Leveraging Event Data for Measuring Process Complexity

Maxim Vidgof[1](✉) and Jan Mendling[1,2]

[1] Wirtschaftsuniversität Wien, Welthandelsplatz 1, 1020 Vienna, Austria
maxim.vidgof@wu.ac.at

[2] Humboldt-Universität zu Berlin, Unter den Linden 6, 10099 Berlin, Germany
jan.mendling@hu-berlin.de

Abstract. Complexity is an important aspect of business processes. Numerous metrics have been introduced to measure process complexity, however, existing metrics view processes merely as sequences of activities, disregarding the corresponding data. This is a major omission since much of the complexity of business processes stems from the variation of data that is associated with it. In this paper, we refer to recent research on how behavioral complexity of business processes can be defined. More specifically, we extend entropy-based complexity metrics such that they are capable of capturing the variation of event data. We provide some first insights into the implications of applying these newly proposed metrics.

Keywords: Process complexity · Event data · Graph entropy

1 Introduction

The central objectives of Business Process Management (BPM) is the improvement of process performance [5]. One of the factors hampering process performance is complexity. For this reason, it is key prerequisite for process improvement to be able to, first, measure process complexity in an appropriate way and, then, define measures to address it.

Prior research has contributed to our understanding of how process complexity can be measured based on event logs [1]. However, it is an important omission that these event-log measures are defined purely based on the behaviour aspects of event sequences. This neglects observations from work on process standardization that identified eleven theoretical dimensions that are tied to process standardization [13]. Notably, two of them relate to inputs & outputs and to data. Also other fields like Machine Learning acknowledge the importance of data complexity and its impact on results of, e.g., prediction models. So far, there is no process complexity measure that reflects the complexity of data.

In this paper, we address this research problem and discuss how the complexity of process-related data can be integrated with process complexity measures. To this end, we extend an existing entropy-based process complexity metric with

M. Montali et al. (Eds.): ICPM 2022 Workshops, LNBIP 468, pp. 84–95, 2023.
https://doi.org/10.1007/978-3-031-27815-0_7

aspects of process-related event data. We provide a preliminary evaluation on an artificial as well as a real-life event logs and discuss directions for future work.

The remainder of this paper is organized as follows. Section 2 introduces existing complexity metrics and their limitations. Section 3 presents our approach. Section 4 shows the preliminary evaluation, its discussion and limitations of this paper. Section 5 concludes the paper.

2 Background

This section discusses the background of our research. We first reflect upon prior contributions to measuring process complexity based on event logs. Then we turn to approaches from neighboring fields on how to measure data complexity.

2.1 Process Complexity Metrics

Over the years, several process complexity metrics have been introduced. They have focused on one of the following aspects: size, variability and distance. Size-based metrics count properties of an event log, such as the number of events, traces, average trace length, etc. Metrics related to variability show the variation in the event log, they often build transition matrices based on directly-follows relations observed in the event log [1] or use the number of unique sequences in the log [12]. Distance-based metrics measure the difference between traces in the event log, e.g. affinity of two event sequences, i.e. the extent to which the directly-follow relations of the sequences overlap [6].

Recently, complexity metrics based on graph entropy have been introduced: *variant entropy, normalized variant entropy, sequence entropy* and *normalized sequence entropy* [1]. The latter one has been proven to capture all the three aspects of process complexity and also correlate with the complexity of the discovered process models. A major drawback of all these metrics is, however, that they are sill solely focused on the behavior and ignore event data.

2.2 Data Complexity Metrics

Machine Learning domain has a long history of measuring data complexity. This is not surprising as the complexity of the input data is expected to influence the performance of the predictions. Researchers in the Machine Learning domain generally used three kinds of complexity metrics proposed in [7] and [8]:

1. Measure of overlap: Fisher's discriminant ratio (F1), volume of overlap region (F2), feature efficiency (F3).
2. Measure of class separability: The minimized sum of the error distance of a linear classifier (L1), training error of linear classifier (L2), the ratio of average intra/inter class nearest neighbor distance (N2), leave one out error rate of the 1-NN classifier (N3).

3. Measure of geometry, topology and density of manifolds: Nonlinearity of linear classifier by Linear programming (L3), nonlinearity of 1-NN classifier (N4), space covering by ϵ-neighborhoods (T1), average number of points per dimension (T2), density (D1).

