Change Detection in Dynamic Event Attributes

Jonas Cremerius(✉) and Mathias Weske

Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
{jonas.cremerius,mathias.weske}@hpi.de

Abstract. Discovering and analysing business processes are important tasks for organizations. Process mining bridges the gap between process management and data science by discovering process models using event logs derived from real-world data. Besides mandatory event attributes like case identifier, activity, and timestamp, additional event attributes can be present, such as human resources, costs, and laboratory values. These event attributes can be modified by multiple events in a trace, which can be classified as so-called *dynamic* event attributes. So far, the process behaviour of event attributes is described in the form of read/write operations or object-lifecycle states. However, the actual value behaviour has not been considered yet. This paper introduces an approach that allows to automatically detect changes in the actual values of *dynamic* event attributes, enabling to identify changes between process activities representing events with the same activity name. This can help to confirm expected behaviour of *dynamic* event attributes, but also allows deriving novel insights by identifying unexpected changes. We applied the proposed technique on the MIMIC-IV real-world data set on hospitalizations in the US and evaluated the results together with a medical expert. The approach is implemented in Python with the help of the PM4Py framework.

Keywords: Process mining · Change detection · Process enhancement

1 Introduction

Businesses organizations seek to find valuable insights out of data stored in information systems with the aim to improve their business processes. Today, such information systems can include data about end-to-end processes or even beyond that. Due to that fact, process mining was developed to understand the actual execution of business processes, providing techniques for process discovery, conformance checking, and enhancement [1].

In process discovery, the discovered process model can be analysed based on the occurred events, their order, and frequency. Event logs might contain additional data, so-called event attributes, providing further information about an event, which can be used to enhance process models [10].

C. Di Ciccio et al. (Eds.): BPM 2022, LNBIP 458, pp. 157–172, 2022.
https://doi.org/10.1007/978-3-031-16171-1_10

Event attributes can be *dynamic* in the sense that they are stored in multiple events, such as an order status or laboratory values, which evolve through the process. As *dynamic* event attributes occur multiple times during the process, understanding their development can be of interest [11]. So far, the process behaviour of event attributes is described in the form of read/write operations or object-lifecycle states [6,16].

However, there is still a lack of describing the actual value behaviour of *dynamic* event attributes. For example, it might be of interest to see if a stay in an intensive care unit (ICU) results in improved laboratory values of a patient in the recovery ward. Thus, we can compare the laboratory values conducted in the ICU to the ones in the recovery ward.

Therefore, this paper provides an approach to automatically detect changes in *dynamic* event attributes, so that it is not only known if the values change throughout the process, but also at which activity representing all events with the same activity name and in which direction (increasing, decreasing).

The remainder of this paper is organized as follows. Section 2 provides related work, and Sect. 3 introduces preliminaries. Section 4 presents the approach for change detection in *dynamic* event attributes, and Sect. 5 applies the approach to the MIMIC-IV real-world data set on hospitalizations. We discuss the approach and its limitations in Sect. 6 before the paper is concluded in Sect. 7.

2 Related Work

The analysis of event attributes has been approached from different perspectives in the literature.

A prominent application is the identification of decision rules, such as in data-aware heuristic mining [16]. Regarding the exploration of event attributes, the multi-perspective process explorer allows investigating the distribution of each event attribute at each activity [18]. Data-enhanced process models add aggregated information about event attributes, such as the mean value, to the process model activities representing the events. In data-enhanced process models, the selection of event attributes for detailed analysis is supported by allowing filtering according to their process behaviour and the degree of variability through the process [11]. In [6], the access to event attributes is described and annotated to the process model, describing the data object lifecycle of each event attribute.

While there exist approaches trying to better explore and understand the actual values of event attributes, there remains, to our knowledge, a lack of understanding the changing behaviour of these values. The work describing the data object lifecycle is already a step in this direction, but lacks support for understanding the change of the actual values behind the event attributes.

Change detection is highly present in time series data, which refers to the problem of finding abrupt changes in data when a property of the time series changes [4]. In terms of process analysis, change detection has been applied to detect and explain concept drifts. In [2], event attributes are used to explain concept drifts, such as that a decrease in the age of customers led to an increase in the prevalence of the email notification activity.

However, time series change detection accepts only one value per time point, which requires methods of aggregations when analysing groups, which is the typical use case in process mining. This leads to information loss and lacks a detailed representation of the analysed group [4].

To overcome this limitation, statistical tests allow comparing two timestamps in more detail. For example, the Wilcoxon Signed-rank Test considers all values of the analysed group and ranks the differences between two timestamps to answer the question, if there is a statistically significant change [15]. This form of change detection is popular in the medical domain, where before-after comparisons are conducted. For example, [9] compares a laboratory value measured at inpatient admission and 72 h after that.

