In *The 9th IEEE International Conference on Advanced Learning Technologies, ICALT 2009, Riga, Latvia, July 15-17, 2009*, pages 93–97. IEEE Computer Society.
- Sireci, S. and Luecht, R. M. (2012). A review of models for computer-based testing.
- Thompson, N. and Wiess, D. (2009). Computerised and adaptive testing in educational assessment. *The transition to computer-based assessment. New approaches to skills assessment and implications for large-scale testing*, pages 127–133.
- Wandall, J. (2011). National tests in denmark—cat as a pedagogic tool. *Association of Test Publishers*, 12(1).
- Wilhelm, O. (2009). Issues in computerized ability measurement: Getting out of the jingle and jangle jungle. *The transition to computer-based assessment*, pages 145–150.
- Williamson, D. M., Xi, X., and Breyer, F. J. (2012). A framework for evaluation and use of automated scoring. *Educational Measurement: Issues and Practice*, 31(1):2–13.
- Yang, Y., Buckendahl, C. W., Juszkievicz, P. J., and Bhola, D. S. (2002). A review of strategies for validating computer-automated scoring. *Applied Measurement in Education*, 15(4):391–412.