

Business Process Representation Learning

Peter Pfeiffer^{1,2}

¹German Research Center for Artificial Intelligence (DFKI), Campus D3_2, 66123 Saarbrücken, Germany

²Saarland Informatics Campus, 66123 Saarbrücken, Germany

Abstract

Data stored in information systems gathered from the execution of business processes is a rich source of information. Process mining aims to extract and gain knowledge from such data, usually captured in event logs, in order to understand and improve the business processes. From a data-science perspective, event log data is a very interesting yet complex data modality. It does not only describe the process from the control-flow perspective, but also contains additional information like entities and organizations involved, temporal aspects and much more. While there is a lot of work on applying existing machine learning techniques on event log data for solving a specific problem, little work has focused on how to learn from such data effectively. This work presents the idea of developing representation learning models for event logs, i.e. neural-network-based methods specifically designed for this data modality, which learn generic and rich representations of events and cases. The representations are expected to be used for solving different business problems such as process prediction, anomaly detection or other process mining tasks more efficiently and effectively.

Keywords

Business Process Data, Representation Learning, Process Analytics

1. Motivation and Problem Description

Extracting and gaining knowledge from data of business process executions, e.g., provided by information systems has gained a lot of attention over the past years. In the research field of process mining, a large variety of methods to analyse such data have been developed where the standard data format to store such information and make analysis on are event logs [1]. While the majority of process mining methods extract hand-crafted features from the information in the event log, like footprint matrices to describe how activities are in relation to each other, some use machine learning (ML) methods based on neural networks to make predictions about the future state of running process instances [2]. On other data modalities like images or text, it has been shown that neural-network-based methods perform equally well as, or outperform, traditional feature-based methods in image [3] or language understanding tasks [4]. For event log data, this has mainly been shown for process prediction [2, 5, 6]. Recently, a study showed that process discovery can be solved with graph neural networks [7] reaching comparable performance to feature-based methods. However, many problems in process mining still rely on hand-crafted instead of learned features.

BPM 2022 Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Track

✉ peter.pfeiffer@dfki.de (P. Pfeiffer)

🆔 0000-0002-0224-4450 (P. Pfeiffer)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

Part of the success of neural-network-based methods on other data modalities like images and texts is due to their ability to learn and generate a rich representation of the concepts in the data in form of a feature vector which allows to be used for a variety of tasks. For instance, neural-network-based language models are trained to fill gaps in sentences and to predict whether two sentences follow each other which teaches them in learning effective representation of words and sentences [4] without relying on labels created by domain experts. Afterwards, they can be fine-tuned on a variety of specific tasks using labeled data. Such approaches belong to representation learning, a field of machine learning that deals with learning a representation of data that "makes it easier to extract useful information when building classifiers or other predictors" [8]. After pre-training using such a self-supervised training method, the same model can be used to solve various downstream tasks utilizing the learned representation. If the pre-trained model produces "good" representations, it is sufficient to add task-specific, simple neural networks to solve certain problems. The idea behind such pre-training is that the features learned thereby are useful for the downstream tasks. Thus, representation learning aims at incorporating features in the representation which are useful for solving different settings and tasks, enabling transfer learning, domain adaption and multitask learning [9].

This two-stage strategy, which has to lead to great success in machine learning, makes solving many tasks more effective which is indicated by a high accuracy on various downstream tasks. One example is BERT [4], a transformer-based model that was pre-trained on a very large corpus of textual data in a self-supervised fashion to generate representations of words and sentences. The learned representations have been used for various different downstream tasks and set new state-of-the-art results in question answering, sentence classification and other tasks that require reasoning over words and sentences. For other data modalities, approaches that adapt these ideas [10, 11] perform equally well or better than existing methods.

For data describing business processes, there is no comparable approach that can produce representations effective for solving different tasks. As the modality of event data is different to text or images, pre-trained models like BERT cannot directly be applied. While textual data is often described as a single sequence of words, a case consists of events with attributes that describe what action has been performed at what time (which is not necessarily a sequence). Additional attributes can be added, e.g., who performed the activity or what objects have been involved. Events can have different numbers of attributes which can be of different types, e.g., categorical, numerical and temporal and have different scales. Furthermore, some attributes change within each event while others are fixed for the whole case. These characteristics make it challenging to learn representations from event log data with neural networks. Furthermore, we argue that it is difficult to process them with existing network architectures like LSTMs or BERT as they do not fully adapt to and account for the characteristics of that data.

