Capitalisation of Analysis Processes : Enabling Reproducibility, Openess and Adaptability thanks to Narration

Alexis Lebis, Marie Lefevre, Vanda Luengo, Nathalie Guin

Capitalisation of Analysis Processes: Enabling Reproducibility, Openness and Adaptability thanks to Narration

Alexis Lebis
Sorbonne Universités, UPMC Univ Paris 06
CNRS UMR 7606, LIP6, Paris, France
alexis.lebis@lip6.fr

Marie Lefevre
Univ Lyon, Université Lyon 1, CNRS UMR 5205, LIRIS
Lyon, France
marie.lefevre@univ-lyon1.fr

Vanda Luengo
Sorbonne Universités, UPMC Univ Paris 06
CNRS UMR 7606, LIP6, Paris, France
vanda.luengo@lip6.fr

Nathalie Guin
Univ Lyon, Université Lyon 1, CNRS UMR 5205, LIRIS
Lyon, France
nathalie.guin@univ-lyon1.fr

ABSTRACT

Analysis processes of learning traces, used to gain important pedagogical insights, are yet to be easily shared and reused. They face what is commonly called a reproducibility crisis. From our observations, we identify two important factors that may be the cause of this crisis: technical constraints due to runnable necessities, and context dependencies. Moreover, the meaning of the reproducibility itself is ambiguous and a source of misunderstanding. In this paper, we present an ontological framework dedicated to taking full advantage of already implemented educational analyses. This framework shifts the actual paradigm of analysis processes by representing them from a narrative point of view, instead of a technical one. This enables a formal description of analysis processes with high-level concepts. We show how this description is performed, and how it can help analysts. The goal is to empower both expert and non-expert analysis stakeholders with the possibility to be involved in the elaboration of analysis processes and their reuse in different contexts, by improving both human and machine understanding of these analyses. This possibility is known as the capitalisation of analysis processes of learning traces.

CCS CONCEPTS

• **Information systems** → **Data analytics**; • **Computing methodologies** → **Knowledge representation and reasoning**; • **Applied computing** → *E-learning*;

KEYWORDS

Learning analytics, analysis processes of learning traces, ontology, context, reproducibility, reuse, adaptability, openness, capitalization.

ACM Reference Format:
Alexis Lebis, Marie Lefevre, Vanda Luengo, and Nathalie Guin. 2018. Capitalisation of Analysis Processes: Enabling Reproducibility, Openness and Adaptability thanks to Narration. In *LAK '18: International Conference on Learning Analytics and Knowledge, March 7–9, 2018, Sydney, NSW, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3170358.3170408

LAK '18, March 7–9, 2018, Sydney, NSW, Australia

ACM ISBN 978-1-4503-6400-3/18/03…$15.00
https://doi.org/10.1145/3170358.3170408

1 INTRODUCTION

In Learning Analytics, analyses are used to produce useful pedagogical information [13]. From a computer perspective, analyses are concretised using analysis processes. These processes are a sequence of identifiable and reusable operations within analysis tools, named operators [20]. However, analysis processes are hardly reproducible.

There is a growing concern regarding the lack of scientific reproducibility and its effect on the credibility of results and the validity of methodologies used. Indeed, this reproducibility is an important driver in scientific research [2]. Paradoxically, the *reproducibility* term is not coined with a unified definition and has several meanings, depending on the considered field. For example, in computer science, reproducibility is strongly related to the openness of both computer code and data, in order to repeat analyses with the same initial data [22]. In section 3, we propose a definition of reproducibility to clearly identify the elements involved.

The issue of reproducibility for analysis processes can be explained because the later combine various techniques, such as statistical and data mining ones [27], also confronted with a reproducibility issue [5]. Indeed, analysis processes have strong dependencies on implementation contexts, specificities of data used (e.g. formalism), and technical specificities (especially regarding analysis tools) [19]. Therefore, these dependencies make the whole procedure of reusing and sharing analysis processes difficult, if not irrelevant, and scarcely comprehensible [10].

Nevertheless, providing the TEL community with the possibility to reproduce existing analysis processes in other contexts would be a major breakthrough. It would enable us to envisage an unified and understandable ecosystem of analysis processes. This ecosystem, if open, could also become a driver within the community for co-constructed analyses, thus making it possible to reuse and adapt such analyses. This is what we mean by the *capitalisation* of analysis processes. We present in this paper our work related to the capitalisation of analysis processes of learning traces. In section 3, we propose a formalisation of capitalisation and compare it to reproducibility.

We propose a paradigm shift for the formalisation of analysis processes, achieved by a narration of these analysis processes. We define the narration as the representation of analyses and their contexts with high-level and structured concepts, setting specificities of technical tools aside. This narration enables the description of the analysis elements (e.g. descriptive, contextual, relational), while avoiding as far as possible biases related to implementation constraints. These described and abstracted analysis processes can be capitalised, as presented by our first experimental results in section 6. In section 4, we present our ontological framework designed to describe such narrated analysis processes. This framework is designed to be used by the analysis designers (e.g. statistician, analyst, researcher). We then present an implementation of our framework and discuss the impact of the narration on existing analysis processes in section 5.

2 RELATED WORKS

Several research studies tried to define analysis processes, leading to an improvement of their theoretical and conceptual grounds. One result of these studies is that analysis processes can be considered as a succession of configured operations. These operations modify the state of their input data in order to produce relevant information as output [18, 20]. In addition, analysis processes, or a sub-part, can be reused in other analysis processes [1], thus providing evidence of the importance of the capitalisation. Moreover, definitions of analysis life cycle have been produced by these studies [1]. These definitions introduce interesting nomenclatures, especially regarding preprocessing, analysis and post-processing steps [14, 25].

A sharing effort exists within the community, primarily concerning the data used in analysis processes. Two main approaches should be noted. The first concerns standardisation of the content used inside educational systems themselves [24], while the second approach proposes data formalisms and specifications, as with the well-known Datashop [17]. While the former potentially limits diversities of educational context, the latter often implies format constraints over data. However, both contribute to the reinforcement of the interoperability possibilities of TEL systems. These approaches also indirectly provide partial responses about interoperability between analysis tools [11]. Moreover, some of these formalisms use dedicated semantic vocabularies for expressing pedagogical data and activities, such as xAPI[1]. The information conveyed by such semantic vocabularies provides prospects about reuse and understanding of analysis processes [9].

In addition to these works on data sharing, efforts have also been made to share analysis processes. Although less common, they are worth noting. Some works envisage general building methods concerning analyses [7]. However, they still have to deal with the entanglement between data, context and technical specificities. Thus, we believe that they are not suited for capitalisation. Another approach consists of an online inventory designed for exploratory approaches. This lists the available analysis processes in a specific analysis tool [20]. However, similar to the above approach, this approach does not solve understandability and reuse issues concerning analysis processes and their related operators [26]. Indeed, technical context constraints alter reuse of such processes inside analysis tools. Analysis contexts (e.g. pedagogical system) are also involved in these issues. The model developed by Chatti & al. [8] gives good prospects regarding consideration of the pedagogical context, reasons, and technical resolutions of analyses. However, consideration of these contexts is not clearly established in works related to analysis processes within the TEL community. The main reason is that analysis tools allow the design of analysis processes only from a computational perspective.