These metrics have been widely used for different tasks, e.g. [9] uses them for the selection of suitable normalization technique for a particular classification problem, [10] uses some of the data complexity measures to estimate the significant intervals for oversampling.

However, such complexity metrics have limited applicability in the process mining domain. First, these metrics measure assume the data has class labels and, moreover, implicitly assume that these labels are fixed. They then measure complexity with respect to this classes, e.g. overlap between classes or class separability. While such metrics seem useful for some applications, e.g. categorical outcome prediction in Predictive Process Monitoring, they would provide little help when the data is not split into classes at all or these classes are not relevant for the problem at hand, e.g. remaining time prediction. Furthermore, even if useful, such metrics would give different results for the same data depending on the problem, e.g. if the same dataset is used for categorical outcome and next activity prediction, the classes for two problems would be different and thus the complexity measurements. Second, a study has shown that while some of the data complexity metrics provide useful information, e.g. are connected with classifier performance, they cannot be used to compare different datasets wihh different characteristics [2]. Finally, these metrics ultimately treat the data as a sample of independent observations, ignoring the process notion and the corresponding relations between the data points, i.e. events. This might be a critical drawbacks for process mining applications.

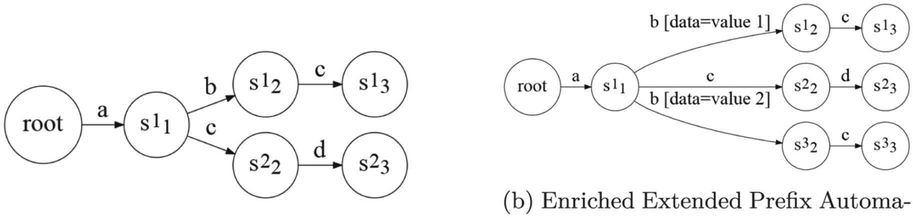
While the former drawbacks could theoretically be fixed by taking a step back and using entropy or Gini index of the entire dataset as a metric of complexity, the latter problem of losing the process notion would still persist. Thus, our goal in this paper is to extend an existing process complexity metric with the capability of considering data complexity as well.

3 Approach

In order to incorporate data complexity into a process complexity metric, we extend the existing complexity metrics based on graph entropy [1]. First, we introduce Enriched Extended Prefix Automata that include event data. Second, we introduce cumulative complexity metrics that allow to study in more detail how the complexity changes as new events are observed.

Extended Prefix Automata (EPA), introduced in [1], are a representation of business processes without abstraction. However, in its basic form, an EPA only contains information about the behavior. It means, the transitions between states are only labeled with activity labels, and the events in the EPA only contain activity label, case ID timestamp and a link to the predecessor event.

Enriched EPAs, or EEPAs for short, are EPAs enriched with other event data. In essence, it is achieved in the following way. First, an event in the EEPA does not only contain its basic attributes (case ID, activity label, timestamp and predecessor) but also an Attribute Container, where all trace and event attributes are stored. The distinction between trace and event attributes is made in order to prevent name collisions, otherwise these attributes are treated equally, and each event in the trace contains all trace attributes of its trace. The EEPA containing such events is then a state automaton with guards. Thus, the transitions of an EEPA are labeled not only with activity labels but also with corresponding attribute values. In order to follow a transition on EEPA, the event thus should have not only a matching activity label but also matching attributes. In case there is no matching transition, a new partition with new state and a corresponding new transition is added to an EEPA, in the same way as a new partition is added to an EPA on a previously unobserved prefix. One can then apply the same complexity metrics to an EEPA as to an EPA – *variant entropy, normalized variant entropy, sequence entropy* and *normalized sequence entropy* – but they will now take data into account as well because the underlying EEPA is partitioned based on behavior and the data.



(a) Extended Prefix Automaton (EPA)

(b) Enriched Extended Prefix Automaton (EEPA)

Fig. 1. Difference between an extended prefix automaton and an enriched extended prefix automaton built from the same event log.