In order to enable ML-based analysis on event log data on a large scale, more research is required how to learn rich representations of concepts describing business processes. For instance, to make such networks aware of what event logs consist of by learning the concepts of events, cases and processes. Existing neural-network-based approaches for event logs either focus on solving one task, like next step prediction [2] or anomaly detection [12], or learn representations using a subset of the information available in the event log. For instance, representations learning approaches of cases [13, 14] or events [15] include categorical attributes, but no numerical or temporal information. This makes such approaches less generic, as they lack

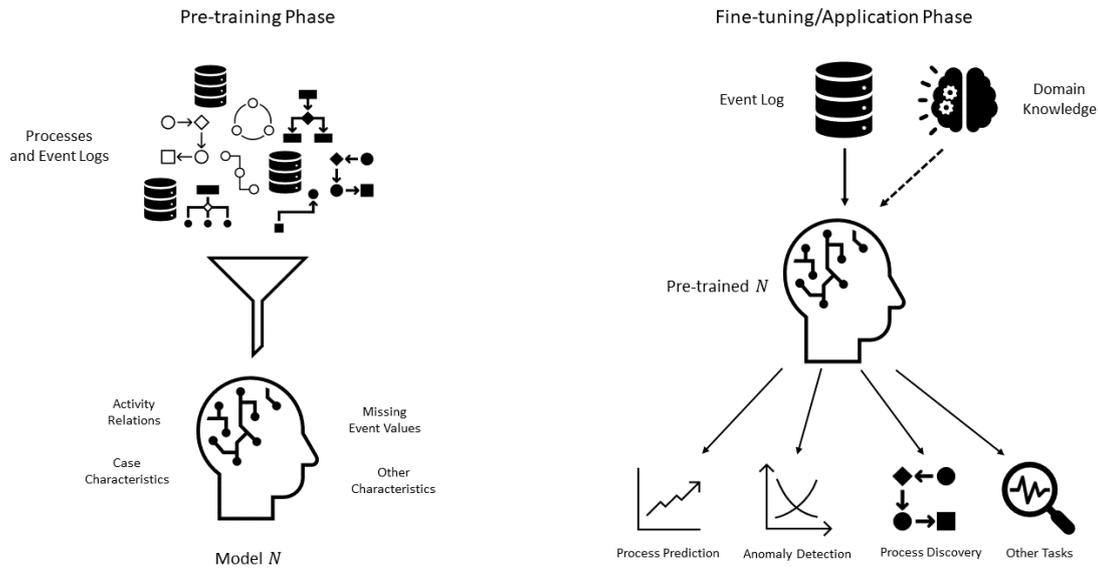


Figure 1: Overview of the intended solution. Left: Pre-training on large sets of business processes and event logs. Right: Application of the pre-trained model on one specific event log solving different tasks.

to include all relevant information found in the event log and only work for certain tasks. Thus, the learned representation of such approaches are task-specific. Finding ways to learn a generic representation is one of the main objectives of representation learning which could, applied to event logs, allow to use such methods for more tasks. Some work [14, 16, 17] indicates that approaches being able to learn rich representations of cases or events containing and combining information from multiple event or case perspectives could be helpful to solve certain tasks on such data. However, this is hard to archive using hand-crafted features.

This PhD project is about developing methods that can learn rich representations of different concepts in data gathered from the execution of business processes to be used for solving analytical tasks more effectively. Thus, the research problem is to develop specialized neural network architecture and training objective which can be used for learning from such data as well as applications and assessments that demonstrate and measure the effectiveness of the representations. For instance, how to adapt and adjust successful training procedures from other modalities to work on event log data as well as encoding approaches which preserve the structure and semantic of the data. Furthermore, we want to investigate what characteristics the representations should have to be effective.

