Learning Analytics Should Analyse the Learning: Proposing a Generic Stealth Assessment Tool

Konstantinos Georgiadis
Faculty of Psychology and Educational Sciences
Open University of the Netherlands
Heerlen, The Netherlands
konstantinos.georgiadis@ou.nl

Giel van Lankveld
Faculty of Psychology and Educational Sciences
Open University of the Netherlands
Heerlen, The Netherlands
gielvanlankveld@gmail.com

Kiavash Bahreini
Faculty of Psychology and Educational Sciences
Open University of the Netherlands
Heerlen, The Netherlands
kiavash.bahreini@ou.nl

Wim Westera
Faculty of Psychology and Educational Sciences
Open University of the Netherlands
Heerlen, The Netherlands
wim.westera@ou.nl

***Abstract*— Stealth assessment could radically extend the scope and impact of learning analytics. Stealth assessment refers to the unobtrusive assessment of learners by exploiting emerging data from their digital traces in electronic learning environments through machine learning technologies. So far, stealth assessment has been studied extensively in serious games, but has not been widely applied, as it is a laborious and complex methodology for which no support tools are available. This study proposes a generic tool for the arrangement of stealth assessment to remove its current limitations and pave the road for its wider adoption. It describes the conceptual design of such a tool including its requirements regarding users, functions, and workflow. A prototype was implemented as a basic console application covering the tool's core requirements, including a Gaussian Naïve Bayes Network utility. Generated input files were used for testing and validating the approach. In a controlled test condition the stealth assessment classification accuracy was found to be inherently stable and high (typically above 92%). It is argued that the proposed approach could radically increase the applicability of stealth assessment in serious games and inform current learning analytics approaches with unobtrusive, more detailed and genuine assessments of learning.**

Keywords*— *learning analytics, stealth assessment, machine learning, serious games, generic tool

I. Introduction

Since the emergence of learning management systems, learning analytics have gained much attention among scholars and educators. Learning analytics in education is regarded as an excellent mean for the collection and analysis of student data to improve the quality and value of the learning experiences in schools and universities [1]. Most notably, the digital traces that students leave in their digital learning environments are considered to be a rich source of information for gaining new insights into educational practices and devising new ways to improve teaching and learning. A related term is educational data mining [2, 3], which is often used as a synonym of learning analytics [4]. Although learning analytics clearly fits in the topical worldwide trends of data science and web analytics, educators have been using educational mining methods for decades, be it that the data were mostly collected from paper-based student questionnaires rather than digital traces [5]. As to date learning management systems are the main source of data, learning analytics is mainly focusing on meso-level institutional indicators such as courses taken, speed, exams passed or failed. These are generally systemic indicators that provide detailed information about traffic and logistics, highly relevant for managing educational institutes. But learning analytics seldom includes data that are directly reflecting the micro-level of learning activities, e.g., students struggling with a specific concept, explanation or formula, which would be expressly informative regarding the very process of learning. Put differently, today's learning analytics is not so much about the analytics of learning! The semantic confusion underlying this situation is that the process of studying (logistics: enrolling in courses) is readily mistaken for the process of learning (cognition: augmenting knowledge and skills).

So far, learning management systems neglect (or are incapable of) micro-level tracking. By exception, educational video games (serious games) already allow for the detailed tracking of student interactions and decisions. Noticeably, serious games are in the spotlight of educational research [6] for their potential as vehicles to enable active learning in well-tailored problem contexts. As such, they are considered to be highly suited for the acquisition and retention of knowledge and skills (i.e. competencies) [7]. Because of the detailed user traces, true analytics of learning is well within reach in serious games. These may set the standards for learning analytics in future learning management systems that will certainly allow capturing detailed data of student actions.

Indeed, assessments based on detailed learner traces have gained increased attention from educational researchers, especially related to serious games [8, 9, 10]. This attention primarily stems from the need to fathom what is learned in these environments and accordingly provide formative feedback to the learners without the need for explicit testing [11]. Still, to achieve this, these assessments must rely on valid and reliable competency constructs [12], which are generally absent in current learning analytics.

A methodology which is considered to be apt for applying such formative assessments in serious games is stealth assessment (SA) [13]. SA is a methodology directly integrated in the gaming environment, which uses logged data from gameplay to classify learners' performance through machine learning (ML). Being unobtrusively embedded in games, SA reduces the saliency of the assessment process, which minimizes test anxiety and improves the validity of the assessment itself [14]. In contrast to traditional test items (e.g. self-report questionnaires, multiple-choice tests, etc.), SA can (a) benefit from the possibility to access high resolution data, hence allowing for more detailed and broad assessment of the

978-1-7281-1884-0/19/$31.00 ©2019 IEEE

learners (including soft skills and non-cognitive competencies), (b) allow the adaptation of the game's responses on the fly to meet the learner's personal needs, (c) provide rapid feedback during the learning process to optimize the learning, and (d) counter inherent issues in traditional tests such as social desirability effects .

