Association for Information Systems

AIS Electronic Library (AISeL)

A Framework of Business Process Monitoring and Prediction Techniques

Frederik Wolf
Westfälische Wilhelms-Universität Münster

Jens Brunk
Westfälische Wilhelms-Universität Münster

Jörg Becker
Westfälische Wilhelms-Universität Münster

Follow this and additional works at: https://aisel.aisnet.org/wi2021

A Framework of Business Process Monitoring and Prediction Techniques

Frederik Wolf, Jens Brunk, and Jörg Becker

University of Münster – ERCIS, Münster, Germany
{f_wolf09, jens.brunk, becker}@ercis.uni-muenster.de

Abstract. The digitization of businesses provides huge amounts of data that can be leveraged by modern Business Process Management methods. Predictive Business Process Monitoring (PBPM) represents techniques which deal with real-time analysis of currently running process instances and also with the prediction of their future behavior. While many different prediction techniques have been developed, most of the early techniques base their predictions solely on the control-flow characteristic of a business process. More recently, researchers attempt to incorporate additional process-related information, also known as the process context, into their predictive models. In 2018, Di Francescomarino et al. published a framework of existing prediction techniques. Since the young field has evolved greatly since then and context information continue to play a greater role in predictive techniques, this paper describes the process and outcome of updating and extending the framework to include process context dimensions by replicating the literature review of the initial authors.

Keywords: Business Process, Prediction, Techniques, Predictive Business Process Monitoring

1 Introduction

Today's economy is highly volatile and uncertain developments, such as the current COVID-19 crisis, pressure organizations to be able to adapt and improve their business processes immediately [1]. Consequently, the domain of predictive business process monitoring (PBPM) gained momentum in business process management (BPM) in the last few years [2]. PBPM leverages prediction techniques to predict and improve operational business processes. PBPM techniques predict the future behavior of a business process during its execution [3, 4] based on predictive models, which were constructed from the historical process event logs [5, 6]. PBPM techniques can address a variety of goals, such as next activity [7], process outcome [8], remaining processing time [9] or pre-determined risks, and apply several different technologies [10]. Due to the complex nature of the often Machine Learning-based (ML) PBPM techniques, their different goals and input requirements, [10] developed a value-adding framework in 2018, which classifies PBPM techniques and supports researchers and practitioners in selecting appropriate PBPM techniques for their endeavor.

Most of the (early) PPM techniques base their analyzes and predictions solely on the control-flow characteristic of a business process, i.e. the process events [10].

Since then, researchers continuously attempt to conceptualize and incorporate additional process-related information, also known as the process context, into their predictive models. The context can be defined as the "minimum set of variables containing all relevant information that impact the design[, implementation] and execution of a business process" [11, p.154]. Context information, originating from sources external [12, 13] or internal (e.g. [14, 15]) to the business process, can improve process predictions since it adds valuable information to the predictive models [5].

For example, [16] confirm a significant relationship between the representation of an event log's context attributes and a DNN's predictive quality in a next activity prediction task. [15] and [17] additionally use a resource attribute of a process log as context information. [14] improve their prediction results by incorporating multiple additional context attributes. [18] explore the effect of different previously used context attributes on a deep learning neural network and compare as well as benchmark their results with previous publications.

Since the initial publication of the above-mentioned Process Prediction Technique Framework (PPTF) by Di Francescomarino et al. in 2018, many new techniques have been developed and existing techniques might have been enhanced. Additionally, the trend of incorporating additional context attributes into the predictive techniques adds another layer of complexity to the task of selecting an appropriate technique for a PBPM endeavor. Therefore, a gap exists for an updated PPTF that also includes detailed information about the capability of incorporating context information. Our research goal (RG) therefore states: *Update the Process Prediction Technique Framework by Di Francescomarino et al. (2018) and extend it by a dimension on context information.*

Section 2 introduces the initial PPTF and other related work. Section 3 describes the applied literature search process to update and extend the PPTF, before the new PPTF is presented in section 4. Section 5 then concludes our work and gives an outlook on future work.