In general, a good representation is one that makes solving subsequent tasks easier [9]. Some general-purpose priors exist which are not task specific and can be followed when developing representation learning methods [8]. However, as the representations in this work are expected to be effective for solving process mining tasks, they should contain characteristics that support problem solving on event data. The following characteristics reflect desirable characteristics from our current point of view but a more systematic approach of collecting and aligning them

with process mining tasks is planned as part of the project. First, representations should be effective for solving different tasks and trainable with small quantities of manually labeled data. Furthermore, the approach should be able to learn representations for events as well as for cases which requires a hierarchical perception of the data. Event representations are expected to contain feature describing the semantic of the activity and all its attributes, its position in the case as well as its context, i.e., the events nearby. Similarly, representations of cases should aggregate the semantics of all event as well as behavioural characteristics of the case itself. Another desired property is that it allows domain adaption, i.e., that representations can also be created for event logs where the model was not trained on. However, while textual data and the features describing it do not change too much when considering one language, different processes might have very different behaviour that makes it challenging to transfer certain features learned from one set of event logs to another. Thus, we want to investigate which features are transferable or how to transfer them with little effort.

Certain process mining task to solve ultimately might require an additional training phase as the generic characteristics learned are not sufficient or the features required are specific to the event log. Thus, domain knowledge (labeled data) or a different training objective is needed. For instance, it could be helpful for some tasks to have the case representation contain information which cases fit the underlying process and which not, i.e., for case classification tasks. Such features are difficult to learn generically and might require an additional training phase or labeled data on the event log of interest.

What makes representation learning different to other machine learning tasks is that no objective or target function exists which can be directly optimized during training to obtain the desired characteristics. Rather, training methods have to be developed that "shape" the representations in the desired form. Nevertheless, pre-training to learn certain generic characteristics has shown to be beneficial on other modalities [9] which makes investigating this idea for event data interesting.

2. Methodology and Techniques

The research project follows the design science methodology [18] including exploratory phases within its cycles that steer the development of a representation learning approach for business process data. The problem and opportunity identification, i.e., to design a representation learning approach for event log data that makes event log analysis more efficient, is part of the relevance cycle. Furthermore, acceptance criteria must be defined. For investigating which characteristics the representations should have to be effective, we follow the general priors for good representations [8] and systematically collect tasks and problems the approach will be applied on. From these observations we will to derive features that are necessary to be contained in the representation for process mining tasks. As discussed earlier, it is very hard to directly measure whether the representations contain the desired characteristics. Instead, representation learning approaches are usually evaluated on a variety of different tasks. If they are able to solve the tasks accurately, the representations are supposed to be effective. We follow the same evaluation procedure by assessing whether the learned event and case representations can be used to solve different process mining tasks, like process prediction or

anomaly detection with similar or better performance to existing algorithms measured in terms of established metrics like precision, recall and accuracy. Furthermore, we try to assess how the learned representations can be utilized for tasks not considered so far.

State-of-the-art machine learning techniques will be used, i.e., neural network architectures and training methods for learning representations as well as knowledge from the process mining field. Experience from both domains is combined to develop new methods to process and learn from event log data. Thereby, additions to the knowledge base in both domains are made, i.e. in learning (hierarchical) representations of complex data modalities as well as how to learn and use such representations to solve process mining tasks.

In the design cycle, new representation learning approaches are being developed in an iterative and exploratory fashion. Knowledge from the corresponding domains is used to enhance the approach and to test its effectiveness on various tasks. Each representation learning method consists of a neural network architecture and a self-supervised pre-training phase. Both need to be combined appropriately to create valuable feature vectors. Techniques applied throughout the design are for instance network architectures like transformer models [19], specifically, customized transformer architectures for modalities like time series [10] which are further extended to appropriately capture the concepts in event logs. For training, self-supervised learning techniques similar to BERT are adopted, e.g., reconstructing missing parts in the input or predicting characteristics of higher level concepts in the data. They will be combined with process analytic knowledge to find effective learning objectives that fit business process data.

3. Solutions and Results

Results archived so far in this PhD project include the Multi-Perspective Process Network (MPPN) [20] which learns representations of cases. With respect to the network architecture and training method, an image encoding approach for time-series was applied that allows to process categorical, numerical and temporal information in the event log in the same way using a self-supervised pre-training phase and an architecture based on convolutional neural networks. Instead of training embeddings for different attributes in the event log each time, we transform all perspectives into distinct image representations and use pre-trained convolutional neural networks to extract features describing the perspective. We first pre-train MPPN on the next event prediction task by predicting several attributes values of the next event at once. Thereby, it learns the general characteristics of the event log, i.e., how activities and attributes influence each other, before being applied on a certain task.