Figure 1 shows the difference between an EPA and an EEPA built from the same event log $L = [\langle a, b, c\rangle^2, \langle a, c, d\rangle]$ where in of the $\langle a, b, c\rangle$ traces the activity b carries event data $value1$ and in the second one $value2$. While the EPA only has 2 partitions and both $\langle a, b, c\rangle$ traces belong to partition 1, the corresponding EEPA makes difference between these two traces based on the event data and thus puts these traces in 2 different partitions and has 3 partitions in total. This necessarily means an EEPA would have more states and partitions than an EPA built from the same log, leading to higher variant entropy. An EEPA is also expected to have higher sequence entropy and normalized sequence entropy as it has more partitions with the same number of events. This is, however, not necessarily the case for normalized variant entropy exactly because an EEPA has more partitions but at the same time more states than a corresponding EPA.

It is important to note though that attribute selection plays a crucial role in building an EEPA. If an event log is rich in attributes, including them all might lead to an EEPA where every trace is represented with a separate partition, which is not too insightful. First, it is recommended to use only categorical variables, since numeric ones have a much lower probability to coincide on different events. Thus, existing numerical attributes should either be disregarded or transformed into categorical bins, where the size of the bins also has significant impact and thus should be chosen with caution. Second, for the same reason it might be meaningful to also perform similar binning even on categorical attributes in case they have a large number of values. Finally, one should consider based on the value ranges as well as the attribute description whether the attribute is relevant at all and possibly reduce the pool of attributes used.

Our claim is that data adds an additional layer of complexity on top of behavior. Thus, it is interesting to observe how complexity of a process increases over time by adding new data values while the behavior stays exactly the same. In order to do so, we also introduce the concept of cumulative complexity. That is, we want to not only measure the total complexity of the entire log but also want to see how it evolved, i.e. how new behavior and/or data influenced the complexity. To this end, we introduce the concept of an *active event* which is an event in the (E)EPA that happened (arbitrarily far in the past) before some threshold timestamp, i.e. an event having a timestamp smaller than some given threshold. Similarly, an *active state* is a state in an (E)EPA that includes at least one active event. Then we only consider active events/states for measuring sequence and variant entropy, respectively.

By gradually increasing the threshold, we can add more and more events to the (E)EPA as if we were building it in real time and get the complexity metrics at each point in time, e.g. at the end of each week, month, year, etc. It is equivalent to measuring complexity after each period and then continuing to build the (E)EPA, however, can be repeated indefinitely with different time granularity over the same automaton. In addition, it enables to use two kinds of normalization.

Normally, the variant and sequence entropy are calculated using all states/events in an EPA. Then, the normalization is done by dividing the metric by $|X|log(|X|)$, where $|X|$ is the total number of states/events in the EPA. When normalizing cumulative metrics, however, there are two possibilities. While variant and sequence entropy are obviously measured over active states/events, when it comes to normalization these metrics can be divided by either the number of active states/events or by the total number of the states/events in the full (E)EPA (containing the full event log). The former option would be indeed equivalent to measuring normalized metrics at the end of each time period, and the latter one allows to observe cumulative growth of the normalized metrics over time. These 6 cumulative complexity metrics – *variant entropy, variant entropy normalized over active states, variant entropy normalized over all states, sequence entropy, sequence entropy normalized over active events, sequence entropy nor-*

malized over all events – equip us with the means of observing how new events (carrying new behavior and/or data) influence the complexity.

4 Evaluation

In this section, we present the preliminary evaluation of our approach. First, with an artificial event log and then with real-life event logs. Next, we discuss our results and report current limitations. The implementation is publicly available on GitHub[1].

4.1 Artificial Event Log

We use an example loan process application from [5] shown in Fig. 2. We manually created an event log with 10 traces. All events have a user associated with it. The event *Loan application received* is always associated with a user *System*, which is not considered further. The events associated with the activity *Assess loan risk* have a categorical variable *Risk* and the events associated with the activity *Appraise property* have a numerical variable *Price*. In the first month, there are 4 traces following 2 variants with 1 user and 2 risk levels. In the second month, additional 2 variants are introduced. In the third month, additional user is added who follows the same variants. Finally, in the fourth month additional risk level is added, while the users and variants are kept the same. The prices vary over the entire event log.