2 Background

The PPTF is based on a literature review and classifies existing PBPM techniques in the dimensions prediction type, input data, tool availability, domain, and family of algorithm [10]. The dimension prediction type includes the prediction goal of the PBPM technique. Prediction goals were identified to be time prediction, categorical outcome(s), sequence of next outcomes/values, risk, inter-case metrics or cost. The inputs required by a technique are captured in the input dimension. Generally, techniques take an event log as input that must contain certain information. Some techniques require additional inputs, like a labeling function. If a tool was developed to support a PBPM technique, it is captured in the dimension tool support. A developed tool facilitates using, evaluating and understanding the technique. A technique can be implemented in a standalone tool or as a plug-in for an existing tool. PBPM techniques are generally evaluated using event logs, which can either be synthetic or recorded from a real-life process. When selecting a technique for implementation, it is better if the technique is validated with an event log from a similar domain as the domain it will be implemented in. Therefore, the domain of the event log used for validation is captured in the domain dimension. PBPM

techniques are usually based on a specific algorithm. For some applications, certain types of algorithms might have benefits over others. Therefore, the family of algorithm is captured in the respective dimension. The framework can be used by both practitioners and researchers. For practitioners, the structure of the framework allows identifying the most suitable PBPM technique for a given scenario. For researchers, it offers a clear classification and characterization of existing PBPM techniques [10].

Many current PBPM techniques incorporate additional process information into their predictive models. Additional process information is also known as the context of a process [19]. It can either be contained in the event log as data elements, or stem from external sources. The context information of a process can consist of several context attributes, which are the specific entities for which information is available. To understand the meaning of context, which types of values a context attribute can have and which technical implications the characteristics of context have, [19] developed a Taxonomy for Business Process Context Information. The taxonomy characterizes context attributes in the dimensions time, structure, origin, relevance, process relation, and runtime behaviour. Values of context attributes of a process can either be known at the beginning of a process instance, become known during runtime or be predicted. These characteristics are contained in the time dimension. The structure dimension refers to the format of the context attribute. Information can either be available in a structured, semi-structured, or unstructured format. The origin dimension captures, from which source a context attribute stems. Immediate context attributes are required to execute a process. Information about the internal environment of the organization is stored in internal attributes. Context attributes that have an indirect influence on a process but are within the business network of the organization are external attributes. Any context from outside the business network of the organization is the environment. Some context attributes have a stronger influence on a process than others. The extent of the influence of a context attribute is captured in the relevance dimension. A context attribute can influence various elements of a process. In that sense, context attributes can influence activities, events, the control flow, or artefacts of a process. Which element(s) are influenced by a context attribute is captured in the Process Relation dimension. Some context attributes change their values during the execution of a process, meaning they are dynamic. Context attributes maintaining their value are referred to as being static. Whether a context attribute is static, or dynamic is contained in the Runtime Behaviour dimension.

3 Research Method

The review protocol applied by the authors in the literature review of the original PPTF followed the guidelines given by [20]. Towards achieving our RG, we adapt and replicate their literature review process. First, we design the research protocol including the definition of guiding questions, electronic databases used, the search string, and the processing of results. In the second step, we conduct the literature search, identify the final list of papers and extract relevant information from them [10]. Di Francescomarino et al. defined four guiding questions (referred to as GQ1, GQ2, GQ4 and GQ5 in our work) to lead the development of the original framework. For the extension of the

framework we add GQ3 to the set of questions because we anticipate that more recent techniques can process contextual inputs beyond the traditional event log:

- GQ1: What aspect do techniques for PPM predict?
- GQ2: What input data do they require?
- GQ3: Which additional input data do they use?
- GQ4: What are their main families of algorithms?
- GQ5: What are the tools supporting them?