Nonetheless, when it comes to applying SA various limitations are encountered. First, SA is a labour intensive, complex, and time-consuming process [14] that requires a broad range of expertise at each step of its implementation (e.g. in ML, programming, psychometrics, statistics, competency construct development, etc.). Second, SA requires access to source code of the digital learning environment game (viz. the game) and ML tools to create and train the required assessment models. Even if these requirements are fulfilled, the mapping of in-game behaviours to competency constructs remains a rather intuitive process. Shute and colleagues [15] describe this as an iterative process of brainstorming and pilot testing. Primarily due to these limitations, SA has not yet been widely adopted.

To overcome these issues, this study proposes the development of a generic tool for applying SA. It would help to lift the barriers of SA and accommodate its practical application in serious games. First, a description of SA along with an overview of its current applications in serious games is presented in Section 2. Next, limitations of SA drawn from relevant literature follow in Section 3. In Section 4 the requirements and conceptual design of a generic SA tool is given. Section 5 describes the running prototype along with preliminary tests, as a proof case of valid SAs. Section 6 discusses the findings and future plans.

II. Stealth Assessment

As explained before, SA is an unobtrusive evidence-based assessment methodology, which employs ML technologies to provide probabilistic reasoning about learners' performances in serious games. To achieve this, SA utilizes a conceptual framework for establishing relationships between observables from gameplay and competency constructs, which in turn translate to statistical models that ML algorithms can process. Specifically, SA combines two main ingredients: (a) the Evidence-Centered Design (ECD) [16, 17] and (b) a machine learning (ML) algorithm called Bayesian Network (BN) [18]. Below, both ingredients of SA are summarized and complemented with an overview of practical application of SA.

A. Evidence-Centered Design

ECD is a conceptual assessment framework consisting of three major elements: the competency model, the task model, and the evidence model. The competency model defines the construct that describes the underlying factors (i.e. facets or sub-competencies) constituting a competency. The task model describes a set of activities in the game that can elicit evidence relating to the competency. The evidence model describes the criteria of how learners' observed performances in the game link to both the competency and task model. The evidence model is described by two components: the evidence rules and the statistical model, respectively. The evidence rules cover the relationship between the tasks and the observed performances, while the statistical model defines the statistical relationships between observed performances and a competency construct. The latter constitute a latent variable model.

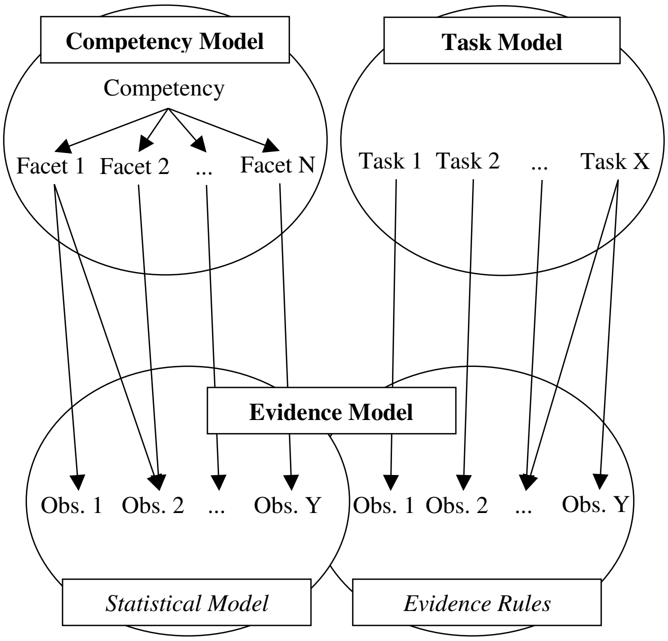

Fig. 1. A view of the Evidence-Centered Design.

Fig. 1 illustrates a view of the exemplary ECD model. It refers to a competency that is to be assessed. This competency, along with an N number of facets associated with it, constitutes the competency model. To elicit data for this competency model, an X number of in-game tasks are developed which form the task model. A Y number of observables (i.e. game variables) are then assigned to the tasks to form the evidence rules. Each task relates to one or more of these observables. The same Y number of observables also relates to the facets of the competency or the competency itself (if the competency is unidimensional) to form the statistical model.

B. Machine Learning in SA

ML is a field of artificial intelligence that enables the computer to learn from data. ML algorithms include both supervised and unsupervised learning algorithms. Supervised learning is the machine learning task of inferring a function from a labelled training dataset (i.e. an annotated dataset with classifications). Unsupervised learning is the machine learning task of inferring a function from an unlabeled training dataset (i.e. a non-annotated dataset without classifications).

SA applies ML to a latent variable model as defined by the statistical model to provide probabilistic reasoning over learners' observed performances. This means that the computer is trained based on a statistical model that uses player performance data to derive assessment outcome for the underlying competence. Originally, the use of BNs was considered for this purpose [13]. A BN is a supervised learning algorithm that can effectively handle evidence through its internal probability distribution function. So far, BNs have been proven to be robust in producing valid and reliable probability statements regarding the mastery of respective competencies. So far, BNs have been successfully applied in SA in qualitative physics [19], persistence [20], and problem-solving skills [21]. However, alternative supervised ML algorithms have also been proposed for SA, such as Decision Trees (DT), Neural Networks (NN), Logistic Regression (LR), Support Vector Machines (SVMs), and Deep Learning [22, 23].