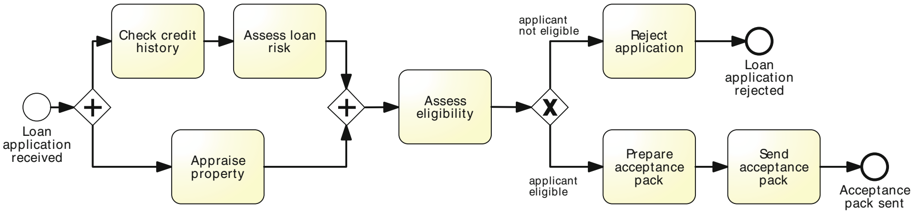


Fig. 2. Loan process, reused from [5].

We then computed the four complexity metrics – variant entropy, normalized variant entropy, sequence entropy and normalized sequence entropy – for this log but varied the data that we took into consideration. Table 1 shows the results. The first row corresponds to an EPA that only considers the behavior and uses no data. The second row corresponds to an EEPA that only uses the *User* variable of the events, and so on. We also split the numeric price into 3 bins to show how numeric data can also be incorporated.

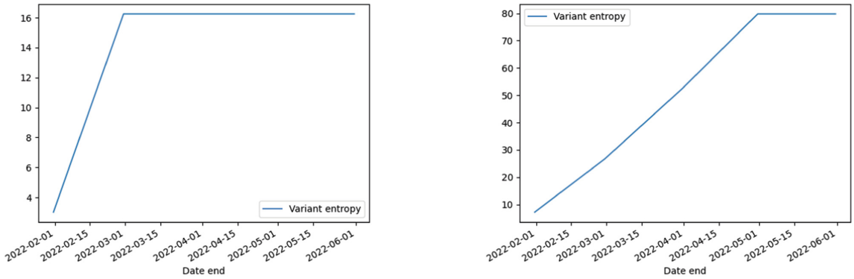
As we can see, the complexity of the EEPAs using additional data on top of behavior is considerably higher than the complexity of an EPA. We also see

[1] https://github.com/MaxVidgof/process-complexity.

Table 1. Complexity of the artificial event log using different amount of event data.

Data	Variant entropy	Normalized variant entropy	Sequence entropy	Normalized sequence entropy
None	16.25	0.4	47.16	0.17
User	42.58	0.53	95.64	0.35
User & Risk	80.0	0.56	118.52	0.44
User & Risk & Price (binned)	109.12	0.59	135.94	0.50
User & Risk & Price (numeric)	109.12	0.59	135.94	0.50

that all metrics continue to grow as we consider more variables since it leads to higher partition counts in the EEPA.



(a) Cumulative variant entropy of simple EPA

(b) Cumulative variant entropy of EEPA with *User* and *Risk*

Fig. 3. Cumulative variant entropy for simple EPA and an enriched EPA with *User* and *Risk* event data.

Cumulative complexity metrics also enable us to observe how the complexity changes as new events are observed. For instance, Fig. 3 shows the development of variant entropy. When only behavior is considered (Fig. 3a), the complexity stops growing as soon as all variants are observed. When the event data is also taken into account (Fig. 3b), however, variant entropy continues to grow even when all variants are observed because of the new data: new user introduced in March and new risk level added in April.

4.2 Real-Life Event Logs

We also conducted a preliminary evaluation of our technique on the Business Process Intelligence Challenge logs from years 2012 [4], 2013 [11] and 2015 [3].

For each event log, we did the following. First, we filtered the event logs such that they contain only categorical attributes, i.e. we removed all attributes having numeric values or representing dates. Second, we generated an Extended Prefix Automaton from each log. We will further refer to these automata as simple EPAs. We calculated variant entropy, normalized variant entropy, sequence entropy and normalized sequence entropy for each of these simple EPAs. Furthermore, we calculated cumulative metrics – variant entropy, variant entropy normalized over active states, variant entropy normalized over all states, sequence entropy, sequence entropy normalized over active events, sequence entropy normalized over all events – for each month from the month of the rirst event in the respective log to the month of the last event. Then, we generated Enriched Extended Prefix Automata (enriched EPAs or EEPAs) from the same logs repeated the same procedures, i.e. calculated the 4 total complexity metrics as well as 6 cumulative complexity metrics over time. As a result, for each log we had 4 complexity metrics for the corresponding simple EPA, 4 complexity metrics of the corresponding EEPA, 6 time series of cumulative complexity metrics for the EPA and 6 time series of cumulative complexity metrics for the EEPA.