In process mining, statistical tests are used to retrieve a variety of insights. For example, the difference of event durations is assessed between two groups in an emergency process [12], which is not a before-after comparison, but still compares the difference of values in two groups. The same holds for process variant comparison, where the event transition frequency is compared between two process variants [19].

In this contribution, we propose to use statistical tests to detect changes of event attributes in the process. In particular, we make use of the before-after comparison of statistical tests to detect changes of dynamic event attributes between process activities, which has not been conducted in process mining so far to our knowledge.

3 Preliminaries

This paper builds on the contribution of Supporting Domain Data Selection in Data-Enhanced Process Models [11], which starts with an event log. An event log consists of sequences of events, which are grouped into traces. An event can have an arbitrary number of additional event attributes. The following definition is based on [17].

Definition 1 (Event log, Trace, and Event). Let V be the universe of all possible values and E_A be the universe of event attributes. An event e is a mapping of event attributes to values, such as $e \in E_A \rightarrow V$. The universe of events is defined as $E_U = E_A \rightarrow V$. If an event $e \in E_U$ has no value assigned to an event attribute $e_{At} \in E_A$, it is denoted as $e(e_{At}) = \perp$. A trace $t \in {E_U}^\star$ is a sequence of events, and $T \subseteq {E_U}^\star$ represents the respective universe of traces, in which all events are unique. An event log L is a set of traces, so $L \subseteq T$, where each trace is unique as well. As events and traces are unique, we say, that two traces $t_1, t_2 \in L$ belong to the same trace variant $t_{Var} \subseteq L$, if the events in the traces have the same activity ordering and number of events. We refer to T_{Var} as the universe of trace variants.

Normally, an event represents an activity which is conducted within a certain case at a given time, represented by a timestamp. These are treated as regular event attributes in this contribution, so we assume activity, case, and timestamp.

The event instances of a given trace are ordered by their timestamp and have the same case. For simplicity, we assume that the timestamps of events in a trace are never equal. We further assume, that the data type of one event attribute is always the same for all events.

Given events $e_i \neq e_j$ in a given trace t_i, let $e_i > e_j$ represent a directly follows relationship, if e_j appears after e_i and there does not exist an event e_k in t_i which appears between e_i and e_j. Let $e_i \twoheadrightarrow e_j$ represent an eventually follows relationship, if e_j appears at any position after e_i in t_i.

Event attributes can be classified according to their process behaviour, which is based on [11].

Definition 2 (Event Attribute Classification). Before an event attribute $e_{At} \in E_A$ can be classified, the activities using the event attribute and the average number of events using it per trace need to be identified.

Given an event attribute $e_{At} \in E_A$ in an event log L, the set $e_{At_{Act}}$ represents all activities in which the event attribute is used.

$$e_{At_{Act}} := \{e(activity) \in V \mid e(e_{At}) \neq \perp, e \in t, t \in L\} \tag{1}$$

With that, it is known which activities have an event attribute, but it remains unclear whether an event attribute is changing during the process. Therefore, $e_{At_{AvgTrace}}$ describes the average number of events having the event attribute per trace. First, the event log is filtered, so that only traces are included which use the event attribute at least once:

$$L_{e_{At}} = \{t \in L \mid (\exists e \in t)[e(e_{At}) \neq \perp]\} \tag{2}$$

Then, the average number of occurrences of the event attribute per trace can be calculated:

$$e_{At_{AvgTrace}} = \frac{\sum_{t \in L_{e_{At}}} \sum_{e \in t} [e(e_{At}) \neq \perp]}{|L_{e_{At}}|} \tag{3}$$

Three different process characteristics (pc) are defined based on the previously defined features $|e_{At_{Act}}|$ and $e_{At_{AvgTrace}}$.

$$pc(e_{At}) = \begin{cases} static, & |e_{At_{Act}}| = 1, e_{At_{AvgTrace}} = 1 \\ semi-dynamic, & |e_{At_{Act}}| > 1, e_{At_{AvgTrace}} = 1 \\ dynamic, & |e_{At_{Act}}| \geq 1, e_{At_{AvgTrace}} > 1 \end{cases}$$

4 Approach

In this contribution, the goal is to describe the changing behaviour of *dynamic* event attributes through a process. To clarify that, Table 1 illustrates an example event log mimicking a hospital process. Besides the mandatory entries, it contains laboratory values in the form of event attributes. As these are associated to

multiple activities and occur multiple times per trace, these are classified as *dynamic* event attributes. Thus, these are suited for the analysis steps proposed in this paper.

Before the approach is presented, we clarify what kind of change we intend to detect. Our idea is to bring meaning behind the timestamps in the form of activity names and allow identifying, how activities potentially influence the values of event attributes. Therefore, we say that an event attribute changes not if it changes at an arbitrary point of time, but when there is a change in the values between activities. On top of this, we want to achieve this by considering all values of the respective activities.