It has been demonstrated [20] that the learned representations are effective for solving different process prediction tasks and, without additional training, suitable for case retrieval, i.e., retrieving contextual similar cases to one case of interest. Nevertheless, MPPN has some shortcomings. The main disadvantages are that a lot of semantic information is lost when transforming perspectives to 2D images and that it does not learn representations for events. Loosing the semantic information, e.g., activity and attribute names and values makes the approach less generic as it requires to train one MPPN per event log which prevents domain adaption.

In the next iteration, a new approach following the two-stage scheme illustrated in Figure 1

is being developed that overcomes the issues of MPPN. Again the model will be pre-trained on a self-supervised task to generate representations which can afterwards be utilized to solve different problems. The new approach learns representations of events and cases simultaneously, compromising as much semantic information found in the event log as possible using a new architecture that is more efficient and flexible.

In the pre-training phase, the objective is to train the model N to recognize and interpret the different concepts in the data and how they relate to each other by predicting different characteristics of business processes on event- and case-level to learn important features. This involves, e.g., to reconstruct missing event values in order to learn features on event-level and predicting case or process characteristics to learn higher-level features. By including and encoding as much semantic information from the event log as possible, such as attribute and activity names, the representations should be rich in information and the approach generic. This is achieved by splitting each event into distinct tokens where each token carries a certain part of the information [19, 10] which enables a more flexible and generic way of processing and learning from event data. For instance, each activity in each event is encoded as a token by contextualising the activities name and combining that with the contextualized attribute name. The same is done for other for attributes that contain semantic information. Depending on the attribute type, appropriate encoding techniques are used to transform the information into tokens which serve as input to the neural network. Similar to the positional encoding used in BERT [19], a "event encoding" is added that indicates which tokens belong to the same event. Thus, each token contains information what attribute it represents, which value/content the attribute has as well as to which event it belongs. Using this token-based encoding approach, which embeds information how event log data is structured into the input given to N allows that data from the event log can be processed generically and that N can interpret the information as intended by the data structure. Additionally, special tokens like the *CLS*-Token in BERT are added which can be used to solve classification tasks on cases or events.

Splitting and encoding the data into tokens allows to process them with transformer-based architectures [19, 10]. After pre-training N on a large set of synthetic and real-world business processes and event log it will be fine-tuned or directly applied on different tasks to measure its effectiveness. For some tasks, no fine-tuning using labeled data for the target task may be required. For instance unsupervised ones such as anomaly detection [12] or clustering-based tasks like behaviour mining [17] or event abstraction [21, 16]. For other tasks, the representations need to be fine-tuned, e.g., for process prediction as demonstrated in previous work [20]. For other tasks, additional labeled data might be required indicated by the dotted arrow in Figure 1.

In order to demonstrate that the representations are helpful, the performance will be compared to existing techniques using common datasets and standardised evaluation procedures. For example next-step and outcome prediction tasks on the BPIC event logs or classifying traces into fitting and unfitting ones using the data provided by the process discovery challenge (PDC)¹. A dedicated representation learning benchmark is also planned which combines different process mining tasks and datasets to test and compare approaches on.

¹<https://www.tf-pm.org/competitions-awards/discovery-contest>

4. Conclusion

Designing customized neural-network-based architectures and training methods for event logs that are generic and learn to interpret the modality, i.e., learn representations of events and cases including their semantics is novel. The idea is different from existing approaches that apply network architectures developed for other data modalities like LSTM and training strategies like next step prediction on event log data in the sense that we aim for approaches that learn the concepts in the data and can be used for various tasks. By separating the training from the application phase, representations can be learned and applied for solving different tasks. This is supposed to be more effective as designing different approaches for different problems. Furthermore, having a rich representation of events and cases makes problem solving easier as a simple classifier with a few samples can be sufficient for solving a specific task.

Using a feature vector representation also brings some limitations as the feature vectors produced by N are not as explainable and understandable by humans as hand-crafted features. Furthermore, pre-training a representation learning model requires a lot of data and careful parameter optimization. Not all tasks in all domains benefit from using one representation which might also apply to this project. However, once the model is pre-trained, it can be used on different datasets and easily fine-tuned to various tasks.