C. *Current Applications*

Three different real-world SA applications are analyzed as an example to pinpoint open issues and practical limitations. The analysis reviews (1) the original approach by Shute and colleagues [21], (2) an alternative approach by Sabourin and colleagues [22], and (3) the DeepStealth approach by Min and colleagues [23]. An overview of these approaches is provided in Table I.

TABLE I. Overview Of The Three SA Applications

Approach	**Characteristics of the SA approaches**			
	ML Algorithm	***Validity***	***Reliability***	***Open Issues***
Shute et al. (2016)	Bayesian Network	Raven's ($r=0.40$, $p<0.01$), MicroDYN ($r=0.41$, $p<0.01$)	Internal ($\alpha=0.67$), External ($\alpha=0.43$)	Probable overfitting and poor alignment with the external measures.
Sabourin et al. (2013)	Naïve Bayesian Network, Neural Network, Logistic Regression, Support Vector Machines, Decision Tree	N/A	N/A	No information about the ML optimizations, the validity and the reliability of the competency construct. Data from external measures was used in the ML process.
Min et al. (2015)	Deep Learning	N/A	N/A	No information about the validity and the reliability of the competency construct.

In the original approach by Shute et al., a serious game called Use Your Brainz (mod of Plants vs. Zombies 2) was used to assess the problem-solving skills of learners. Shute et al. conducted a literature review to identify and develop the construct of the competency model for problem-solving skills. An iterative process of brainstorming and pilot testing was followed for detecting a list of relevant observables to establish the statistical model. Cronbach's α revealed a good internal reliability regarding the construct, but a less than ideal reliability relative to an external measure (MicroDYN). The validity of the construct was also examined, which revealed significant but weak positive correlations with external measures (Raven's Progressive Matrices and MicroDYN). Shute suggested that the BN probably suffered from overfitting due to having a small sample size (N=55).

In Sabourin's et al. approach, a serious game called CRYSTAL ISLAND was used to assess the learners' self-regulation behaviour. In this case, no clear description of an explicit ECD model was provided, although reference to SA exists in relevant work [24]. Thus, information about the validity and reliability of a competency construct representing self-regulation was not provided. Instead, a set of features was selected in a somewhat intuitive manner with no clear mappings to a valid and reliable competency construct for generating the statistical model. These features included observables not only from the game, but also items from several questionnaires (Achievement Goals Questionnaire, Big Five Inventory, Cognitive Emotion Regulation Questionnaire, and Demographics). This approach is in contrast to the principal notion that the ML algorithms in SA should only contain evidence related to actions bound to the gameplay. Nevertheless, this approach is a useful showcase of applying different ML algorithms for evidence-based assessment in serious games: Five different supervised learning algorithms were used including Naïve BN, NN, LR, SVM, and DT. Based on a comparative analysis, the authors suggested that the DT and LR algorithms outperform the rest in terms of predictive accuracy. However, no details were provided on how these ML algorithms were tuned.

As for the DeepStealth approach, a game-based learning environment, called ENGAGE, was used to assess learners' computational thinking skills. More specifically, learners were asked to represent digital data in binary sequences. A simple ECD model was devised, including a unidimensional competency model (i.e. a competency without facets), four observables, and sixteen tasks. The validity and reliability of the selected observables was not examined. The novelty of this approach lies in the use of Deep Learning. The authors provided a detailed description of the Deep Learning optimizations. The results showed that Deep Learning outperforms Naïve BN and SVM in terms of predictive accuracy.

In all three approaches, apart from the accuracy, no other ML performance measures [25] were provided, such as kappa statistic, mean absolute error, root mean squared error, relative absolute error, and root relative squared error.

III. Current Barriers To Applying Stealth Assessment

This chapter discusses principal limitations in applying SA in serious games as identified both from the analysis on current applications and existing literature. The limitations are mainly expressed in terms of complexity and laboriousness. Other practical limitations are also briefly discussed.

A. *Complexity*

The complexity of applying SA in serious games primarily relates to the expertise that is needed with respect to three perspectives.

Firstly, a great amount of expertise is needed from an educational and psychometric perspective. This includes instructional design knowledge required to properly design the in-game tasks in accordance with the competency to be assessed, as well as knowledge of the learning material in order to set the game content. It also encompasses knowledge about the underlying competency constructs (e.g. from psychometrics or relevant literature if available), and of the evidence that maps to the competency constructs (to set the statistical model) and defines the mastery levels (to label or mark data records for supervised learning algorithms).

Secondly, considerable technical expertise is required, such as knowledge of game design, of game development, and ML expertise. Game design knowledge is needed to implement the in-game tasks (as defined by the instructional design) by steering the graphical user interface, the narrative, the levels, the audio, etc. This is crucial for eliciting proper evidence with minimum noise introduced from irrelevant covariates that may affect the learning process. In addition, game development expertise mainly refers to knowledge of programming languages and game development tools (e.g. game engines) that allow modifying or developing from scratch a serious game that suits the learning goals. ML expertise means knowledge on how to properly implement ML algorithms by taking into account their representation, evaluation, and optimization aspects [26]. It is crucial to underline the importance of transparency regarding those aspects as to allow replication, ensure scientific integrity, and secure pedagogical value.