The databases used for the literature review are *Scopus, SpringerLink, IEEE Xplore, ScienceDirect, ACM Digital Library* and *Web of Science*. These databases are the same as in the original literature review and were selected as they cover publications in the research field of Computer Science [10]. The original authors used the search string *("predictive" OR "prediction") AND ("business process" OR "process mining")* for running their queries in October 2017. They state that, after removing duplicates, the search in all the databases named above resulted in 779 papers [10]. However, running the same search string in the same databases in February 2020, yielded over 90.000 results. We could not identify the reasons for this discrepancy either on our own or together with the original authors. In our literature review, we added the term *("method" OR "algorithm" OR "technique")* with an AND connector to the search string to focus the results on those studies presenting a PBPM technique and exclude high-level studies on the general topic. Therefore, the final search string is: *("predictive" OR "prediction") AND ("business process" OR "process mining") AND ("method" OR "algorithm" OR "technique")*. To further narrow down the results of the search, we apply selected filters. First, we exclude all papers published before 2017, as these should already be contained in the original framework. Second, the subject area is narrowed down to "Computer Science", the sub-discipline is selected to be "Data Mining and Knowledge". Lastly, all non-English papers are excluded. However, it is not possible to apply all filters in all databases. We executed the queries on February 8th, 2020. Since the high number of results, we assume that in each database, no further valuable sources can be expected after the first 500 results, sorted by relevance. Table 1 gives an overview of the filters applied, the number of results and the number of results considered for each database.

For the processing of the results, we again proceed very similar to the process of the original authors, which includes seven steps [10]. Since we expect that some papers, which we find in our literature review, are already contained in the original framework (namely those which were published between January and October in 2017), we added step 3 to the process to filter those papers. In the first step, duplicates are removed. Duplicates are defined as papers with the same title and the same authors. Second, results are filtered by the title of the study. All documents that are not proper research papers (e.g. white papers, editorials) and all studies that relate to a different research area are excluded. Third, all studies that are already contained in the original framework are excluded. In the next step, position papers and workshop papers were excluded because results in these studies are often less mature as those in conference papers or journals. Fifth, results are filtered by their abstract, assessing their relevance. In the sixth step, the full texts of the results are accessed and filtered by whether the study proposes a novel technique to the field of PBPM. Finally, additional papers are added via a backward

Table 1. Literature Review: Applied filters, Number of Results and Number of Results considered

Database	Applied Filters	Number of results	Results considered
Springer	Discipline "computer science", subdiscipline "Data Mining and Knowledge Discovery", Language "English", Date 2017 - 2020	2114	First 500 by relevance
ScienceDirect	Date 2017- 2020	2704	First 500 by relevance
Web of Science	Date 2017 - 2020	26	26
Scopus	Subject Area "Computer Science", Language "English", Date 2017 - 2020	206	206
IEEE Xplore	Date 2017 - 2020	1327	First 500 by relevance
ACM Digital Library	Date 2017 - 2020	204	204
Total		6581	1936

search. Table 2 shows the number of results remaining after each step in the literature review process.

Table 2. Literature Review: Processing Steps and Number of Resulting Papers

	Processing Step	Number of results remaining
0	None	1936
1	Remove Duplicates	1840
2	Filter by title of studies	77
3	Remove papers existing in original framework (incl. original framework study)	67
4	Remove workshop papers, position papers	60
5	Filter by abstract	28
6	Filter by full text	25
7	Add papers via backward search	29

4 Process Prediction Technique Framework

The results of our literature review confirm that the majority of more recent techniques incorporate context information (at least partially). In total, 19 of the 27 identified techniques use context information. Most of these approaches leverage context information which is contained within the applied event logs. One technique stands out and aims to incorporate information from outside of the event log. The technique provided by [12]

analyses data on the sentiment of the news media at the time of process execution to add it to the prediction. Feeding the context information into the prediction model is typically done with one-hot encoding, assigning each value of a context attribute a new column in the input vector [e.g. 9, 21]. In addition, [14] use a min-max normalization to encode continuous data features. The authors state, that in the future the approach might even take images as an input. [22] compare predictions with one-hot encoding and predictions with encoding via entity embedding to predictions without adding context information. They find that entity embedding results in more accurate predictions than one-hot encoding. Both approaches outperform the prediction without context information.