First, we wanted to evaluate whether the new metrics adequately depict the additional complexity introduced by event data. Two-sided t-test reported significant difference between normalized sequence entropy of the enriched and the simple EPA. In all cases, except the normalized variant entropy, the metric for the enriched EPA was greater than of its simple counterpart. Thus, we also performed one-sided t-tests. While the p-values were considerably smaller in all cases, normalized sequence entropy still remained the only one with significant difference (p-value 0.01). Interestingly, difference in variant entropy was also close to significant (p-value 0.09). More observations might render it significant as well.

For each of the logs we also compared the time series of the 6 cumulative complexity metrics measured with the simple and enriched EPAs. Here, we not only performed two-sided t-tests that would say whether the difference in means of the two samples is significant but also performed two-sided Kolmogorov-Smirnov tests that would assess whether two samples come from the same continuous distribution. It is important to note that some event logs carry events from before the observation period, e.g. BPIC 2012 includes some events from late 2011. This introduces periods having only 1 event and thus entropy metrics equalling 0, which might influence the value distribution. Thus, in such cases we also filter the metrics for the corresponding event log, keeping only non-zero observations. Periods with non-zero observations are naturally the same for the metrics computed with EPA and EEPA.

The results of these tests can be seen in Table 2. The columns in the table represent the metric, the rows are different time series pairs (for a simple and enriched EPA) and cells indicate whether there was a significant difference between two time series. T means t-test reported significant difference and K means Kolmogorov-Smirnov test reported significant difference. We say the difference is significant when the p-value is below 0.05.

Table 2. Differences in cumulative complexity metrics of enriched extended prefix automata and extended prefix automata for real-life event logs. T stands for significant difference reported by t-test, K stands for significant difference reported by Kolmogorov-Smirnov test.

Data	Variant entropy	Normalized variant entropy (active)	Normalized variant entropy (all)	Sequence entropy	Normalized sequence entropy (active)	Normalized sequence entropy (all)
BPIC12	TK				TK	
BPIC13					K	
BPIC13 filtered					TK	
BPIC15_1	TK	K			TK	
BPIC15_2					K	
BPIC15_2 filtered					TK	
BPIC15_3	TK	K			TK	
BPIC15_3 filtered	TK	K			TK	
BPIC15_4		K			K	K
BPIC15_4 filtered	T	K			TK	
BPIC15_5	T				K	
BPIC15_5 filtered	T	K			TK	

As we can see, sequence entropy normalized over active events significantly differs for all event logs with Kolmogorov-Smirnov test and for almost all event logs with t-test. Variant entropy shows significant difference with Kolmogorov-Smirnov test in 4 logs and with t-test in 7 logs. Variant entropy normalized over active states shows significant difference in Kolmogorov-Smirnov test in 6 logs. Finally, sequence entropy normalized over all events shows significant difference with Kolmogorov-Smirnov test in 1 log.

4.3 Discussion

The evaluation on the artificial log shows that the new metrics are capable of highlighting the complexity introduced by new event data. While some of this increased complexity could be uncovered by using existing process complexity metric in conjunction with auxiliary metrics, e.g. the added user could be also spotted with Social Network Analysis and multimple risk levels could be extracted from internal documentation or a BPMS, this would not necessarily work with all data, especially if this data comes from external sources. It is also important to note that while binning indeed allows taking numerical data into consideration, the efficiency of such method largely depends on the granularity, since if set too high it might bring no additional value compared to directly using numerical data.

Evaluation on the real-life logs further confirms these results. Normalized sequence entropy seems to highlight the increase in complexity due to data in the most effective way. This is not surprising as normalized sequence entropy also the only one that significantly correlates with, e.g. model complexity [1] and may just be a better metric.

When it comes to the cumulative metrics, sequence entropy normalized over active events shows best significance, also confirming the above stated ideas. As expected, also the differences in variant entropy are significant. The underlying idea that with the same behavior more distinct data would lead to more branching and more partitions in the EEPA than in the EPA of the corresponding log, which would also logically lead to higher variant entropy, seems to have found its confirmation. The fact that such effect is observed not in all event logs may be attributed to lower difference in data in the other logs. However, it needs further and more detailed investigation.