Interesting approaches are introduced by studies and works about workflows. Workflows are used in several fields to represent a variety of processes (e.g. biology). The flow of analyses is represented and is supposed to be understandable per se. Therefore, runnable workflows were considered for a time as a solution for reproducibility of analysis processes. Moreover, some works are concerned with an open science perspective. These works require additional information as proof of analysis validity, like experimental protocols or supplementary resources [12]. However, as noted by Belhajjame & al. [3], workflows break off, due to their sensitiveness to technical constraints and their poor adaptability possibilities. This is imputable to the computational prerequisite of workflows and the lack of descriptions attached to the elements involved (e.g. operators). These issues hinder the reuse and adaptability of workflows. To reduce such constraints, works concerning the semantic descriptions of process components can be considered, like wf4ever [23]. These constitute an excellent means of enhancing reuse and sharing of processes, as shown by Bowers & al. [6]. Thus, these works introduce new ways of considering capitalisation of analysis processes of learning traces using TEL specificities, and provide new ways to involve TEL actors.

To our knowledge, only one work [19] emancipates analysis processes from the computational aspect. This computational dependency generates considerable technical constraints, thus affecting the concepts conveyed by these analyses [4, 19]. Moreover, it also hinders the description of these analyses. Actual description possibilities are often plain text, leading to ambiguous understanding. Therefore, analysis choices cannot be clearly expressed or discussed, which has an impact on analysis adaptability. This is why we propose a new paradigm to formalise analysis processes, based on a narrative approach, instead of using current designed formalisms based on computational requirements.

3 ADDRESSING THE POLYSEMY OF REPRODUCIBILITY AND CAPITALISATION

Several works show that the meaning of reproducibility itself is vague and greatly dependent on the field where the term is used [16, 22]. Therefore, capitalisation of analysis processes, as understood in this paper, cannot be considered without proper formalisation of reproducibility. In this section, we propose to define both capitalisation and the reproducibility terms.

According to various works such as the VIM (International Vocabulary of Metrology) or ACM (Association for Computing Machinery) recommendation concerning reproduction, several dimensions in reproducibility can be noted [15, 16]. Two recurrent dimensions are (1) whether or not a material is used by persons who initially designed it, and (2) whether or not the setup for use of this material is identical to the initial one. Less often, a third dimension in reproducibility can nevertheless be considered. This concerns the

[1] https://experienceapi.com/

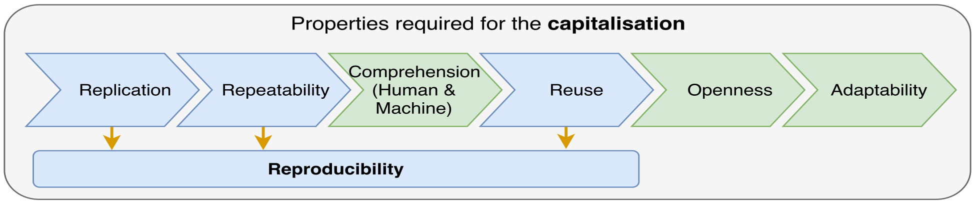
Properties required for the **capitalisation**
Replication
Repeatability
Comprehension (Human & Machine)
Reuse
Openness
Adaptability
Reproducibility

Figure 1: Illustration of the properties required for capitalising analysis processes, organised hierarchically depending on their dependencies. The hierarchy means that a property requires the property directly to its left in order to be consistent.

similarity of resources (e.g. data) used with the material compared to those used initially.

Additionally, we found that three terms are often used alongside reproducibility, acting as a specification of its own meaning, namely the *replication*, *repeatability* and *reuse* terms. Moreover, these terms seem to be more consistent regarding their definition across the various fields. Therefore, we have used these works and definitions as a basis to clarify what we mean by reproducibility for analysis processes, and eventually by capitalisation of analysis processes. We consider reproducibility as the result of an analysis process that is simultaneously replicable, repeatable and reusable.

Replication means that an analysis process has its operations clearly identified and their order defined. Therefore, replication does not specify any contextual requirement and acts as a representation of an analysis.

Repeatability introduces the possibility to redo an analysis with the same dataset, to verify the results produced. Therefore, contextual dependencies are introduced. Without such repeatability, there is no information about whether an analysis is prone or not to biases and misleadings (scientific or technical). Repeatability is an important part of an open-science dynamic.

Reuse implies that an analysis is sufficiently well designed to be used on other datasets. Thus, only minimum modifications can be made to its implementation, so as not to alter expected results and scientific theory backing it. This reuse requires reduction of context dependencies.

With these definitions, a hierarchical structure within the reproducibility of analysis processes is clearly visible. Figure 1 summarizes the hierarchy of these three properties, in blue. These blue rectangular arrows indicate that the reuse property relies on the property of repeatability, which itself relies on the replication property. These three properties constitute reproducibility.

However, from our perspective, capitalisation encompasses reproducibility and involves three more properties: a property of *comprehension*, a property of *openness* (including the sharing property), and a property of *adaptation*.

A **comprehensible** analysis process means that the different aspects of the analysis can be understood by the analyst. Not only should technical information be described, but also more conceptual information, such as the goal of the analysis, the scientific theories employed or the data used. Comprehension of analyses is required to perform complex tasks, and this lack of comprehension is often the reasons why analyses are not reused [3].

Making an analysis process genuinely **open** can be a complex task, as shown in section 2. Openness requires an analysis process to be accessible, thanks to an open repository for example. However it also needs to maintain its scientific and implementation consistencies while outside of its former analysis tools. Therefore, an open analysis process must use unambiguous definitions for both itself and its inner components.

Finally, **adaptation** of an analysis process indicates that modifications can be performed to address other needs. These needs should have a context similar to the context of the initial analysis. However, modifications performed require respect and matching of the conceptual ground of the analysis itself, as otherwise improper analysis processes could emerge.

The rectangular arrows in Figure 1 show how these three properties coexist with reproducibility to create capitalisation of analysis processes. Again, the succession of rectangular arrows indicates that a property is required by the property following it. Without one of these six properties, we believe that capitalisation of analysis processes cannot be correctly performed.

4 AN ONTOLOGICAL FRAMEWORK

Our goal is to enable capitalisation of analysis processes of learning traces inside the TEL community. We believe that capitalisation can play the role of a vector of improvement for Learning Analytics, in that it can assist analysis stakeholders in elaborating analyses, in interpreting and validating analyses results, and in sharing analyses. Moreover, it will enable TEL stakeholders (e.g. teachers, educational institutions, students, etc.) to be involved, mostly *via* the consultation and annotation of such capitalised analysis processes.