Context Information

Context information was already superficially incorporated in the original PPTF. The *Input* dimension is used to shortly describe the inputs needed for a technique. If the technique takes context information as an input, an attribute like event log (with context information) or similar is contained in this dimension. However, this kind of information is not sufficient to select the correct technique if a PBPM project plans on incorporating context information, as the kind of context information supported vary from technique to technique. It is necessary to incorporate a classification of the type of context information into the PPTF.

Therefore, we combine the framework with the Taxonomy of Business Process Context Information [19] as an addition to the *Input* dimension. The taxonomy enables a classification of context information of business processes in the six dimensions *Time, Structure, Origin, Relevance, Process Relation,* and *Runtime Behavior*. The *Time* dimension relates to the point in time at which the context information is known. *Structure* describes the data model of the context information. The source of the context information is contained in *Origin*. *Relevance* classifies the importance of the context information to the business process. In the *Process Relation* dimension, the part of a process to which the context information is connected, is captured. Finally, *Runtime Behaviour* states whether the context information changes throughout a process instance execution or not [19]. Some of the PBPM techniques assume that context information is stored for an entire case (e.g. the loan amount in a credit-granting process) instead of storing it on activity or event level. None of the characteristics in the dimension *Process Relation* fits this assumption. To overcome this, the additional characteristic *Instance* is introduced to the dimension *Process Relation* for the combination of the PPTF and the taxonomy.

In the PPTF, the taxonomy dimensions describe which context information a technique supports. Most approaches work based on machine learning and implicitly or explicitly assign weights to each piece of information. Therefore, the *Relevance* dimension is of little value and is thus neglected in the framework.

Extension of the Technique Framework

Towards the construction of the extended PPMF, the techniques contained in the original framework and the newly identified techniques are inserted. Information on techniques

already contained in the original framework is adopted and enriched with more details on context information. For all new techniques, the characteristics of all dimensions of the extended framework are extracted and inserted into the framework. In the literature search, we identified four techniques that are extended versions of techniques that were already contained in the original framework ([9, 23–25]). In these cases, we removed the older techniques in favor of the new and extended ones. In total, the extended PPTF now contains 77 PBPM techniques. Analogous to Di Francescomarino et al., the framework can be read from left to right. The techniques in the framework are sorted hierarchically by their *Prediction Type, Detailed Prediction Type, Inputs* and *Tool Support*. These dimensions can be used to identify candidate techniques with given characteristics. Afterward, the dimensions *Context Information, Domain, Family of Algorithm* and *Comment* can be inspected separately to further narrow down the list of candidate techniques. In the first step of identifying candidate techniques, a user can filter the complete list by the type of prediction. The second column *Detailed Prediction Type* contains information on the concrete type of prediction a technique is performing. Second, techniques can be filtered by their *Input* (columns three to five). Usually, the techniques take an event log including timestamps as input. Some techniques however require additional inputs, like a process model or a labeling function. If these inputs are not available or should not be integrated into the prediction, the respective techniques can be excluded. Third, candidate techniques can be identified by the type of *Tool Support* they offer, which is described in the sixth column of the PPTF. Some techniques offer code for implementation, or plugins for software like ProM, YAWL or Camunda. On the other hand, some techniques do not offer any tool support. Techniques offering tool support require less implementation effort, as at least parts of the implementation are already available. After these three steps for identifying candidate techniques, the PPTF offers further dimensions for assessing each technique individually. *Context Information* is included in columns seven to eleven of the framework. These columns reflect the dimensions of the Taxonomy of Business Process Context Information (i.e. *Time, Structure, Origin, Process Relation* and *Runtime Behaviour*) and contain the characteristics of the context information that a technique supports. If a certain context information needs to be included into the prediction, the techniques that do not support it can be excluded. All of the techniques in the framework are usually evaluated by using an event-log from a real-life process. In column twelve of the framework, the *Domain* from which the event-log stems is referenced. If a technique was validated in the same domain as the process that should be predicted, for example automotive, it could be an indicator that the technique is suitable for that kind of domain. All techniques in the framework are based on a specific type of algorithm, which is contained in the dimension *Family of Algorithm*. Examples are neural networks [e.g. 15], clustering [e.g. 26] and regression [e.g. 12]. This is included in the framework because the relative performance of algorithm families varies depending on the specific PBPM project. Table 3 views several exemplary entries of the extended PPTF with its dimensions. The entire PPTF is available as a digital appendix in a GitLab repository[1], since including as well as viewing all entries as part of this paper is not feasible.