Table 1. Example event log describing a high level hospital process having laboratory values as event attributes

Case ID	Activity	Timestamp	Bicarbonate value	Creatinine value
1	Admit to hospital	1	140	0.7
1	Treat in medical ward	2	200	0.7
1	Discharge patient	3	120	0.8
2	Admit to hospital	1	135	0.6
2	Treat in ICU	2	100	0.6
2	Discharge patient	3	150	0.7

Looking at the example event log in Table 1, there is a difference in the development of the Bicarbonate laboratory value, dependent on which ward is visited during the hospital process. While it increases in the "Treat in Medical Ward" activity, it decreases in the "Treat in ICU" ward. In the following approach, we identify these changes not in single traces, but make statements for all traces in the event log, deriving a common behaviour of *dynamic* event attributes in the process.

4.1 The Three Dimensions of Change

In this contribution, a three-dimensional perspective is suggested to identify changes in *dynamic* event attributes, which is illustrated in Fig. 1.

The first dimension on the x-axis is the event attribute, because it is the goal of this paper to understand the behaviour of event attributes. The second dimension on the y-axis shows all directly follows and eventually follows relations in the event log, which represent the points of change in the process. Lastly, the z-axis adds information about changes in trace variants, which provides additional context to the relation information on the y-axis. This information is important to preserve the process context, as it might be the case that the process before and after any relation might have an influence on the behaviour of an event attribute.

We start formalizing this construct by defining a change detection cube:

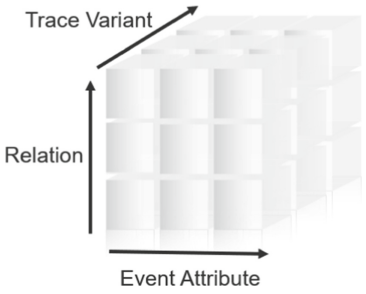

Fig. 1. The three dimensions of change

Definition 3 (Change Detection Cube). We define a change detection cube (CDC_L) for a given event log $L \subseteq T$ as a set of change analysis cells (cac), such that $CDC_L := E_{A_L} \times DFR_L^+ \times T_{Var_L}$, where $E_{A_L} \subseteq E_A$ is the set of event attributes being assigned a value $\neq \perp$ at any event $e \in t, t \in L$ and $DFR_L^+ \subseteq V \times V$ is the transitive closure of directly follows relationships, such that it contains the eventually follows relationships as well. The elements of DFR_L^+ consist of the respective activity names, so if $e_i > e_j$, $(e_i(activity), e_j(activity)) \in DFR_L^+$ and if $e_i \nrightarrow e_j$, $(e_i(activity), e_j(activity)) \in DFR_L^+$. $T_{Var_L} \subseteq T_{Var}$ refers to the set of all trace variants in L.

A change analysis cell $cac \in CDC_L$ represents one cell in the cube, such that $cac = (e_{At} \in E_{A_L}, rel \in DFR_L^+, t_{Var} \in T_{Var_L})$. One cell in the cube refers to a single change detection, for example, looking at Table 1, the Bicarbonate value between the activities "Treat in Medical Ward" and "Discharge Patient" decreases in a trace variant in which the activity "Admit to hospital" is included.

The idea of analysing three-dimensional data in a cube perspective goes back to on-line analytical processing (OLAP), where so-called OLAP cubes were introduced, which can be of higher dimensions as well [8]. These allow operations, which can be applied on the change detection cube:

- Slice: Reduces the cube to a two-dimensional view by selecting a specific value for one dimension, such as the analysis of all changes for one event attribute
- Dice: Creates a sub-cube where specific values for all dimensions can be specified, e.g., analyse all changes for a subset of event attributes
- Pivot: Rearranges the dimensions, such that event attributes and relations swap their axis
- Drill up/down: Changes the level of aggregation in the dimensions, e.g., trace variants could be merged together

With CDC_L defined, each element $cac \in CDC_L$ refers to a change detection analysis, which is defined next.

Definition 4 (Change Detection Analysis). Given an event log $L \subseteq T$ with its respective change detection cube CDC_L, we define a change detection analysis

(CDA) as a function mapping each $cac \in CDC_L$ to a pair of two values representing the result of the change analysis, such that $CDA_L = CDC_L \rightarrow V \times V$.

The result of the change analysis consisting of a two-value pair is generated by statistical tests, which are described next.

4.2 Change Detection as a Before-After Comparison

Given a change detection cube CDC_L for a given event log $L \subseteq T$, we propose to detect changes for each change analysis cell $cac \in CDC_L$ with its elements $e_{At} \in E_{A_L}$, $rel \in DFR_L^+$, and $t_{Var} \in T_{Var_L}$. The relation rel in a cell consists of two activity names (a_1, a_2) of which the events are in a directly follows or eventually follows relationship.