This section presents our ontological framework for capitalising analysis processes of learning traces. Further to an examination of works on the independence of analysis processes [19], we decided to adopt a more conceptual representation of analysis processes. Currently, representation of analysis processes depends on the analysis tool used, and is dependent on computation prerequisites. We propose a narrative representation of analysis processes. Briefly, the narration consists of an emphasis on non technical elements and notions conveyed by and within an analysis, as well as on the relations that they share. The narration structures information to enable several reading levels. It also sets direct computational possibilities aside, to prevent any biases implied by technical constraints.

4.1 Proposition

The actual analysis paradigm cannot efficiently take into account analysis processes, their inner components, their related information, and the relations of their inner elements. Indeed, capitalisation is impaired by computational biases related to analysis tools. Our

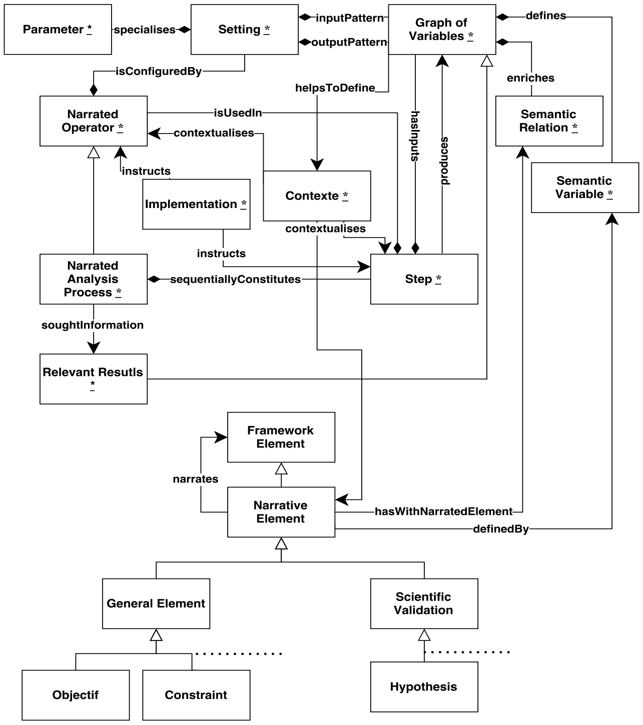

Figure 2: Simplified transposition of our ontology into UML. The * symbol means inherits from Framework Element.

framework formalises these complex analysis elements thanks to semantic elements, structured together. It acts as a projection from these technical elements to higher-level concepts, in order to foster both human and machine comprehensions. This constitutes the narration of analysis processes. Therefore, narrated analysis processes are not designed to be directly computable (*i.e.* do not process data).

Below, we present the ontological framework in detail and explain how the narration is shaped. We present it in an incremental way, to emphasise which ontological elements are the most involved in each layer of capitalisation's properties (see Figure 1). Figure 2 is a simplified transposition of our ontology into UML. The full version of the ontology is accessible online[2]. It reuses some terms from other works, such as xAPI or wf4ever. The goal is to foster interoperability between works and technologies, and ultimately between communities.

An analysis tool T_i has a set of operations OP_{T_i}, a set of operation attributes $B^{T_i}_{OP}$ and a set of specificities W_{T_I}. Let us define it as $T_i = \{OP_{T_i}, B^{T_i}_{OP}, W_{T_i}\}, i \in \mathbb{N}$.

Definition 4.1. There is a narrative function n_i projecting the elements of an analysis tool T_i into our ontological framework F, such as:

$$n_i : T_i \longrightarrow F \tag{1}$$

4.1.1 Replication. According to the given definition of replication in section 3, to replicate an analysis process requires definition of the operators used and their order. With capitalisation as a goal, we believe that it is not possible to use current developed operators or formalisms, as both are dedicated to particular needs and cannot be used for a generic approach. Thus, they are not suitable candidates for replication (see Section 2). However, considering common denominators between similar implemented operators has proved to be a good procedure for producing technically independent operators [19]. Therefore, we define the concept of narrated operator as the semantic conveyed by similar implemented operators, with no computational capability. This means that implemented operators sharing same goals can be grouped under the same narrated operator, independently of their analysis tool.

Definition 4.2. Let OP^k_{Sim} be a set of similar operators OP^k belonging to n different analysis tools T_i, $i, n \in \mathbb{N}$, such as $OP^k_{Sim} = \bigcup_i^n OP^k_{T_I}$. We define the associated narrated operator NOP^k as follows:

$$NOP^k = n_i(OP^k_{Sim,i}) \tag{2}$$

Where n_i is the narrative function associated with T_i (see eq. 1).

As an illustrative example, let us consider a filter operator. Depending on the analysis tool, its implementation, its requirements, its configurations and even its behaviours may be different. A filter used in a workflow environment will be different from that used programatically or from that used on semantic data (e.g. using a SPARQL query). However, the intent is the same: namely to filter something. A narrated operator *filter* will represent these various forms with a semantically unified definition.

The behaviour of a narrated operator is defined by its input pattern E_i, output pattern B_i, and parameter pattern P_i. The top of Figure 2 illustrates these relations. The input pattern of narrated operator defines what elements are expected to be used by it. This means that applying implemented operators of the narrated operator type is expected to be consistent if the elements concerned match the input pattern. The output pattern represents how the data descriptions are supposed to evolve when an operator of this type is used.

Analogously, we define the concept of the narrated analysis process. It represents the semantic conveyed by analyses implemented in different analysis tools, but with the same goals. It can be assimilated to a high-level methodology of an analysis. In our ontology, the narrated analysis process concept is subsumed by the narrated operator concept. Consequently, narrated analysis processes can be used as an operator in the description of other narrated analysis processes, in accordance with the imbrication property [1].

Definition 4.3. Let AP^i be an analysis process implemented in an analysis tool T_i, made up of H ordered operators $\{OP_{T_i}, h\}, h \in H \in \mathbb{N}$ the position. We have the associated narrative analysis process NAP^i as follows:

$$NAP^i = n_i(AP^i) \tag{3}$$

However, a narrative analysis process is not just the combination of the implemented operators which have been described within our framework. It also consists of the other elements ε of the framework F presented in the sections below, where $\varepsilon \subseteq F$.

$$NAP^i = \sum_{h=0}^{H} n_i(OP^h_{T_i}) + \varepsilon \tag{4}$$

[2] https://liris.cnrs.fr/~alebis/research/CAPTEN/ontology.html

4.1.2 Repeatability. Data formats and data granularities need to be unified among analysis processes, in order to guarantee the repeatability property. To overcome these limitations, we reduce granularity of data to the essential by conserving only the data descriptions, *i.e.* the variables. By so doing we preserve the global semantic of traces, while avoiding technical constraints. Additionally, these variables are supposed to be linked together according to the implicit relations they share. By (manually) extracting such implicit relations, the resulting graph of variables conveys information that can be used as resources for repeatability. Indeed, a graph of variables is an overview of variables and their relations. Therefore, it can be associated with any data formats or structures, as long as the information conveyed by the data match such a graph.