Demonstration

In the following, we demonstrate the PPTF at the example of one of the techniques that is listed in Table 3 in more detail. Specifically, we explain why the technique shows the respective values of the framework dimension. As exemplary technique, we select [21], which is the sixth line in Table 3 and highlighted for readers convenience. The authors state that the goal of their technique is to efficiently produce a prediction model for any case-level prediction tasks. The authors specifically name next activity prediction and remaining time prediction as examples. Therefore, the technique can be found twice in the framework. First, with Prediction Type *time* and Detailed Prediction Type *remaining time* (as displayed in the framework below) and second with Prediction Type *categorical outcome* and Detailed Prediction Type *Next Activity* (not contained in the exemplary entries). The technique takes an event log as input. The event log should also contain timestamps and event attribute data. Therefore, the Input dimension is set to *Event log with timestamps with context information* (shortened to *event log* for readability purposes in the framework below). The authors of the technique made the source code of the prediction engine publicly available on GitHub. It does not represent a plugin for an existing tool. Therefore, the Tool dimension is set to *Y (impl.)*. Regarding the context information, the technique uses one-hot encoding to encode event attributes. As event attributes are used, the process relation dimension of the context is *event*. An event attribute can only be known once the event took place, not before. Therefore, the Time dimension is set to *runtime*. The fact that context information is processed on the event-level also tells us that the Runtime Behavior of the context information processed can either be *static* or *dynamic*. This is because the value of each can change from event to event and does not have to stay static over the whole instance. As described above, the technique takes an event log as input. No further inputs are needed. This means that all the context information that can be processed need to be contained in the event log. Following this, the Origin of the data has to be either *immediate*, meaning it is the information needed to carry out the process, or *internal* which is information directly related to the process. External or environmental context information would stem from outside the process and are therefore usually not contained in an event log. One-hot encoding maps key:attribute value pairs into vectors. The fact that the context information has a given structure, key:attribute pairs, but not a limited set of attributes of values leads to the Structure dimension being *semi-structured* for this technique. To test the technique, the authors used data sets of BPI challenges from the years 2012, 2013, 2014 and 2018. These data sets represent processes from the financial, automotive, customer support and public administration domains. Therefore, these values are written into the domain dimension. The technique exploits event attributes into recurrent neural network (RNN) prediction models by clustering events by their attribute values and using the cluster labels in the RNN input vectors. Therefore, the families of algorithm used in this technique are clustering and neural networks which concludes the dimensions of the PPTF.