To detect changes, we need to derive the respective event attribute values of e_{At} for both activities (a_1, a_2) from the trace variant $t_{Var} \in T_{Var_L}$. For that, we define a multiset EAV_{cac} for each change analysis cell cac, in which the elements consist of event attribute value pairs $(e_i(e_{At}), e_j(e_{At}))$ with $e_i(activity) = a_1$, $e_j(activity) = a_2$, and $e_i(e_{At}), e_j(e_{At}) \neq \perp$, where the traces including the events e_i, e_j are in the respective trace variant $t_{Var} \subseteq L$. If the respective events are directly following, we only consider directly follows relations in the traces, as it could be the case that a trace includes the directly follows relationship and at some point an eventually follows relationship of both events. Thus, there can be cases where a separate analysis of directly and eventually follows relations makes sense, which could be solved by treating these as separate relations in DFR_L^+, where one is the directly follows and the other the eventually follows relation.

Further, this approach might lead to multiple entries for one case, if the trace includes loops containing the same directly follows or eventually follows relationship. This is intended, as we are interested in the changing behaviour between both activities. However, it could be interesting to investigate the looping behaviour in more detail, such that a value tends to change in the first occurrence of the relation, but remains constant after that. This could be implemented by adding a loop index to each change analysis cell, resulting in separate change analysis cells for each loop iteration. For example, if the relation (a, b) occurs twice in a trace, one could analyse the changing behaviour for the first and second occurrence of (a, b) separately.

With EAV_{cac} representing event attribute value changes for a change analysis cell $cac \in CDC_L$, there exist multiple event attribute values for both activities, given that there are multiple traces related to the change analysis cell. Understanding the changing behaviour between two sets of values is a typical use case in the field of statistical analysis, especially before-after comparisons, e.g., the comparison of laboratory values between two timestamps [9]. As this approach investigates the behaviour of directly follows and eventually follows relationships, we can perform such a before-after comparison for each change analysis cell $cac \in CDC_L$.

We will now introduce statistical tests used for comparing event attribute values in $cac \in CDC_L$.

4.3 Statistical Tests

To conduct statistical tests, two hypotheses need to be provided. First, the null hypothesis states that there is no difference between two samples. These two samples are the event attribute values of two activities represented by EAV_{cac}. Thus, the null hypothesis says, that there is no change in the event attribute values. The task of the statistical test is to either reject or confirm the null hypothesis. By rejecting the null hypothesis, the alternative hypothesis, saying that there is a change in the event attribute values, can be confirmed. We can never say that there is a change for each sample taken, but provide a probability that a given result would occur under the null hypothesis [21]. This probability is the p-value. Thus, the lower the p-value, the lower the chance, that a given sample is not changing. That is the reason why a significance threshold α is used to reject the null hypothesis, which is typically 0.05.

If multiple tests are conducted on the same samples, which is the case when multiple event attributes are analysed for the same relation and trace variant, α can be adjusted by performing a Bonferroni correction [5]. For example, if 10 event attributes are under analysis, one would divide α by 10, resulting in $\alpha = 0.005$. We will not determine a concrete α, but suggest using 0.05 with the option to apply Bonferroni correction, as the application of the correction method depends on the analysis goal. For example, if one wants to determine, if there is no change in any event attribute (universal null hypothesis), the correction should be applied [5].

Choosing the appropriate statistical test is based on three factors. The first factor is the event attribute type, which is either continuous or categorical. We will use the method proposed in [11] to identify the variable type of event attributes in event logs by comparing the total number of values vs. the amount of unique values of a variable. Second, the distribution of data is important. As we cannot make any assumptions about the distribution of each event attribute, we make use of so-called non-parametric tests. Lastly, the relation between the samples under comparison needs to be considered, which is either paired or unpaired. In our case, we have paired samples, because the event attribute values from both activities come from the same case and are not independent. Considering these factors, we end up with the Wilcoxon Signed-rank Test for continuous event attributes and the Stuart-Maxwell Test for categorical event attributes [20].

Wilcoxon Signed-Rank Test. Given a change analysis cell $cac \in CDC_L$ with its event attribute values EAV_{cac}, the Wilcoxon Signed-rank test performs pairwise comparison of each element $(e_i(e_{At}), e_j(e_{At})) \in EAV_{cac}$, given that e_{At} is continuous. The test makes use of the *Simple Difference Formula*, which results in the difference between the proportion of favourable and unfavourable pairs $RBC = f - u$, the so-called matched-pairs rank-biserial correlation, whereas favourable/unfavourable represent the pairs where the differences have the same sign (increasing or decreasing) [15]. As we do not test for a specific direction, we will speak of increasing/decreasing instead of favourable/unfavourable. Table 2

demonstrates an example, where all pairs in EAV_{cac} are compared according to their difference in the activities specified in *cac*. The test calculates each difference, which is shown in the "Change" column. Dependent on the degree of change, ranks are assigned, where increasing/decreasing changes are differentiated in the respective column.