Thirdly, expertise is needed from a statistics perspective. This includes knowledge of statistical methods such as correlation and factor analysis in order to be able to develop, validate, and verify competency constructs. It is essential to underline the importance of validating and verifying the competency constructs as to avoid threats such as construct underrepresentation and construct irrelevant variance [27].

B. *Laboriousness*

Apart from being complex, the process of applying SA in serious games is also quite laborious. SA is originally defined [13] as an assessment methodology that is directly woven into the game environment fabric. As a result, all existing applications so far have been developed in a hardcoded manner, meaning embedded in the game source code itself. Thus, every time SA is to be applied, it would require software development and validation from scratch. This comes at a great cost, as multiple steps are involved for setting the entire workflow, introducing unattractive routine works (e.g. manual labelling of a training dataset) that can be prone to mistakes (e.g. mislabeling). Even if one manages to successfully apply SA, reconfiguring the system to fit new or updated assessment needs requires additional scripting and manual tweaking on all fronts (e.g. game design and development, construct development, validation processes, ML optimizations, etc.). The laboriousness of applying SA has contributed to the development of weak business cases so far.

C. *Practical Limitations*

Noticeably, applying SA does not only meet limitations due to complexity and laboriousness but also due to some dead ends. For example, an ECD model is often missing or is difficult to define (e.g. for soft skills). Even if one manages to develop a competency construct (for which no psychometrics or literature exists) following the construct definition process [12], existing games (even after modifying them) may not provide for all the observables mapping to it, hence rendering the statistical model partly deficient [14].

IV. Specifying A Generic Stealth Assessment Instrument

To tackle the limitations of SA and accommodate its practical application, this study introduces a conceptual design of a generic SA tool. Within this framework, SA is described as a stand-alone software tool that is detached from the game source code. In this way, the need for game development expertise to apply SA could be eliminated, while log files from any serious game could be used (provided that certain format standards are respected) without the need for additional manual labour (e.g. coding).

We argue, that implementing SA as a stand-alone software tool is feasible due to the inherent generic nature of the SA methodology. That is because its main ingredients, the ECD and the ML algorithms, can support generic construct representations. In detail, ECD can describe any competency construct within the competency model, and ML algorithms can adjust their representation to match any statistical model regardless shape or size. In addition, the number of tasks or observables declared in ECD is not restricted. The same holds for the relationships that can be expressed within ECD. Instead, they could be defined on a case-by-case basis following particular assessment needs.

A high order view of the proposed generic framework for the software prototype compared to the original framework [13] is presented in Fig. 2. In the original framework, SA is hard-coded directly in the game. Thus, emerging data logs are directly processed, while feedback is provided on the fly. On the contrary, the proposed generic framework detaches SA from the game itself as well as from any direct data logging or feedback process. As a result, the amount of expertise needed is drastically minimized, while dependencies to hardcoded solutions for data logging and feedback are avoided. Also, external tools capable of producing learning analytics (for various stakeholders e.g. institutions, assessors, learners, etc.), and adaptation can benefit from the outputs of the generic SA tool. Nevertheless, technical integration of the tool functionality within the game, to allow for instant assessments (and feedbacks) during game play, is still possible, when the tool would be converted into a software component compliant with the RAGE client-side software architecture of game components [28] to ensure its interoperability and portability.

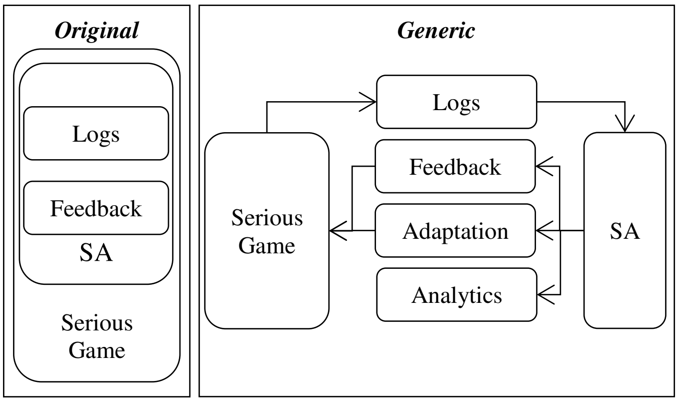

Fig. 2. A view of the original (left) and the generic (right) SA frameworks.

A. Functional Requirements

The main focus of the generic tool is to provide functionalities that reduce the amount of expertise and manual labour needed to apply SA. From an educational and psychometric perspective, the tool should be able to assist its users to easily define their assessment optimizations regarding both ECD, game logs, and ML. Therefore, three functional requirements are set for optimizing the assessment:

- Data import: The tool should allow for importing data from log files of various formats and standards.
- Setting the ECD: The tool should allow setting the ECD, that is defining the competency constructs and the statistical models that link the elements to observables in the game.
- ML optimization: The tool should allow for declaring desirable ML optimizations (e.g. the ML algorithm type and its inner options). At this point, it is important to note that except from supervised ML algorithms, unsupervised ML algorithms should also be available (in contrast to current applications) to accommodate the use of unlabeled datasets.