Table 3. Exemplary Entries of the Process Prediction Technique Framework

Prediction Type		Input			Tool	Context					Domain	Family of Algorithm
Overall	Detailed	#1	#2	#3		Time	Structure	Origin	Process Relation	Runtime Behavior		
time	timestamp of next activity	event log	-	-	ProM Plugin	-	-	-	-	-	logistics	stochastic petri net
time	timestamp of next activity	event log	-	-	N	runtime	semi-structured	immediate	activity	static, dynamic	finance, customer support	probabilistic model
time	maintenance time	event log	-	-	N	-	-	-	-	-	automotive	time series, probabilistic model
time	remaining time	event log	-	-	ProM Plugin	a priori, runtime	semi-structured	immediate, internal	activity	static	public administration	regression
time	remaining time	event log	-	-	Y	runtime	structured	immediate	activity	static	healthcare	stochastic petri net
time	remaining time	event log	-	-	Y (impl.)	runtime	semi-structured	immediate, internal	event	static, dynamic	financial, automotive, customer support, public administration	neural network, clustering
time	average sojourn time of activities	event log			Y	-	-	-	event	-	healthcare, financial, customer support	neural network
...	...	...	...	...	...	...	...	...	...	...	...	...
categorical	next activity	event log	-	-	Y (impl.)	runtime	structured, semi-structured	immediate	event, instance	static, dynamic	financial, automotive	neural network
categorical	next activity	event log	labeling function	process model	ProM Plugin	a priori, runtime, future	structured, semi-structured	immediate, internal, external, environment	event, artefact	static	no validation	classification
categorical	violation of process constraints	event log	context data	-	N	runtime	semi-structured	immediate, internal, external, environment	event, instance	static, dynamic	automotive, logistics	clustering, time-series
...	...	...	...	...	...	...	...	...	...	...	...	...
sequence of outcomes	sequence of future activities	event log	-	-	N	a priori, runtime	semi-structured	immediate, internal	event	static, dynamic	financial	probabilistic model, regression
risk	risk	event log	labelling function		Camunda Plugin	-	-	-	-	-	financial	similarity weighted process instance graph, statistical analysis
...	...	...	...	...	...	...	...	...	...	...	...	...
inter-case metrics	inter-case metrics	event log	-	-	N	runtime	structured	immediate	event	static, dynamic	no validation	classification, regression, time series
...	...	...	...	...	...	...	...	...	...	...	...	...
cost	cost	event log	threshold(s)	-	N	-	-	-	-	-	transport, logistics	neural network
...	...	...	...	...	...	...	...	...	...	...	...	...

5   Conclusion

This paper addressed the RG to update the PPTF, which was originally developed by [10] in 2018, and extend its dimensions with context information. We reached this goal through a new literature review that builds upon the already existing review results of the initial authors. We integrated the original results with our own and extended the PPTF by context information dimensions, as proposed by [19]. Section 4 describes the updated PPTF and its dimensions. Table 3 shows some exemplary entries of the entire PPTF, which is available as a digital appendix[2], due to its large size.

We believe that this updated PPTF will support researchers and practitioners, who intend to use or develop business process prediction and monitoring techniques. Since the selection of an appropriate prediction technique is not only dependent on the given dimensions of the PPTF but is also strongly influenced by other project-dependent factors, we plan to address this limitation in future work. For example, the PPTF could be leveraged in the development of a PBPM implementation reference process that supports practicioners and researchers in the introduction and implementation of PBPM (e.g. [6]).

References

1. Poll, R., Polyvyanyy, A., Rosemann, M., Röglinger, M., Rupprecht, L.: Process forecasting: Towards proactive business process management. In: Proceedings of the 16th International Conference on Business Process Management. pp. 496–512. Springer (2018)
2. Breuker, D., Matzner, M., Delfmann, P., Becker, J.: Comprehensible predictive models for business processes. MIS Quarterly 40(4), 1009–1034 (2016)
3. Maggi, F.M., Di Francescomarino, C., Dumas, M., Ghidini, C.: Predictive monitoring of business processes. In: Proceedings of the 26th International Conference on Advanced Information Systems Engineering. pp. 457–472. Springer (2014)
4. Schwegmann, B., Matzner, M., Janiesch, C.: A method and tool for predictive event-driven process analytics. In: Proceedings of the 11th International Conference on Wirtschaftsinformatik. pp. 721–735. AISeL (2013)
5. Márquez-Chamorro, A.E., Resinas, M., Ruiz-Cortés, A.: Predictive monitoring of business processes: a survey. IEEE Transactions on Services Computing 11(6), 962–977 (2017)
6. Becker, J., Brunk, J., Ding, W., Niemann, M.: Conceptualization of an integrated procedure model for business process monitoring and prediction. In: 22nd IEEE Conference on Business Informatics, CBI 2020, Antwerp, Belgium. IEEE (2020)
7. Tama, B.A., Comuzzi, M.: An empirical comparison of classification techniques for next event prediction using business process event logs. Expert Systems with Applications 129, 233–245 (2019)
8. Teinemaa, I., Dumas, M., Rosa, M.L., Maggi, F.M.: Outcome-oriented predictive process monitoring: Review and benchmark. ACM Transactions on Knowledge Discovery from Data 13(2), 1–57 (2019)