Table 2. Wilcoxon Signed-Rank Test example

Case ID	Treat in ICU	Discharge patient	Change	Increasing	Decreasing
1	150	200	50	5	–
2	140	160	20	3	–
3	100	110	10	1	–
4	150	135	−15	–	2
5	150	180	30	4	–
6	200	185	−15	–	2

As mentioned before, the test considers the rank sums, which are 13 for the increasing and 4 for the decreasing pairs. *RBC* is then the relative difference of both, which is 13/17–4/17 = 0.523. It can take values between −1 and 1, dependent on whether the majority of changes are increasing or decreasing. Thus, it does not only consider if there is a difference in one direction, but also provides information about how many of the major changes go into the respective direction. In combination with a p-value, we can say, that the difference is statistically significant as well.

The major advantage of this test is its simplicity, with its comprehensible calculation of the difference between two groups. Additionally, its result is directional, which automatically identifies an increasing or decreasing behaviour [15].

Stuart-Maxwell Test. If the event attribute e_{At} is categorical, the Stuart-Maxwell test, which is also called the Generalized McNemar test, can be used to identify changing behaviour. In comparison to McNemar, this test can deal with an arbitrary amount of categories [23]. Tests for categorical variables use so-called contingency tables, which represent the transition frequency from one category to the others for before-after comparison. Table 3 illustrates an example of a contingency table of a variable with three categories. It can be seen, for example, that there are 100 cases, where the event attribute remains high and that the event attribute changes from high to normal in 50 cases.

The test checks for so-called *marginal homogeneity*. *Marginal homogeneity* refers to equality between one or more of the row marginal proportions and the corresponding column proportions [23]. For example, the category high in Table 3 has no marginal homogeneity, because the proportion of the row is different to the proportion of the column including the respective category (first row(50) vs. first column(0) without high/high). The test checks this for all categories

Table 3. Contingency table example

–	High	Normal	Low
High	100	50	0
Normal	0	50	25
Low	0	0	75

and results in a p-value p and a chi-squared value χ^2, indicating a change in the respective variable or not, whereas p provides information about statistical significance and χ^2 gives information about how marginal proportions are not homogeneous. Thus, the higher the proportion are not homogeneous, the higher the change in the categories. The exact calculation will not be covered in this paper, but is conducted as described in [23].

The results of the statistical tests of each change analysis cell $cac \in CDC_L$ will be represented as a change detection analyses, such that $CDA_L(cac) = (p, t)$, where the test-statistic t is RBC for continuous event attributes and χ^2 for categorical event attributes.

4.4 Connecting Continuous and Categorical Event Attributes

The differentiation between continuous and categorical event attributes enforces a separate analysis of both. Nevertheless, some event attributes might be connected to each other. A categorical event attribute could describe different states for a continuous event attribute, such as being high, normal, or low. A prominent example are laboratory values, which have these states in addition to their plain value. Thus, there is one attribute for the continuous laboratory value and another one for the categorical laboratory value in the event log. Another example are sensor data, such as temperature measurements etc. Thus, we propose to connect continuous and categorical event attributes by creating a link between change analysis cells $cac \in CDC_L$. This allows to identify, whether a changing behaviour in a continuous event attribute is also represented in the respective categorical event attribute and the other way around. Thus, we define $EAC_L = CDC_L \rightarrow CDC_L$ as an event attribute connection, linking the respective change analysis cells. If there exists no connection, we denote that as $EAC_L(cac) = \perp$.

The linking has to be performed manually, as we do not know of any standardized naming of event attributes in event logs. For example, one could name them equally and assign a variable type to them, which would make the connection trivial.

Next, the proposed approach is evaluated on a real-world healthcare data set, derived from the MIMIC-IV database.

5 Evaluation

The proposed approach was implemented in Python with the help of the PM4Py framework[1] [7]. The relevance of this approach is illustrated in a medical environment, where we generated an event log from the Medical Information Mart for Intensive Care IV (MIMIC-IV) database. The reason for choosing this database is its richness of data, allowing to generate event logs with multiple *dynamic* event attributes.

5.1 Dataset

MIMIC-IV is a relational database including hospital processes of different patients, with procedures performed, medications given, laboratory values taken, image analysis conducted, and more. Its purpose is to support research in healthcare and is therefore publicly available [14].

The event log extracted from MIMIC-IV incorporates a high-level process, describing department visits of patients during their hospital stay, such as emergency department or intensive care unit (ICU). The event log contains 3447 hospital process instances with 13795 events of acute kidney failure (AKF) patients. AKF was chosen together with a medical expert, because of its high prevalence and its measurable disease progression by kidney specific laboratory values.