A graph of variables is a directed graph defined by $g = (V, A)$ where V is the set of variables, and A the set of the directed edges $a_n = (v_i, v_j)$, where $a_n \in A$, and $v_i, v_j \in V$. These graphs of variables are used to represent data in narrated analysis processes. They are also used to represent the input pattern E_i and the output pattern B_i of a narrated operator i. Figure 2 shows how this concept of graph of variables fits into the ontological framework.

Although narrated operators do not compute data to produce other data, they still use graphs. Matching of a narrated input pattern with a graph of variables must result in a new graph of variables considered as the output. The resulting graph acts as a view of what is happening to the variables, at a specific moment in the narrated analysis process. This is equivalent to a state transition of the variables of the analysis when a narrated operator is applied. Based on Kreuseler et al.'s work [18], we define the following function $\gamma : I \times NOP \times Z \rightarrow O \mid I, O \in G$, as the state transition function between graphs of variables and narrated operators, where I is a set of inputs, NOP is a set of narrated operators, Z is a set of steps (described further in section 4.1.4), O is a set of outputs, and G the set of the graphs of variables.

Figure 3 represents a step in a narrated analysis process (see section 4.1.4). It shows the effect of a narrated operator (here, a correlation) on an input graph of variables via the output graph of variables. The variables used with a narrated operator are associated with its input pattern nodes. Here, the green variables of the input are those linked to the pattern variable *Numerical Entity.* The output pattern of the operator represents the generic variable expected of such a correlation narrated operator: a coefficient of correlation. This generic correlation variable has to be contextualised according to the current state (see section 4.1.6) before being attached to the output graph. In this illustration, a linear coefficient of correlation is represented.

4.1.3 Comprehension. Information described directly in implemented analyses is scarce and unstructured. This is a result of computation and execution necessities of the analysis tools [3]. From our observation, we find that information described in analyses has often three dimensions: technical, methodological, and related to the utilisation of results. While the technical dimension relates to very specific information about analysis tools, the two others provide a more comprehensive understanding of analyses. Methodological information tends to explain the validity of the analysis (such as a scientific validity), while utilisation information tries to prevent exploitation misuses.

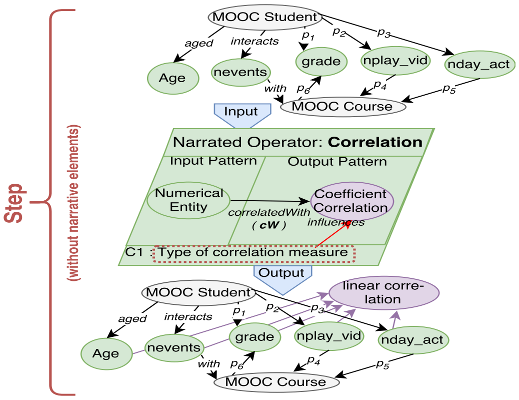

Figure 3: Representation of a step in a narrated analysis process. A step (section 4.1.4) involves a narrated operator (section 4.1.1), the green trapezoid at the centre, applied on input, described by a graph of variables on the top (section 4.1.1). The produced output is described by the bottom graph. Some semantics are applied on vertices and directed edges (section 4.1.5).

Our narrative approach is designed to represent this information. Our framework defines the notion of a narrative element (see Figure 2). The framework structures information related to an analysis with independent subsets of typed information. Thus, a narrative element $m_i \in M$ represents a specific type of information with a predefined meaning (e.g a name, a goal, an hypothesis).

Moreover, a narrative element cannot exist on its own in our framework but it is always related to strictly one framework element. The resulting relation is a directed edge $a_i = (m_i, x)$, where x is an element of our framework F, such as $x \in F$. This implies that each represented element of a narrated analysis process or operator can be justified, explained or described. This relation property also defines complex descriptive structures. A narrative element is defined as a framework element and consequently can also have other dedicated narrative elements.

Definition 4.4. Let τ^{-1} be a transposing function for an implemented operation from a tool to our framework, where $\tau_i^{-1} : T_i \rightarrow NOP \times NAP$. Let ϕ be a function extracting the related information associated with an implemented operation, where $\phi_i : T_i \rightarrow N$, where $N = \{M, A_M\}$ with $A_M = \{v_i, v_j\}$, where v_i is a narrative element, such as $v_i \in M$. We can further define the narrative function (1) as a combination of the transposing function τ_i^{-1} and of the extracting function ϕ_i:

$$n_i = \tau_i^{-1} + \phi_i \tag{5}$$

As an example, let us consider an implemented analysis where a statistical model is built. Then, the null hypothesis for this model is tested. The threshold used here for the p-value is lowered to 0.005 instead of a more usual threshold of 0.05. In our framework, it is possible to describe in a structured way information associated with this threshold lowering. For example, this lowering could be based on a hypothesis derived from the analyst's reflection provided by a studied paper, dedicated to the analysis field. Here, *based*, *derived from*, *provided by* and *dedicated to* are the relations (or directed edge) that link the narrative elements *A hypothesis*, *the analyst's reflection*, *a studied paper* and *the analysis field*.

4.1.4 Reuse. In order to be considered as reusable, an analysis process implementation and its results should be able to withstand slight modifications. Otherwise, implementation issues may arise as well as inconsistent outcome results. These modifications may originate from similar datasets or from a change of analysis tools.

To address this necessity for sustainability to slight changes, we aim to capture the intent of the analyst when an operator is used within an analysis process, in an analysis tool. We define the notion of step. Unlike a narrated operator which only represents an operation to perform, a step conveys information about the use of this operation on data. Therefore, a step z encapsulates a narrated operator, the input graph of variables and the output graph of variables, as shown in Figure 3. A step also encapsulates the intents it conveyed by means of the narrative elements. We have the following tuple $z = \{I_z, NOP_z, O_z, N_z\}$. Therefore, a step is a narrative description of the state evolution of the inputs. It is the building block of the narrative analysis processes (see Figure 2). Moreover, the relation shared by steps can be represented by a directed edge $a_{z_i z_j} = (z_i, z_j)$.

To prevent both misinterpretation and misuse of results, the expected results of an analysis are formalised in our ontology. We define the notion of relevant results K_i that should be produced by a narrated analysis process i. To describe the general meaning of these relevant results, variables are used. The relevant results K_i are also represented as a graph of variables which is a subset of outputs produced by the steps, where $K_i \subseteq O_i, O_i \subseteq O \in G$. Narrative elements can also be used to describe these results. Narrative elements should be used to add specificities related to the results (e.g. to list data obtained during an analysis). Narrative elements can also be used to describe important information, such as the expected scope of the validity of results.