To minimize the need for ML expertise, the tool should also provide access to automated built-in ML functions. This yields three additional functional requirements:

- Supporting multiple ML algorithms: The tool should be able to apply different ML algorithms that automatically adjust their representation according to the provided ECD.
- ML execution: The tool should allow the automatic execution of the selected ML algorithms.
- Outputs: The tool should produce detailed output about both students' performances and ML algorithms' performances for evaluation purposes.

B. User Requirements

The users of the generic software tool should not necessarily acquire detailed SA expertise (technical, educational, etc.). For example, the users could be game developers, educators, or any other candidate assessor, possessing just enough knowledge to benefit from the tool at an operational level.

To further reduce the complexities that SA poses and to assist the users of the tool, a set of support functions could be developed:

- Verification and validation of competency constructs: Such validation could be covered by automatic correlation analysis between the outputs from an external measure (e.g. psychometric test, expert ratings, etc.) and the assessment outputs of the tool.
- Generating an ECD: The users of the tool may not always have an ECD, competency model or statistical model available, or cannot define such models, for instance if no psychometrics or literature on the topic exist. A set of support functions could be readily available to assist them. For example, if the users have access to raw data from an external assessment measure (which means that a competency model is also available) and data from a log file, but no knowledge of the statistical model, then a support function could attempt its automatic generation by applying a correlation analysis approach. If the users have only access to a log file but no knowledge of the competency model, then a support function could be available to attempt its automatic generation through a factor analysis approach. A series of forthcoming empirical studies will detail and evaluate the aforementioned support functions of the generic tool. Obviously, these support functions would raise the need to extend the functional requirements of the generic tool. For example, the tool should allow the user to import data from an external measure.
- User guidance: Several widgets (e.g. tutorial, built-in help menu, user manual, installation guide, etc.) should be developed to enhance the usability of the proposed tool and enforce the guidance of the users.

C. Technical Design

To realize these functions, we propose a technical design of the generic tool that includes two main subsystems: (a) a software wizard to tailor the procedure of setting the assessment optimizations, and (b) a machine learning software component to serve the ML functions. A view of the proposed technical design is illustrated in Fig. 3.

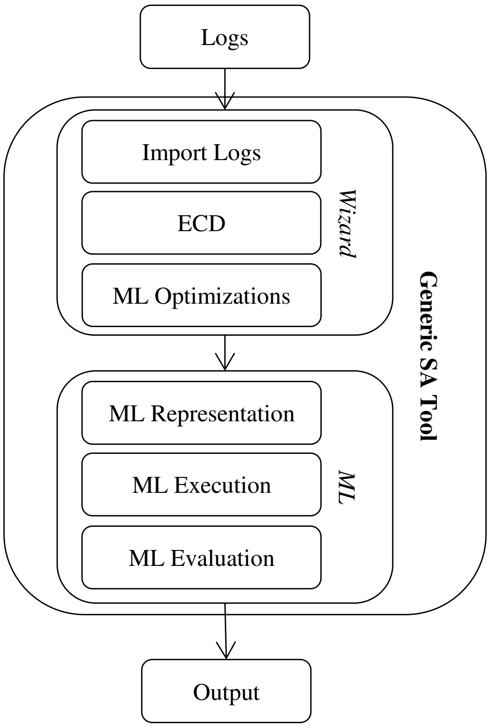

Fig. 3. View of the generic tool for SA in serious games.

D. Workflow Design

Fig. 4 illustrates a view of the SA tool's workflow based on the requirements and including the aforementioned support functions.

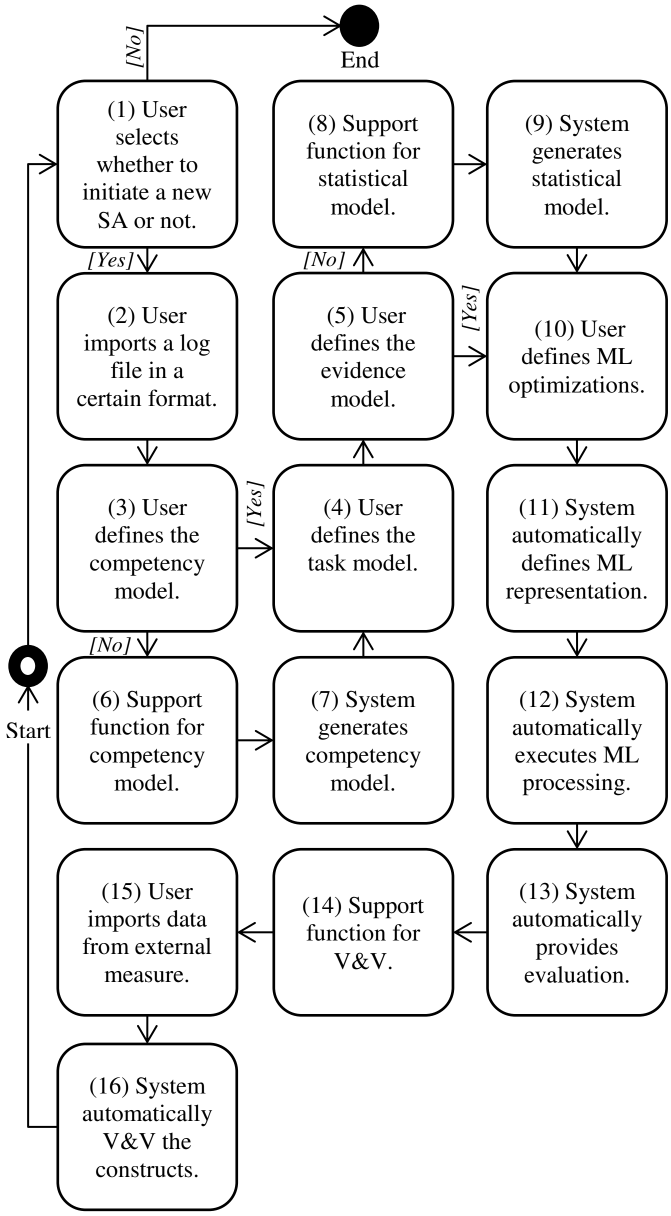
End
[No]
(1) User selects whether to initiate a new SA or not.
[Yes]
(2) User imports a log file in a certain format.
(3) User defines the competency model.
[Yes]
[No]
(4) User defines the task model.
(5) User defines the evidence model.
[No]
[Yes]
(6) Support function for competency model.
(7) System generates competency model.
(8) Support function for statistical model.
(9) System generates statistical model.
(10) User defines ML optimizations.
(11) System automatically defines ML representation.
(12) System automatically executes ML processing.
(13) System automatically provides evaluation.
(14) Support function for V&V.
(15) User imports data from external measure.
(16) System automatically V&V the constructs.
Start