4.4 Limitations

This paper is a work in progress and thus suffers from a range of limitations. First, there are limitations in terms of the implementation. While it is capable of handling smaller event logs, it does not scale well, thus restricting evaluation and, more importantly, real-life application of the metrics. Second, the attribute selection in the real-life log evaluation was superficial. It considered all of the categorical attributes and none of the numeric ones. More thorough selection of categorical attributes as well as meaningful binning of the numeric ones is expected to give more adequate results. Third, only basic statistical methods were used for the analysis, especially when it comes to cumulative metrics. While they are definitely time series, no analysis techniques specific to this kind of data has been applied yet.

5 Conclusion and Future Work

Complexity is important aspect of business processes that requires thorough studying. While existing process complexity metrics are successful in measuring behavioral complexity of the processes, they completely ignore the data associated with the events and thus miss the next layer of complexity that is added by this data. On the other hand, there exist data complexity metrics, however, they do not have the notion of process and also have other implicit assumptions that limit their usability in process mining.

In this paper, we proposed a set of new process complexity metrics that take into account event data in addition to behavior. These metrics are based on existing complexity metrics for Extended Prefix Automata but use an updated version of such automata – Enriched EPAs. We conducted preliminary evaluation on a small artificial example as well as on a set of real-life event logs.

The initial results show that our new metrics capture the data complexity in addition to behavior complexity. We plan to extend our evaluation on more real-life logs, improve the implementation and analyse the results in more detail.

References

1. Augusto, A., Mendling, J., Vidgof, M., Wurm, B.: The connection between process complexity of event sequences and models discovered by process mining. Inf. Sci. **598**, 196–215 (2022). https://doi.org/10.1016/j.ins.2022.03.072
2. Cano, J.R.: Analysis of data complexity measures for classification. Expert Syst. Appl. **40**(12), 4820–4831 (2013). https://doi.org/10.1016/j.eswa.2013.02.025, https://www.sciencedirect.com/science/article/pii/S0957417413001413
3. van Dongen, B.B.: BPI challenge 2015, May 2015. https://doi.org/10.4121/uuid:31a308ef-c844-48da-948c-305d167a0ec1, https://data.4tu.nl/collections/BPI_Challenge_2015/5065424/1
4. van Dongen, B.: BPI Challenge 2012, April 2012. https://doi.org/10.4121/uuid:3926db30-f712-4394-aebc-75976070e91f, https://data.4tu.nl/articles/dataset/BPI_Challenge_2012/12689204
5. Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management, Second Edition. Springer, Berlin, Heidelberg (2018). https://doi.org/10.1007/978-3-662-56509-4
6. Günther, C.: Process mining in flexible environments. Ph.D. thesis, Technische Universiteit Eindhoven (2009). https://doi.org/10.6100/IR644335
7. Ho, T.K., Basu, M.: Measuring the complexity of classification problems. In: 15th International Conference on Pattern Recognition, ICPR'00, Barcelona, Spain, 3–8 September 2000, pp. 2043–2047. IEEE Computer Society (2000). https://doi.org/10.1109/ICPR.2000.906015
8. Ho, T.K., Basu, M.: Complexity measures of supervised classification problems. IEEE Trans. Pattern Anal. Mach. Intell. **24**(3), 289–300 (2002). https://doi.org/10.1109/34.990132
9. Jain, S., Shukla, S., Wadhvani, R.: Dynamic selection of normalization techniques using data complexity measures. Expert Syst. Appl. **106**, 252–262 (2018). https://doi.org/10.1016/j.eswa.2018.04.008
10. Luengo, J., Fernández, A., García, S., Herrera, F.: Addressing data-complexity for imbalanced data-sets: a preliminary study on the use of preprocessing for C4.5. In: Ninth International Conference on Intelligent Systems Design and Applications, ISDA 2009, Pisa, Italy, November 30–2 December 2009, pp. 523–528. IEEE Computer Society (2009). https://doi.org/10.1109/ISDA.2009.233
11. Steeman, W.: BPI Challenge 2013, incidents, April 2013. https://doi.org/10.4121/uuid:500573e6-accc-4b0c-9576-aa5468b10cee, https://data.4tu.nl/articles/dataset/BPI_Challenge_2013_incidents/12693914
12. van der Aalst, W.M.: Process Mining: Data Science in Action, Second Edition. Springer, Berlin, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4
13. Wurm, B., Schmiedel, T., Mendling, J., Fleig, C.: Development of a measurement scale for business process standardization. In: European Conference on Information Systems (ECIS 2018). Association of Information Systems (2018)