For each department visit, the event log provides up to 62 event attributes, including laboratory values and demographic information. 56 event attributes represent laboratory values, which are classified as *dynamic*. 28 *dynamic* event attributes are continuous and 28 are categorical. The categorical laboratory values store information about abnormality of the respective continuous value. Thus, we present an event log with multiple *dynamic* event attributes being on different scales with a balance between categorical and continuous event attributes.

5.2 Results

We applied the proposed approach on the event log introduced above. The resulting change detection cube CDC_L can be explored with our artefact. The artefact supports the proposed OLAP operations (Slice, Dice, Pivot, and Drill up/down), where we decided to always slice the cube to enable the exploration of the change analysis results. Therefore, we end up with a two-dimensional event attribute change matrix. Figure 2 illustrates an arbitrary view of CDC_L, showing a subcube with continuous event attributes chosen together with the medical expert and relations having the most changes. The cube was sliced and drilled down to represent all trace variants. Further views, which also consider trace variants, are provided in the already mentioned GitHub repository.

Each cell in the matrix represents one change analysis cell $cac \in CDC_L$ and the number inside displays the test-statistic of the change detection analysis

[1] https://github.com/jcremerius/Change-Detection-in-Dynamic-Event-Attributes.

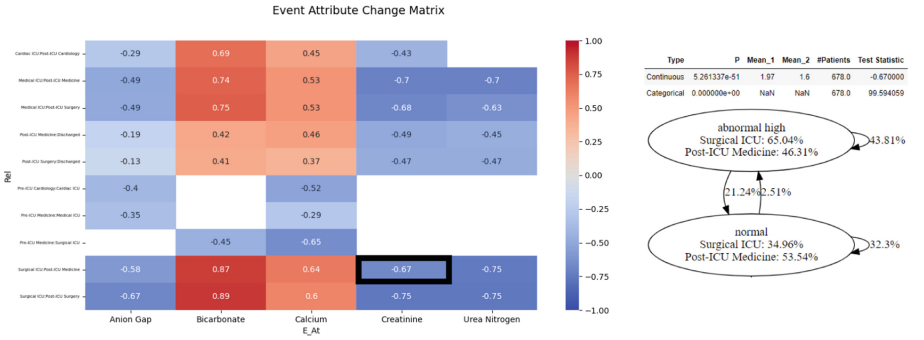

Fig. 2. Change Detection Analysis, illustrating an Event Attribute Change Matrix with the significant event attribute changes and a detailed view of one cell with the connection to the respective categorical event attribute.

$CDA_L(cac)$, which is the RBC value for continuous event attributes. The colour of the cell illustrates the change direction, where blue is decreasing and red is increasing. The cell is blank, if there is no statistical evidence for a change according to the given significance threshold α, which is 0.05 in this case.

The transition between the department visits is shown as relations on the y-axis of the matrix. Emergency department is not listed, as it does not contain any *dynamic* event attributes. The matrix shows, that the laboratory values change differently dependent on the patient's progress through the hospital. For example, we observe no value changes of creatinine between pre-ICU and ICU treatment, whereas it decreases after the ICU stay significantly. On the other hand, the values of calcium tend to decrease in the ICU and increase after that. The developed artefact allows displaying significant changes in a process model, which is shown in Fig. 3, presenting significant changes of calcium.

One can also analyse a change analysis cell in more detail by clicking on it in the matrix, which is shown in Fig. 2 on the right-hand side, where the cell marked with the black box is selected. The figure shows the test results in more detail and illustrates the event attribute connection EAC_L, where the cell of the continuous event attribute "Creatinine" is connected to the respective categorical event attribute "Abnormal Creatinine". The graph nodes show the respective categories, which are "abnormal high" and "normal". The arrows are annotated with the amount of samples changing their state. As the degree of change is high with 21% from "abnormal high" to "normal", the categorical test ended up with a p-value so close to 0 that it is displayed as being 0. This shows the importance of the test-statistic, because the p-value only says, that a change is present, but not how high the degree of change is. Thus, the change in the continuous event attribute results in a change in the categorical event attribute as well.

To verify the attribute value changes, we looked into medical literature and asked a medical expert for consultation. Urea nitrogen and creatinine are estab-

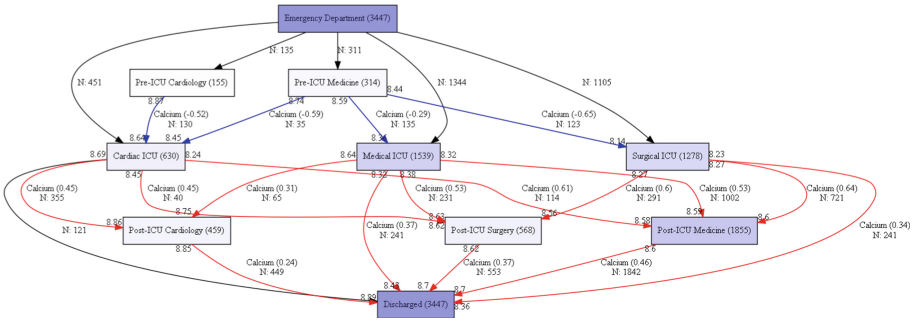

Fig. 3. Directly-Follows Graph enhanced with event attribute changes. The edge labels show the event attribute name with its RBC value and sample size. The colours illustrate the value direction, where blue is decreasing and red is increasing. The ends of the edges show the mean value of the event attribute at the respective activity. (Color figure online)