Fig. 4. A view of the generic tool's workflow.

At start, the user selects whether to initiate a new SA or not (1). If not, then the workflow terminates. Else, the user can proceed to the use of the software wizard and import a log file (2). Next, the user can set the ECD, that is, the user can define the competency model (3), the task model (4), and the evidence model (5), respectively. It is important to notice here, that while defining the competency model and the statistical model (within the evidence model) is critical for the assessment, it is only optional to define the task model and the evidence rules. This is because the latter are relevant for developing the instructional design and the content of the serious game rather than for performing the actual assessment process. If the user encounters problems in defining the competency model, then a support function is available (6). This support function automatically performs a factor analysis approach on the data from the imported log file to attempt the generation of a competency model if appropriate latent variables are found (7). If the user encounters issues in defining the statistical model, then another support function is available (8). This support function requires the import of raw data from an external measure in a certain format. When imported, the system automatically performs a correlation analysis between the data from the external measure and the log files to attempt the generation of a statistical model if significantly strong correlations are found (9). When the ECD definition is done, the user can declare the desired ML optimizations (10). Thereafter, the system automatically defines the ML representation (11), executes the ML processing (12), and provides output regarding both the students and ML performances for evaluation purposes (13). Finally, the user can select whether or not to use a support function to verify and validate (V&V) the used competency constructs (14). If not, the system boots to its initial state. Else, the user imports data from an external measure (15). In this case the data is not raw as in (8), but it is the final classification (i.e. labelling) of the students' performances. In turn, the system automatically performs a correlation analysis approach between the output of the tool (student performances) and output from an external measure (16).

V. Preliminary Results

A. *Technical Implementation*

A software prototype for the proposed SA tool was implemented as a stand-alone client-side console application in C# using the .NET framework. Essentially, it is an early version of the envisioned tool only addressing core requirements. Consequently, it goes with some practical limitations, be it not principal ones. For instance, so far the tool can only handle log files from a spreadsheet format. Later versions may include handling data from other formats as well, such as .XML or .JSON files. Also, it is assumed that the log file only contains numerical data. Ordinal, categorical or Boolean data should be converted to numeric beforehand. The numerical values of the data should all be positive and must be ordered so that they reflect ascending scores. This requires simple data transformations that can be easily added at a later stage. In accordance with the xAPI tracking standard, each learner's record must always be contained in one line and never span multiple lines in the log file. İncomplete records as well as outliers should be removed from the data. Labelled data (if it exists) should be included in the log file and be stated separately (i.e. in different column) for each facet and competency declared in the ECD model. The declaration of ECD models is now provisionally covered by a configuration file, as no specific editor was provided at this stage. Similarly, the ML optimizations are currently set within the source code instead of allowing the user to control the ML parameters externally. Utilities for removing these limitations can be easily added, once sufficient proof cases of the approach have become available. In addition, various external libraries were used, e.g. the EPPlus library was used to enable importing data from spreadsheet files, and the Accord.NET framework was used for the machine learning functions. Altogether, the prototype allows for executing the very core of SA, as it 1) allows for defining competency models, it 2) allows for defining statistical models, which is the latent variable model that covers the statistical relationships between observed behaviours and competency constructs, it 3) handles data inputs representing game observables (log files), it 4) arranges the ML representation, execution and evaluation, and 5) it outputs the assessment results, which then can be compared with reference data (labels).