---

[2] https://wiwi-gitlab.uni-muenster.de/j_brun17/process-prediction-technique-framework/

9. Verenich, I., Dumas, M., Rosa, M.L., Maggi, F.M., Teinemaa, I.: Survey and cross-benchmark comparison of remaining time prediction methods in business process monitoring. ACM Transactions on Intelligent Systems and Technology 10(4), 1–34 (2019)

10. Di Francescomarino, C., Ghidini, C., Maggi, F.M., Milani, F.: Predictive process monitoring methods: Which one suits me best? In: Business Process Management. pp. 462–479. Springer International Publishing (2018)

11. Rosemann, M., Recker, J.C.: Context-aware process design: Exploring the extrinsic drivers for process flexibility. In: Proceedings of the 18th International Conference on Advanced Information Systems Engineering. pp. 149–158. Namur University Press (2006)

12. Yeshchenko, A., Durier, F., Revoredo, K., Mendling, J., Santoro, F.: Context-aware predictive process monitoring: The impact of news sentiment. In: Proceedings of the 19th OTM Confederated International Conferences "On the Move to Meaningful Internet Systems". pp. 586–603. Springer (2018)

13. Weinzierl, S., Revoredo, K.C., Matzner, M.: Predictive business process monitoring with context information from documents. In: Proceedings of the 27th European Conference on Information Systems. pp. 1–10. AISeL (2019)

14. Schönig, S., Jasinski, R., Ackermann, L., Jablonski, S.: Deep learning process prediction with discrete and continuous data features. In: Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software Engineering. pp. 314–319. Science and Technology Publications (2018)

15. Evermann, J., Rehse, J.R., Fettke, P.: Predicting process behaviour using deep learning. Decision Support Systems 100, 129–140 (2017)

16. Weinzierl, S., Stierle, M., Zilker, S., Matzner, M.: A next click recommender system for web-based service analytics with context-aware LSTMs. In: Proceedings of the 53rd Hawaii International Conference on System Sciences. pp. 1542–1551. IEEE (2020)

17. Camargo, M., Dumas, M., González-Rojas, O.: Learning accurate lstm models of business processes. In: Proceedings of the 17th International Conference on Business Process Management (BPM). pp. 286–302. Springer (2019)

18. Brunk, J., Stottmeister, J., Weinzierl, S., Matzner, M., Becker, J.: Exploring the effect of context information on deep learning business process predictions. Journal of Decision Systems pp. 1–16 (2020)

19. Brunk, J.: Structuring business process context information for process monitoring and prediction. In: Proceedings of the Conference on Business Informatics 2020. Antwerp, Belgium (2020)

20. Kitchenham, B.: Procedures for performing systematic reviews. Keele, UK, Keele University 33(2004), 1–26 (2004)

21. Hinkka, M., Lehto, T., Heljanko, K.: Exploiting event log event attributes in rnn based prediction. In: Data-Driven Process Discovery and Analysis, pp. 67–85. Springer (2018)

22. Wahid, N.A., Adi, T.N., Bae, H., Choi, Y.: Predictive business process monitoring–remaining time prediction using deep neural network with entity embedding. Procedia Computer Science 161, 1080–1088 (2019)

23. Mehdiyev, N., Evermann, J., Fettke, P.: A novel business process prediction model using a deep learning method. Business & information systems engineering 62(2), 143–157 (2020)
24. Senderovich, A., Di Francescomarino, C., Maggi, F.M.: From knowledge-driven to data-driven inter-case feature encoding in predictive process monitoring. Information Systems 84, 255–264 (2019)
25. Ceci, M., Spagnoletta, M., Lanotte, P.F., Malerba, D.: Distributed learning of process models for next activity prediction. In: Proceedings of the 22nd International Database Engineering & Applications Symposium. pp. 278–282 (2018)
26. Bernard, G., Andritsos, P.: Accurate and transparent path prediction using process mining. In: European Conference on Advances in Databases and Information Systems. pp. 235–250. Springer (2019)