Definition 4.5. A narrated analysis process produces relevant results for a specified need. The inner differentiation we make between a narrated operator and a narrated analysis process is that a narrated operator does not produce relevant results. This differentiation can be expressed as follows:

$$NAP_i = NOP_i \leftrightarrow K_i = \emptyset \tag{6}$$

Narrated operators and narrated analysis processes are represented in a non computational way in our framework. Nevertheless, information about implementation is formalised and embedded into narrated operators, narrated analysis processes and steps. It consists in indicating which analysis tools can perform the specified task, and the associated operations and configurations. Thus, implementation information acts as an injective function $\tau : NOP \times NAP \longrightarrow T$, allowing a narrated operation to be instantiated in an analysis tool.

4.1.5 Openness. Disparate implementation of analysis tools has a direct impact on the openness of analysis processes. This creates semantic divergences when analysis processes are used outside their former context. To address this issue, our framework uses the semantic web and proposes a controlled vocabulary W. It is defined as a set of semantic terms $\{w_1, \ldots, w_n\}$, where $w_i \in W, i \leq n, n = card(W)$. These terms are represented by their Internationalized Resource Identifier (IRI) in the framework. Therefore, works such as xAPI or wf4ever can be used in the controlled vocabulary W to promote interoperability and uniformity.

These semantic terms are used to describe the framework elements. To do so, a semantic term is either a semantic class $w_i^v \in W^v$, $W^v \subseteq V$ or a semantic property $w_i^p \in W^p$, such as $W = W^v \cup W^p$. A directed edge $a_j = \{v_0, v_1\}$ is used with a w_i^p semantic property to create a semantic triple a'_j such as $a'_j = (v_0, w_i^p, v_1)$. The conveyed semantic of the relation between two elements is then expressed in our framework.

It is possible to enrich the controlled vocabulary with new terms and use them afterwards. Thus, for n new semantic terms, we defined the vocabulary enrichment as $W \cup \{w_1, \ldots, w_i, \ldots, w_n\}$, where $0 < i \leq n$. We believe that this property can also lead to a unified and shared vocabulary inside the TEL community.

As a comprehension example, let us consider Figure 3. The narrated operator is named Correlation. This name is a narrative element of the class *name*, semantically identified by the term *<IRI:name>*, which has been instantiated. Here, *<IRI:Correlation>* is its content, also a semantic term. The graphs of variables (inputs, outputs and patterns) are also semantically enriched. Terms are used for the relations and for the variables. Here, use of xAPI vocabulary could become extremely relevant to gain even more interoperability.

4.1.6 Adaptability. Finally, we were interested in making the narrated analysis processes adaptable. We have observed that adaptability and context are closely related. However, this important information is either not representable or sparse and badly structured in the analysis tools. This lack is at the origin of the efforts made in works concerning additional resources attached to workflows [4]. Therefore, we propose a formalisation of the context of an analysis in our framework.

We propose to formalise the context C in three categories; C^1 category, which defines the analysis context itself (i.e. dependencies generated by the elements used in the analysis); C^2 category, which defines the utilisation context of the analysis (i.e. pedagogical situation in which a process is usable); and C^3 category, which defines the viability context of produced knowledge (i.e. the scope in which knowledge is supposed to be relevant), such as $C = \{C^1, C^2, C^3\}$.

The context C is extracted from an implemented analysis by studying its available resources and understanding their relation to the analysis. However, even data itself can have a direct impact on contexts. That is why our framework can represent the effects of data on the context C, mostly through the narrative elements. Figure 2 shows the relations existing between different elements of the framework.

Definition 4.6. Let ψ be a function extracting the context of an implemented analysis from an analysis tool to our framework, such as $\psi_i : T_i \longrightarrow N$. Let κ be a function extracting specificities of an implemented operation from an analysis tool to our framework, such as $\kappa_i : OP_i \times B^i_{OP} \longrightarrow N$. The function extracting the related information of an implemented operation (5) is defined as the following combination of these two functions above:

$$\phi_i = \kappa_i + \psi_i \tag{7}$$

These structured contexts are used in our ontology to indicate critical points in narrated analysis processes and steps. Thus, contexts are semantically related to these elements. Contexts are also used to specialise the generic behaviour of a narrated operator when it is used inside a step. This is illustrated by the function *ParameterInfluence*, *VariableContextualisation* and *VariablesRelation* in Algorithm 1. Algorithm 1 presents, at a step $z \in Z$, the state transition of an input graph of variables I_z processed by a narrated operator NOP_z, and the resulting output graph of variables O_z. The γ state transition function first requires that all variables of the NOP_z input pattern are associated with an input variable of the input graph. The same applies to the parameters. Then, according to the NOP_z behaviour defined by its output pattern, the new graph O_z is produced with the new variables and relations contextualised.

Algorithm 1 γ state transition function

```
Require: $E_z, P_z, B_z \in NOP_z, I_z \in I \in G, z \in Z$
  for all $e \in E_z$ do
    $v_e \leftarrow$ AssociateVariable($e, I_z$)
    $U \leftarrow \{e, v_e\}$
  end for
  for all $p \in P_z$ do
    $v_p \leftarrow$ ParameterInfluence($p, I_z$)
    if $v_p \neq \emptyset$ then
      $U' \leftarrow \{p, v_p\}$
    end if
  end for
  $O_z \leftarrow I_z$
  for all $b \in B_z$ do
    $V \leftarrow$VariableContextualisation($b, z, U, U'$), $V \in O_z$
    if $U \neq \emptyset \vee U' \neq \emptyset$ then
      for all $u \in U$ do
        $A \leftarrow \{u, b,$VariablesRelation$(u, z)\}, A \in O_z$
      end for
      for all $u' \in U'$ do
        $A \leftarrow \{u', b,$ VariablesRelation$(u', z)\}, A \in O_z$
      end for
    end if
  end for
```

As a final example, let us consider again the step presented in Figure 3 which defines a correlation between the age of a student, his/her grades, his/her events and videos played in a MOOC. The threshold used to define whether or not variables are correlated is, in fact, contextualised by a MOOC environment (*i.e.* corresponds to C^1). In a more pervasive learning context for example (also C^1), it could be *hypothesized* that the correlation threshold should be relaxed, due to the diversity of resources available to the student. Therefore, taking context specificities into account leads to adaptation recommendations before reusing this step, thus possibly avoiding misuses and misinterpretations.

5 INSTANTIATION OF OUR FRAMEWORK

We have developed the prototype CAPTEN (Capitalization of Analysis Processes for Technology Enhanced learNing) for the capitalisation of analysis processes of learning traces. The prototype currently implements an important subset of our ontological framework. All the elements presented in Figure 2 have been implemented, at least partially, and can be used to represent analyses. It is a client side application, based on web technologies. It is available online[3] and can be installed locally. This section explains the methodology used to reify several existing analysis processes in our instantiated framework. We then comment the impact of the narration.