B. *Operational Validation*

Apart from extensive technical testing, specific tests were carried out to assess the overall functioning of the tool and to evaluate the outcomes produced. To this end, various R scripts were used to generate simulation data to represent game log files. Before turning to real-world game data, which are likely to suffer from incompleteness, unknown biases and other imperfections, well-controlled large-scale reference data were deemed essential for principal validation. Basically, the following ad hoc procedure was used as a preliminary test. Two abstract competence constructs were defined, composed of two facets and three facets, respectively. A set of 4 observable variables was chosen to represent the two-facet competency model; 7 observables were defined to link to the three-facet model. In accordance with these models, different normal distributions were used to randomly sample user data to represent the log files. Labelling of the randomly generated data, which was needed to train the ML algorithm, was established by applying a k-means clustering algorithm, assigning each data point to the cluster with the nearest mean (i.e. centroid) value. In this study, the number of clusters was set to three (k=3), reflecting three separate performance classes (low, medium, high) to be used by the ML classifier. Each run was composed of 10,000 full user records. A split rule was applied of 66% the training the ML algorithm and 34% for testing the outputs. For different sampling conditions a set of 80 runs was executed to enhance statistical power. Fig. 5 shows exemplary SA output of the cumulative classification accuracy for a Gaussian Naïve Bayesian Network (GNBN).

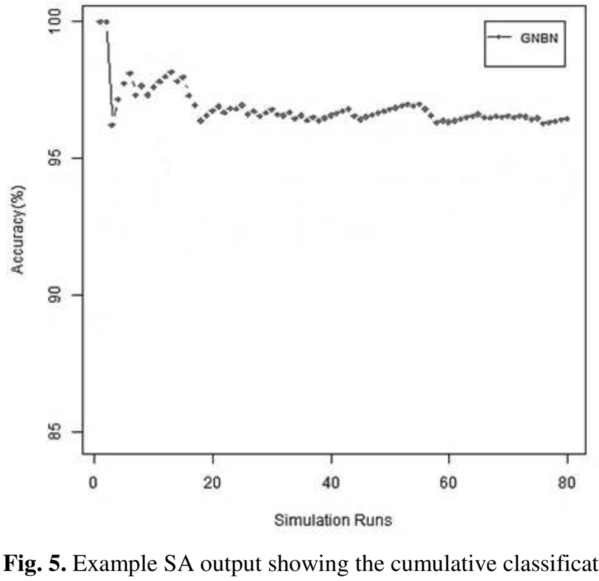

Fig. 5. Example SA output showing the cumulative classification accuracy of a Gaussian Naïve Bayesian Network.

While setting a confidence level of 0.99, the confidence intervals of the accuracies after 80 runs ranged from 92,5% to 94.5% and from 95.2% to 96.8%, for the two competency models respectively.

VI. Discussion And Conclution

This result presented above demonstrates the feasibility of the approach. First, the classification accuracy of the GNBN for the two-facet competency model is overall high, typically above 95%; while for the three-facet competency it is above 92% Second, at increasing number of runs, the cumulative classification accuracy of the GNBN converges to a stable level, leaving a very small confidence interval.

Technically, the prototype fulfils the principal requirements of being able to handle various competency constructs regardless of shape and size, the statistical model relating observed behaviours and the competency constructs, and various data logs containing any type of numerical data (discrete or continuous). The proposed conceptual design and the associated SA prototype have been proven adequate for the generalized application of SA. Although, so far preliminary and narrow, the validation of the SA tool greatly enhances the potential of learning analytics by opening up the route toward true analytics of learning, as it allows to process and assess performances from micro-level traces of learners engaged in any (digital) learning environment.

Four separate strands of follow-up activities are foreseen to continue this research and to eventually end up with a practical and viable tool that could be widely adopted and used in teaching and training. First, log data from real world serious games (or similar) should be used to further collect empirical evidence of its ecological validity. The imperfections of real-world datasets may readily affect the quality of assessment and yield reduced accuracies that – worst case - would be unacceptable for practical application. Also, real-world games may not always go with datasets that are large enough to train the ML models appropriately. Second, having the tool available as a generic implementation of the SA methodology opens up ample opportunities for investigating the robustness and applicability of the SA methodology itself. As has been the case in this study, randomly generated datasets can be used for this. A follow-up study along these lines will be reported elsewhere, including the systematic comparison of different ML algorithms, both parametric and non-parametric. Third, for easy adoption and application of the tool by educators or game developers, current console application should be extended with a simple and practicable user-interface, highlighting the core steps of the procedure, including a (competence) editor, data conversion options, cleaning and alignment tools, prerequisite violation checks, and user guidance and support functions. Although these topics do not pose any principal problems, it is well-recognized that these are most critical factors for the successful application of advanced ICT in educational practice. Fourth, since SA is readily viewed as the icing on the cake of learning analytics, it should preferably comply with relevant learning analytics interoperability standards and specifications, such as xAPI and LTI. Moreover, the tool should be made compliant with the RAGE architecture of client-side applied gaming components [28] to ensure its easy integration with a variety of relevant software tools and game engines.

Altogether, this study presented first results from a SA prototype that could lift the current barriers to SA and enhance its applicability in serious games and other digital learning environments. Tools like this are good candidates of becoming part of current learning analytics suites as they have the capability of providing detailed assessment of learners. Learning analytics would thus gain value beyond the level of overall institutional system performance, and seize the opportunities of analyzing detailed student interactions with learning content to optimize the individual's learning. Most importantly, this progress does not alter the fact that awareness should be raised regarding the ethical and social aspects of unobtrusive assessment methodologies (e.g. Big Brother) such as SA. Thus the importance of requesting the learners' consent before applying these should be emphasized.