The results archived so far indicate that representations can work for different predictive tasks as well as for retrieval. In the future, expect to get insights how the architecture of the neural network, encoding methods and training objective have to be designed for learning effective representations of event logs and which features are useful for solving process mining tasks. As the data modality of event logs is challenging, learning representations makes it a interesting problem not only from a process mining but also from the machine learning perspective. This could enable to solve other process mining tasks, that yet rely on hand-crafted feature more effectively, enabling ML-based analysis in process mining on a larger scale.

References

- [1] W. M. P. van der Aalst, *Process Mining: A 360 Degree Overview*, Springer International Publishing, Cham, 2022, pp. 3–34. doi:10.1007/978-3-031-08848-3_1.
- [2] J. Evermann, J.-R. Rehse, P. Fettke, Predicting process behaviour using deep learning, *Decision Support Systems* 100 (2017) 129–140.
- [3] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems* 25 (2012).
- [4] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: *NAACL-HLT*, 2019.
- [5] N. Tax, I. Teinmaa, S. J. van Zelst, An interdisciplinary comparison of sequence modeling methods for next-element prediction, *Software and Systems Modeling* 19 (2020) 1345–1365.
- [6] W. Kratsch, J. Manderscheid, M. Röglinger, J. Seyfried, Machine learning in business process monitoring: A comparison of deep learning and classical approaches used for outcome prediction, *Business & Information Systems Engineering* 63 (2021) 261–276. doi:10.1007/s12599-020-00645-0.

- [7] D. Sommers, V. Menkovski, D. Fahland, Process discovery using graph neural networks, in: 3rd International Conference on Process Mining (ICPM), IEEE, 2021, pp. 40–47.
- [8] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, *IEEE transactions on pattern analysis and machine intelligence* (2013) 1798–1828.
- [9] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016. URL: <http://www.deeplearningbook.org>.
- [10] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, C. Eickhoff, A transformer-based framework for multivariate time series representation learning, in: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, Association for Computing Machinery, 2021, p. 2114–2124. doi:10.1145/3447548.3467401.
- [11] A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman, J. Carreira, Perceiver: General perception with iterative attention, in: *International Conference on Machine Learning*, PMLR, 2021, pp. 4651–4664.
- [12] T. Nolle, A. Seeliger, M. Mühlhäuser, Binet: Multivariate business process anomaly detection using deep learning, in: *Business Process Management*, Springer International Publishing, 2018, pp. 271–287.
- [13] S. Luettgen, A. Seeliger, T. Nolle, M. Mühlhäuser, Case2vec: Advances in representation learning for business processes, *Process Mining Workshops, ICPM 2020*, Springer International Publishing, 2020, pp. 162–174.
- [14] A. Seeliger, S. Luettgen, T. Nolle, M. Mühlhäuser, Learning of process representations using recurrent neural networks, in: *International Conference on Advanced Information Systems Engineering*, Springer International Publishing, 2021, pp. 109–124.
- [15] P. De Koninck, S. vanden Broucke, J. De Weerd, act2vec, trace2vec, log2vec, and model2vec: Representation learning for business processes, in: *Business Process Management*, Springer International Publishing, 2018, pp. 305–321.
- [16] A. Rebmann, Abstracting low-level event data for meaningful process analysis, in: *Proceedings of the Demonstration & Resources Track, Best BPM Dissertation Award, and Doctoral Consortium at BPM 2021 co-located with the 19th International Conference on Business Process Management*, 2021.
- [17] L. Abb, C. Bormann, H. van der Aa, J. R. Rehse, Trace clustering for user behavior mining, in: *30th European Conference on Information Systems (ECIS 2022)*, 2022.
- [18] A. R. Hevner, A three cycle view of design science research, *Scandinavian journal of information systems* 19 (2007) 4.
- [19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: *Advances in neural information processing systems*, 2017, pp. 5998–6008.
- [20] P. Pfeiffer, J. Lahann, P. Fettke, Multivariate business process representation learning utilizing gramian angular fields and convolutional neural networks, *Business Process Management*, Springer International Publishing, 2021, pp. 327–344. doi:10.1007/978-3-030-85469-0_21.
- [21] S. J. van Zelst, F. Mannhardt, M. de Leoni, A. Koschmider, Event abstraction in process mining: literature review and taxonomy, *Granular Computing* 6 (2021) 719–736. doi:10.1007/s41066-020-00226-2.