5.1 Reifying Analyses in CAPTEN

Before importing existing analysis processes in our prototype, we populated it with common operators and TEL concepts. First, we looked into which TEL terminologies were likely to be used. To do so, we manually studied several datasets and papers in order to extract recurring terminologies, such as *Student* or the *answers* action. We then defined them in the vocabulary of the prototype, with a built-in interface.

Afterwards, we consulted 5 analysis tools: Orange/UnderTracks, SPAD, R, Knime, Weka. We looked for basic operations, such as addition or filter. The goal was to identify which operations are common in the majority of these tools (majority was defined as at least 4 tools out of 5). These common operators were then defined in our prototype as narrated operators. The 3 patterns of a narrated operator were defined by observing the common behaviour of each operator inside its analysis tool.

On completion of population, we selected 9 TEL analyses. These analyses were mainly derived from our current research project. The reason for this choice is related to the convenience of analysis accessibility and the discussion between the initial analyst(s) and ourselves. The description of the 9 analyses used are accessible online[4]. These analyses were also chosen because they were implemented using different analysis tools - sometimes twice or more. Some of them were tabular tools (e.g. Excel), programmatic tools (e.g. R), visual tools (e.g. UnderTracks/Orange) or scientific prototypes (e.g. Usage Tracking Language [9]).

Each analysis was treated independently of the others. For each analysis, we first began by studying the needs it meets, as well as its goals. We then searched for contextual information within associated documentation and data, before carefully studying the initial data used in the analysis process and identifying their associated variables. Then, we extracted the relations existing between these identified variables, mainly thanks to documentation. Following this, we defined the corresponding graph of variables, by using a dedicated graph editing interface from our prototype.

Subsequently, for each analysis, we vetted each operation. This time consuming task had several goals. First, to identify the operation used, as well as its settings, its processed inputs, and the resulting outputs. Second, to correctly understand the intent behind applying such an operation. The final goal was to extract the information conveyed by the operation. Finally, we associated these elements with their related abstract concepts inside the prototype. An encountered operation did not always have a direct match with a narrated operator. The reason for this is either that the narrated operator representing this operation concept had not yet been described in our prototype, or the concerned operation is impaired

[3] https://github.com/alexislebis/CAPTEN

[4] https://hubble.lip6.fr/

by technical specificities. For the former case, we created the dedicated narrated operator, while for the latter, we considered the contiguous operations of the operation concerned, in order to shape a consistent operation concept. We then matched this association with a narrated operator.

We then defined a step with the corresponding narrated operator and the input graph of variables. The output graph was automatically processed by our prototype, under our supervision. We manually specialised some generated variables and relations to match their actual meaning. The information extracted, related to the analysis, was associated with narrative elements placed in relation with this analysis. Finally, we identified which variables had to be considered as the relevant results, according to the initial analysis.

5.2 Discussion

We successfully implemented the 9 existing analyses into our prototype. Several times, inside an analysis, a sub narrated analysis process could be defined, with its own goal. We therefore defined it too. This sub narrated analysis process was then used as a narrated operator for the main narrated analysis process. Also, we encountered some difficulties in managing operations which implicitly act as a *for each* operation or a *group by* operation. We suspect that these operations convey several goals at a time. However, these goals do not seem to be to be consistent depending on analyses. We addressed these issues by breaking them down into sub narrated analysis processes.

Furthermore, the quality of narrated analysis processes is closely related to the efforts deployed to narrate it. The task of reifying an existing analysis in our prototype was a meticulous and time consuming one, requiring huge efforts to understand the analysis, as well as a deconstruction-based approach. This is because technical format specificities have to be understood in order to properly match elements of analysis tools to elements of our framework. However, we expect fewer efforts to be required if the analyst who performed the analysis is the actor of this reification.

6 EXPERIMENTATIONS

6.1 Protocol

To further test our approach and its relevance for the TEL community, we conducted experimentations. Figure 4 shows the capitalisation cycle. The reification phase that we performed in section 5 tests the green part of the capitalisation cycle, namely the capitalisation of the implemented analyses. However, in order to consider that these reified analyses are effectively capitalised, the reuse and adaptation properties of the narrated analysis processes for other needs have to be tested (as defined by Figure 1). Therefore, the goal of these experimentations was to evaluate how end-users of our platform understand and possibly find, reuse and adapt relevant narrated analysis processes (or a sub-part of them) to their needs. This results in evaluating both the blue and red parts of Figure 4.

The experimentations involved 6 persons, one person at a time. Each experimentation lasted three hours per person. These persons are all used to working in the TEL field. They themselves evaluated their expertise level as to the analysis of learning traces, on a scale from 0 to 10 meaning *"no expertise at all"* and *"expert"*, respectively.

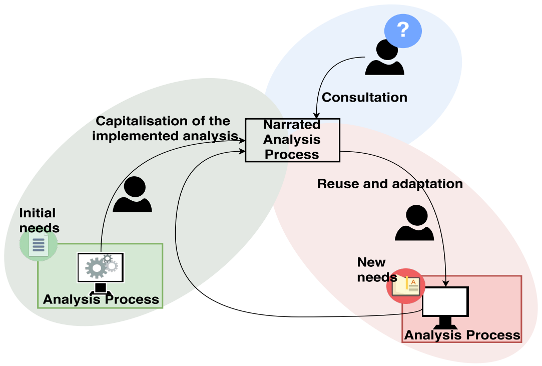

Figure 4: Illustration of the capitalisation cycle and the three major phases related to capitalised analyses.

While results were disparate, on average they were all used to conducting such analyses (mean $\bar{x} = 5.83$, variance $\sigma^2 = 2.47$). Therefore, our approach was tested with different analyst profiles.

The first part of the experimentation was dedicated to presenting the analyst with our narrative approach, *via* the CAPTEN prototype. A predefined narrated analysis process, as well as its related elements, was presented and explained. The theory behind the framework elements was also explained. This part lasted approximately half an hour, depending on the questions asked by the evaluated analyst.

During the second part of the experimentation, the evaluated analyst had to lead an analysis in order to answer one analysis need (or more, if time permitted). He/she had a choice between two predefined needs. The first need was as follows: *"Predict a student certification at the end of a course"*. The second need was as follows: *"Identify student profiles, and if possible, by course"*. Both needs were to be answered using a MOOC dataset [21] that we provided, as well as its official documentation. For an hour and half, the analyst was autonomous. He/she was allowed to access any information medium to help him/her in the analysis task, excepted the CAPTEN prototype. Moreover, the analyst was free to choose the analysis tools he/she wanted. No help was provided concerning the analysis task.