lished parameters for renal recovery and are expected to decrease after ICU treatment [22]. Additionally, bicarbonate use in the ICU for treatment of anion-gap metabolic acidosis avoids the need for dialysis, which is generally the first-line therapy for acidosis [13]. That explains the increase of bicarbonate and the decrease of anion gap after ICU treatment. The value behaviour of calcium is an interesting observation, as it decreases in the ICU and increases after that, resulting in no significant change between pre-ICU and post-ICU treatment. Together with the medical expert, we found out, that decreased calcium levels (Hypocalcemia) are expected in ICU patients [3], which explains that development. Other attributes not being shown in Fig. 2 were also discussed, such as the glucose value, which did not make much sense, as it tends to change frequently. These attributes require a more fine granular process to make sense for observation through a process. However, the event attributes mentioned above do not tend to change frequently and can be compared department wise.

Another observation was, that patients visiting surgical departments have a stronger tendency to value changes in anion gab, bicarbonate, and calcium, represented by a higher RBC value, which could also be confirmed by the medical expert.

This presentation shows, that *dynamic* event attribute changes with their direction of change can be identified, allowing to derive additional insights out of data stored in event logs.

6 Discussion

This paper proposes an approach to detect changes in *dynamic* event attributes through the process by applying statistical tests on event attributes, relations, and trace variants. With that, we provide a method to analyse the behaviour of

dynamic event attributes and allow identifying in which activities value changes occur.

We have shown an example use case in the healthcare domain and could confirm expected laboratory value behaviour, which was evaluated with a medical expert. As statistical tests are broadly accepted in the medical domain, it was possible to explain how we detect changes to the medical expert, who could understand the p-value and test statistics. We discussed, that a more fine granular process could bring additional insights, such as the comparison of different treatment paths and their laboratory value developments, allowing to evaluate, if different treatment activities have different effects on the patient's state.

However, we see potential for other application domains and do not want to limit the application to the healthcare domain. For example, other data intensive processes, such as manufacturing processes with sensor data, like temperature or vibration, could be of interest when looking at different manufacturing steps of one or multiple machines.

This contribution suggests identifying changes in three dimensions, which leads to a high amount of statistical tests conducted. Thus, we see one limitation in the exploration of changes, which is so far solved by looking at the statistical significant changes only from a two-dimensional perspective. Loops bring more complexity as well by adding more trace variants and relations when one is interested in comparing different loop iterations. Therefore, other perspectives or methods reducing cognitive load could be more suitable for different use cases. For example, when analysing loops, one could cluster the respective loop iterations according to their changing behaviour. The same holds for trace variants, which could be clustered as well.

Furthermore, the changes could be described in more detail by considering other aspects, such as time, resources, or other event attributes. For example, the longer one activity takes, the higher the difference between activities or the other way around. Additionally, changes in event attributes could be correlated with each other, such as creatinine and urea nitrogen in the evaluation.

The usage of statistical tests enables a detailed analysis of two samples, but requires a sufficient sample size as well. In general, the higher the sample size, the better the expressibility (power) of the test. Additionally, these tests cannot say that there is a guaranteed change for any given process instance, but can only give an indication that there is a non-random change in the given samples. Thus, there are almost always cases showing a changing behaviour and others do not. Understanding why some change and others do not is also not covered by us.

It should also be noted, that the statistical tests detect changes which go into one direction, such as from normal to high, resulting in a different distribution of the categories or continuous values. However, when we have changes in both directions, such as 50 from normal to high and 50 from high to normal, the marginal proportion is the same and no change would be detected. The same holds for continuous tests, where the RBC value would be close to 0 in this case. As the goal of this paper is to derive a common behaviour of *dynamic*

event attributes in the process, this property suits us well. However, it might be interesting to investigate this kind of change and derive characteristics of increasing and decreasing cases.

In general, we see different use cases for change detection in *dynamic* event attributes. Besides exploring changes, one could also use this method to derive interesting variables for time-series machine learning tasks, such as process outcome prediction, by identifying process sensitive event attributes. Additionally, the changing behaviour could be used as a feature for decision mining, trace clustering or concept drift detection.

7 Conclusion and Future Work

This contribution researches methods to detect changes in *dynamic* event attributes from a three-dimensional perspective, represented as a change detection cube. This allows to understand the process behaviour of their actual values, as it can be seen between which process activities the values change. We see this method as a step forward to connect data-science with process science, allowing an even more comprehensible analysis of the data represented in event logs.