Acknowledgment

This research was funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 644187, the RAGE project (www.rageproject.eu).

References

[1] Siemens, George, and Phil Long. "Penetrating the fog: Analytics in learning and education." EDUCAUSE review 46.5 (2011): 30.

[2] Baker, Ryan SJD, and Kalina Yacef. "The state of educational data mining in 2009: A review and future visions." JEDM| Journal of Educational Data Mining 1.1 (2009): 3-17.

[3] Romero, Cristobal, and Sebastian Ventura. "Educational data mining: A survey from 1995 to 2005." Expert systems with applications 33.1 (2007): 135-146.

[4] Zouaq, Amal, Srecko Joksimovic, and Dragan Gasevic. "Ontology Learning to Analyze Research Trends in Learning Analytics Publications." LAK (Data Challenge). 2013.

[5] Martin, Taylor, and Bruce Sherin. "Learning analytics and computational techniques for detecting and evaluating patterns in learning: An introduction to the special issue." Journal of the Learning Sciences 22.4 (2013): 511-520.

[6] Gee, J. P. "What video games have to teach us about learning and literacy". Macmillan. 2014.

[7] Wouters, Pieter, et al. "A meta-analysis of the cognitive and motivational effects of serious games." Journal of educational psychology 105.2 (2013): 249.

[8] Shute, Valerie J., Eric G. Hansen, and Russell G. Almond. "You can't fatten A hog by weighing It–Or can you? evaluating an assessment for learning system called ACED." International Journal of Artificial Intelligence in Education 18.4 (2008): 289-316.

[9] Shute, Valerie J., and Diego Zapata-Rivera. "Educational measurement and intelligent systems." of the International Encyclopedia of Education. Oxford, UK: Elsevier Publishers (2010).

[10] Shute, Valerie J., and Gregory R. Moore. "Consistency and validity in game-based stealth assessment." Technology enhanced innovative assessment: Development, modeling, and scoring from an interdisciplinary perspective 296 (2017).

[11] Shute, Valerie J. "Focus on formative feedback." Review of educational research 78.1 (2008): 153-189.

[12] Belland, Brian R. "The role of construct definition in the creation of formative assessments in game-based learning." Assessment in game-based learning. Springer, New York, NY, 2012. 29-42.

[13] Shute, Valerie J. "Stealth assessment in computer-based games to support learning." Computer games and instruction 55.2 (2011): 503-524.

[14] Moore, Gregory R., and Valerie J. Shute. "Improving learning through stealth assessment of conscientiousness." Handbook on Digital Learning for K-12 Schools. Springer, Cham, 2017. 355-368.

[15] Shute, Valerie J., et al. "Measuring problem solving skills via stealth assessment in an engaging video game." Computers in Human Behavior 63 (2016): 106-117.

[16] Mislevy, Robert J., Linda S. Steinberg, and Russell G. Almond. "Focus article: On the structure of educational assessments." Measurement: Interdisciplinary research and perspectives 1.1 (2003): 3-62.

[17] Mislevy, Robert J. "Evidence-Centered Design for Simulation-Based Assessment. CRESST Report 800." National Center for Research on Evaluation, Standards, and Student Testing (CRESST) (2011).

[18] Pearl, Judea. "Probabilistic reasoning in intelligent systems: networks of plausible inference." Elsevier, 2014.

[19] Shute, Valerie J., Matthew Ventura, and Yoon Jeon Kim. "Assessment and learning of qualitative physics in newton's playground." The Journal of Educational Research 106.6 (2013): 423-430.

[20] Ventura, Matthew, Valerie Shute, and Matthew Small. "Assessing persistence in educational games." Design recommendations for adaptive intelligent tutoring systems: Learner modeling 2 (2014): 93-101.

[21] Shute, Valerie J., et al. "Measuring problem solving skills via stealth assessment in an engaging video game." Computers in Human Behavior 63 (2016): 106-117.

[22] Sabourin, Jennifer L., et al. "Understanding and predicting student self-regulated learning strategies in game-based learning environments." International Journal of Artificial Intelligence in Education 23.1-4 (2013): 94-114.

[23] Min, Wookhee, et al. "DeepStealth: leveraging deep learning models for stealth assessment in game-based learning environments." International Conference on Artificial Intelligence in Education. Springer, Cham, 2015.

[24] Sabourin, Jennifer Lynne. "Stealth Assessment of Self-Regulated Learning in Game-Based Learning Environments." (2013).

[25] Correa, M., Bielza, C., & Pamies-Teixeira, J. (2009). "Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process." Expert systems with applications, 36(3), 7270-7279.

[26] Domingos, Pedro M. "A few useful things to know about machine learning." Commun. acm 55.10 (2012): 78-87.

[27] Messick, Samuel. "Validity of psychological assessment: Validation of inferences from persons' responses and performances as scientific inquiry into score meaning." American psychologist 50.9 (1995): 741.

[28] Van der Vegt, Wim, et al. "RAGE architecture for reusable serious gaming technology components." International Journal of Computer Games Technology 2016 (2016): 3.