Afterwards, if the analyst completed his/her analysis (2 out of the 6 persons did), or if he/she encountered difficulties preventing him/her from continuing the analysis (3 out of the 6 persons), or if there was no time to continue the analysis (1 out of the 6 persons), we made CAPTEN accessible to the analyst. We loaded into the prototype the narrated analysis processes reified previously (see section 5). The goal of using CAPTEN depended on the state of progress of the analysis. If completed, it was mostly to improve the quality of the analysis implemented by the analyst, and to give him/her new insights into what can be obtained. Otherwise, it mostly consisted of assistance in helping him/her to finish the analysis, or at least to improve the overall quality. To do so, analysts had to search inside CAPTEN for narrated analysis processes, or sub parts of them, that could be adapted and reused. While the analysts were as far as possible autonomous, we sometimes had to intervene for ergonomic reasons, in order to explain the prototype interfaces and to assist in their navigation.

Finally, the last fifteen minutes were dedicated to answering questions on a form that evaluated various aspects of the experimentation. It also evaluated the analyst's opinion of our proposition and how our approach fits into the TEL field. All the experimental materials (including the results of the linear scales used to calculate the means and variances below) are available online[5].

6.2 Results and discussions

As a first observation, the two needs were approximately equally chosen by the analysts (4 persons chose the first one, 2 the second). In any case, all the evaluated analysts have stated they already had an idea how to design the analysis in order to address the need. This seems coherent since we let them choose between the needs. However, one person did not know what was to be sought to answer the need, although he/she had an idea of what analysis to perform. We believe that analysts have some analysis patterns in their mind and that these are triggered by keywords. However, these patterns are then specialised according to the context. This could explain why this person had an idea about how to design the analysis without knowing its answer. The other analysts tended to choose the need they could answer (4 out of the 6 persons, independently of needs).

During the experimentation, the evaluated analysts manipulated a diversity of analysis tools, sometimes two at a time (this was the case for two analyses). Five analysis tools were used: Excel, RapidMiner, R, Coheris Analytics SPAD and SAS Enterprise Miner. The choice criterion was almost always related to the expertise of the analyst with the tool (4 out of 6 persons). However, tool efficiency also became a choice criterion for three persons (arguably related to the allotted time of the experimentation analysis phase).

To the question "*Was CAPTEN helpful to your analysis?*", one person answered "*No, CAPTEN does not help me*". This answer is important and has narrowed down our thought about when capitalisation can be helpful. Indeed, the profile type of this analyst was an expert one. He/she has already worked on very similar case contexts (also using similar datasets) and led several similar analyses related to student profiles in MOOC. Therefore, we suspect that these kinds of expert profiles are not the first to be concerned by the assistance provided by the capitalisation. They will inherently rely on their own significant expertise. However, these kinds of expert profiles are important for acting as a provider of capitalised analyses and to improve them (the green part in Figure 4 and the retroactive arrow in the red part).

To that same question, the other 5 evaluated analysts answered that "*Yes, it helped me to improve my analysis*". Four of them specified that they reused narrated analysis processes, or sometimes sub-parts, inside their own analysis process, in order to improve it. The last person indicated that CAPTEN gave him/her insights into other analysis methods and that it helped him/her to conclude his/her initial analysis. CAPTEN was also used to search for precise information (4 out of the 5 persons). Some information was related to the choice of the appropriate variables for the analysis, while other information concerned consistency of operations and how and where to use these operations in the analysis. These preliminary results support the relevance of our approach regarding capitalisation *via* narration. The last column of in Table 1 is an observation of the helpfulness of framework elements in understanding the narrated analyses (it concerns all 5 analysts).

Framework Elements (# of analysts)	Trace (2)	Context (3)	Analysis (5)
Narrated Analysis Process	1	3	5
Narrated Operator	1	0	3
Graph of Variables	1	1	4
Step	2	3	5
Knowledge	0	0	4
Narrative Element	0	2	5

Table 1: Overview of the effect of the framework elements that helped analysts to understand traces, contexts and narrated analysis processes. The number in each cell indicates how many analysts were concerned by an element.

We then observed these five analysts to find out how they reused and adapted the chosen narrated analysis processes. We used a scaled notation from 0 to 5 meaning *"absolutely not"* and *"totally"*, respectively. To the question asking if they managed to adapt the narrated analysis processes that they chose, we have a mean of $\bar{x} = 3$, and a variance of $\sigma^2 = 2$. Since adaptation of an analysis is a complex task, these preliminary results are very motivating. It shows that the subset of our ontology implemented in our prototype already has a strong impact on adaptability. To the question related to the reuse of the chosen narrated analysis processes, we have a mean of $\bar{x} = 1.6$ and a variance of $\sigma^2 = 1.36$. Analysts feedback shows that these results are mostly related to the lack of dedicated implementation instructions. Indeed, the prototype, which is still in its early development phase, currently encompasses implementation information in narrative elements and not in a dedicated structure.

We further investigated reuse and adaptability entanglement with data, contexts and goals. Our goal in this case was to outline the similarity level required between information available during the analysis and information encompassed by a narrated analysis, in order to perform reuse and adaptation tasks. We used a scaled notation from 0 to 5 meaning *"independent"* and *"identical"*, respectively, to collect feedback. Concerning data, we obtain an estimated level of required similarity of $\bar{x} = 2.6$, with a variance $\sigma^2 = 1.04$. Concerning context, we obtain a higher level of similarity required, with a mean of $\bar{x} = 3.4$ and a variance of $\sigma^2 = 0.64$. Finally, with respect to similarity of goals, we have $\bar{x} = 2.4$, with a variance of $\sigma^2 = 0.24$. These preliminary results reinforce our intuition about the dependencies of analyses on contexts.

Our prototype has been well endorsed as an assistance to reuse and adaptation of analysis processes ($\bar{x} = 6, \sigma^2 = 2$, on a scale from 0 to 10 meaning *"useless"* and *"indispensable"*, respectively). Our approach was greatly preferred to textual approaches and also preferred to workflow approaches. Moreover, we track the side effect that our prototype had on the comprehension of the traces and the context of the analysis. Two analysts indicate that they were able to improve their initial comprehension of traces by consulting several narrated analysis processes in CAPTEN. The second column of Table 1 shows, for these two analysts, which elements were

[5] https://liris.cnrs.fr/~alebis/research/CAPTEN/capten_xp.html

involved in this improvement. Three analysts also indicate that their context comprehension of the analysis had been improved thanks to CAPTEN. The third column of Table 1 shows, for these three analysts, which elements were involved in this improvement.

We also collected feedback from analysts as to what CAPTEN had provided them with. Besides the fact that CAPTEN was a support to analysis elaboration, we can extract three major assistance areas. The first concerns assistance in analysis setting and design (e.g. "*review some operator settings*"). The second concerns comprehension of the analysis and the needs (e.g. "*to search for new types of results [for the analysis]*"). Finally, the third area concerns assistance regarding analysis quality (e.g. "*to have another method to compare myself*").

As a global observation, our narrative approach for capitalisation of analysis processes of learning traces assisted the majority of the analysts evaluated in their analysis. This narrative approach provided them with the possibilities to reuse, with proper contextualised adaptations, existing analysis processes to their needs.