Future work could focus on enhancing the methodology by explaining the changes in more detail, for example, the correlation with other event attributes, such as time, or deriving characteristics of changing and non-changing cases. Additionally, other dimensions of change could be researched and evaluated regarding their suitability for different use cases. Lastly, the analysis of looping behaviour and trace variants could be improved by applying clustering, for example.

References

1. Process Mining. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4_16
2. Adams, J.N., van Zelst, S.J., Quack, L., Hausmann, K., van der Aalst, W., Rose, T.: A framework for explainable concept drift detection in process mining. In: Polyvyanyy, A., Wynn, M.T., Van Looy, A., Reichert, M. (eds.) BPM 2021. LNCS, vol. 12875, pp. 400–416. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-85469-0_25
3. Afshinnia, F., Belanger, K., Palevsky, P.M., Young, E.W.: Effect of ionized serum calcium on outcomes in acute kidney injury needing renal replacement therapy: secondary analysis of the acute renal failure trial network study. Ren Fail **35**(10), 1310–1318 (2013)
4. Aminikhanghahi, S., Cook, D.J.: A survey of methods for time series change point detection. Knowl. Inf. Syst. **51**(2), 339–367 (2017)
5. Armstrong, R.A.: When to use the Bonferroni correction. Ophthal. Physiol. Opt. **34**(5), 502–508 (2014)
6. Bano, D., Zerbato, F., Weber, B., Weske, M.: Enhancing discovered process models with data object lifecycles. In: 2021 IEEE 25th International Enterprise Distributed Object Computing Conference (EDOC), pp. 124–133 (2021)

7. Berti, A., et al.: Process mining for python (pm4py): bridging the gap between process- and data science. CoRR abs/1905.06169 (2019). http://arxiv.org/abs/1905.06169
8. Chaudhuri, S., Dayal, U.: An overview of data warehousing and OLAP technology. SIGMOD Rec. **26**(1), 65–74 (1997)
9. Cooper, L.B., et al.: Serum bicarbonate in acute heart failure: relationship to treatment strategies and clinical outcomes. J. Card Fail **22**(9), 738–742 (2016)
10. Cremerius, J., Weske, M.: Data-enhanced process models in process mining (2021). https://arxiv.org/abs/2107.00565
11. Cremerius, J., Weske, M.: Supporting domain data selection in data-enhanced process models. In: Wirtschaftsinformatik 2022 Proceedings 3 (2022)
12. Ibanez-Sanchez, G., et al.: Toward value-based healthcare through interactive process mining in emergency rooms: the stroke case. Int. J. Environ. Res. Public Health **16**(10), 1783 (2019)
13. Jaber, S., et al.: Sodium bicarbonate therapy for patients with severe metabolic acidaemia in the intensive care unit (BICAR-ICU): a multicentre, open-label, randomised controlled, phase 3 trial. Lancet **392**(10141), 31–40 (2018)
14. Johnson, A., Bulgarelli, L., Pollard, T., Horng, S., Celi, L.A., Mark, R.: Mimic-iv (2020). https://doi.org/10.13026/A3WN-HQ05
15. Kerby, D.S.: The simple difference formula: An approach to teaching nonparametric correlation. Compr. Psychol. **3**, 11.IT.3.1 (2014)
16. de Leoni, M., van der Aalst, W.: Data-aware process mining: discovering decisions in processes using alignments. In: Proceedings of the 28th Annual ACM Symposium on Applied Computing, p. 1454–1461. SAC 2013. Association for Computing Machinery, New York, NY, USA (2013)
17. de Leoni, M., van der Aalst, W., Dees, M.: A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. Inf. Syst. **56**, 235–257 (2016)
18. Mannhardt, F., de Leoni, M., Reijers, H.: The multi-perspective process explorer. In: CEUR Workshop Proceedings, vol. 1418, August 2015
19. Nguyen, H., et al.: Multi-perspective comparison of business process variants based on event logs. In: Conceptual Modeling, pp. 449–459. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-030-00847-5_32
20. Parab, S., Bhalerao, S.: Choosing statistical test. Int. J. Ayurveda Res. **1**(3), 187–191 (2010)
21. Reed, J.F., Salen, P., Bagher, P.: Methodological and statistical techniques: what do residents really need to know about statistics? J. Med. Syst. **27**(3), 233–238 (2003)
22. Schiffl, H.: Discontinuation of renal replacement therapy in critically ill patients with severe acute kidney injury: predictive factors of renal function recovery. Int. Urol. Nephrol. **50**(10), 1845–1851 (2018). https://doi.org/10.1007/s11255-018-1947-1
23. Sun, X., Yang, Z.: Generalized McNemar's test for homogeneity of the marginal distributions. In: SAS Global Forum, vol. 382, pp. 1–10 (2008)