To conclude, the theory behind CAPTEN was understood by those who were evaluated. These experimentations yield strong preliminary results about the reuse and adaptability of already capitalised analysis processes. Finally, our approach was rated as a potential candidate for the capitalisation of analysis processes inside the TEL community ($\bar{x} = 8, 17$, $\sigma^2 = 0, 47$, based on a 0 to 10 notation meaning *"useless"* and *"fulfil the goal"*, respectively).

7 CONCLUSION

This paper presents an approach for capitalising analysis processes of learning traces inside the TEL community. It uses an ontological framework dedicated to their narration. We also propose a formalisation of our ontological framework. The experimental results seem to confirm that it is possible to shift the actual paradigm of analysis processes to one based on a narrative approach. This will lead to inherently comprehensible and open analysis processes. Moreover, this will enable analysis processes to be designed with real reuse and adaptation properties.

The major challenge behind this approach is to group TEL efforts and to provide a new way of designing analyses inside the community. By reusing and adapting what already exists in our community, we believe that co-constructed emergent behaviours will emerge. The results will be a generalised involvement, reinforcing the dynamics inside the TEL community, and will also contribute to an improvement of the overall scientific and pedagogical quality. Moreover, it will foster the emergence of new needs, techniques and specificities related to the TEL field (such as the common TEL vocabulary we expect to see emerge, as explained in section 4.1.5).

We plan to further evaluate our framework by reifying other analyses existing in the learning analytics literature. Furthermore, our principal focus will concern assistance regarding capitalisation. We have made our framework inference-ready. We seek to create inference rules and to automate them. The goal is to assist people involved in analyses, from consultation to adaptation of analyses to other contexts. This introduces exciting analysis co-construction prospects between the TEL community and knowledge-based systems.

ACKNOWLEDGMENTS

This work has been supported by the HUBBLE project (ANR-14-CE24-0015).

REFERENCES

[1] Ryan SJD Baker and Kalina Yacef. 2009. The state of educational data mining in 2009: A review and future visions. *JEDM* 1, 1 (2009), 3–17.

[2] Glenn Begley and John Ioannidis. 2015. Reproducibility in Science. *Circulation Research* 116, 1 (2015), 116–126. https://doi.org/10.1161/CIRCRESAHA.114.303819

[3] Khalid Belhajjame et al. 2012. Why Workflows Break — Understanding and Combating Decay in Taverna Workflows. In *Proceedings of the 8th International Conference on E-Science*. IEEE Computer Society, Washington, DC, USA, 1–9.

[4] Khalid Belhajjame et al. 2015. Using a suite of ontologies for preserving workflow-centric research objects. *Web Semantics: Science, Services and Agents on the World Wide Web* 32 (2015), 16–42.

[5] Daniel J. Benjamin et al. 2017. Redefine statistical significance. *Nature Human Behaviour* (2017). https://doi.org/10.1038/s41562-017-0189-z

[6] Shawn Bowers and Bertram Ludäscher. 2004. An Ontology-Driven Framework for Data Transformation in Scientific Workflows. In *Data Integration in the Life Sciences: First International Workshop, DILS 2004, Leipzig, Germany, March 25-26, 2004. Proceedings*, Erhard Rahm (Ed.). Springer Berlin Heidelberg, 1–16.

[7] Christopher A Brooks, Craig Thompson, and Stephanie D Teasley. 2014. Towards A General Method for Building Predictive Models of Learner Success using Educational Time Series Data.. In *In workshop on LA and ML of LAK 2014*.

[8] Mohamed Amine Chatti, Anna Lea Dyckhoff, Ulrik Schroeder, and Hendrik Thüs. 2012. A reference model for learning analytics. *International Journal of Technology Enhanced Learning* 4, 5-6 (2012), 318–331.

[9] Christophe Choquet and Sébastien Iksal. 2006. Usage tracking language: a meta language for modelling tracks in tel systems.. In *Proceedings of ICSOFT'06*. INSTICC, 133–138.

[10] Doug Clow. 2013. An overview of learning analytics. *Teaching in Higher Education* 18, 6 (2013), 683–695.

[11] Adam Cooper. 2013. Learning Analytics Interoperability-a survey of current literature and candidate standards. (2013). Retrieved Nov. 22, 2017 from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.650.3428&rep=rep1&type=pdf

[12] David De Roure et al. 2010. Towards open science: the myExperiment approach. *Concurrency and Computation: Practice and Experience* 22, 17 (2010), 2335–2353.

[13] Tanya Elias. 2011. *Learning Analytics : Definitions , Processes and Potential.* Technical Report.

[14] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. 1996. From data mining to knowledge discovery in databases. *AI magazine* 17, 3 (1996), 37.

[15] Joint Committee for Guides in Metrology. 2008. *International Vocabulary of Metrology-Basic and General Concepts and Associated Terms.* Technical Report.

[16] ACM Inc. 2016. Artifact Review and Badging. (2016). Retrieved Nov. 22, 2017 from http://www.acm.org/publications/policies/artifact-review-badging

[17] Kenneth Koedinger et al. 2010. A data repository for the EDM community: The PSLC DataShop. *Handbook of educational data mining* 43, Article 4 (2010).

[18] Matthias Kreuseler, Thomas Nocke, and Heidrun Schumann. 2004. A history mechanism for visual data mining. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on.* IEEE, 49–56.

[19] Alexis Lebis, Marie Lefevre, Vanda Luengo, and Nathalie Guin. 2016. Towards a Capitalization of Processes Analyzing Learning Interaction Traces. In *Proceedings of the EC-TEL'16.* Springer, 397–403.

[20] Nadine Mandran, Michael Ortega, Vanda Luengo, and Denis Bouhineau. 2015. DOP8: merging both data and analysis operators life cycles for technology enhanced learning. In *Proceedings of LAK'15.* ACM, 213–217.

[21] MITx and HarvardX. 2014. HarvardX-MITx Person-Course Academic Year 2013 De-Identified dataset, version 2.0. (2014). https://doi.org/10.7910/DVN/26147

[22] Nature Publishing Group. 2016. *Reality check on reproducibility.* Vol. 533. 437. https://doi.org/10.1038/533437a

[23] Kevin Page et al. 2012. From workflows to Research Objects: an architecture for preserving the semantics of science. *Proceedings of the 2nd International Workshop on Linked Science* (10 2012).

[24] Ricardo Queirós and José Paulo Leal. 2011. A survey on eLearning content standardization. In *World Summit on Knowledge Society.* Springer, 433–438.

[25] Cristobal Romero and Sebastian Ventura. 2007. Educational data mining: A survey from 1995 to 2005. *Expert systems with applications* 33, 1 (2007), 135–146.

[26] George Siemens et al. 2011. *Open Learning Analytics: an integrated & modularized platform.* Technical Report. Society for Learning Analytics Research.

[27] George Siemens and Phil Long. 2011. Penetrating the fog: Analytics in learning and education. *EDUCAUSE review* 46, 5 (2011), 30.