

# FEEED: Feature Extraction from Event Data

Andrea Maldonado<sup>1,2,\*</sup>, Gabriel Marques Tavares<sup>3</sup>, Rafael Oyamada<sup>3</sup>, Paolo Ceravolo<sup>3</sup>  
and Thomas Seidl<sup>1,2</sup>

<sup>1</sup>Ludwig Maximilians Universität München, Munich, Germany

<sup>2</sup>Munich Center for Machine Learning, Munich, Germany

<sup>3</sup>Università degli Studi di Milano, Milan Italy

## Abstract

The analysis of event data is largely influenced by the effective characterization of descriptors. These descriptors serve as the building blocks of our understanding, encapsulating the behavior described within the event data. In light of these considerations, we introduce FEEED (Feature Extraction from Event Data), an extendable tool for event data feature extraction. FEEED represents a significant advancement in event data behavior analysis, offering a range of features to empower analysts and data scientists in their pursuit of insightful, actionable, and understandable event data analysis. What sets FEEED apart is its unique capacity to act as a bridge between the worlds of data mining and process mining. In doing so, it promises to enhance the accuracy, comprehensiveness, and utility of characterizing event data for a diverse range of applications.

## Keywords

Featurization, Event log behavior, Event data, Feature extraction

## 1. Introduction

The analysis of event data behavior holds a paramount role in a wide array of domains, spanning from critical sectors like healthcare, finance, to the ever-vigilant realm of cybersecurity. It enables crucial tasks such as anomaly detection, pattern recognition, and informed decision-making. However, the quality and effectiveness of these analyses depend significantly on the ability to extract meaningful descriptive features from event data. Yet, existing literature predominantly relies on simplistic descriptors such as the number of activities, variants and traces. But these descriptors fall short in capturing the intricate sequential and concurrent dynamics inherent in event data. Fig4PM[1] proposes a collection of event log measures extracted from existing literature, combining control-flow and statistical metrics. While they provide a fundamental set of features, they do not offer a comprehensive representation of all aspects necessary to fully characterize an event log. Often when approaching process mining from a data mining perspective, a transformation step is often necessary to map event log behavior into a numerical feature space. Unfortunately, this transformation frequently yields non-interpretable features [2].

---

✉ maldonado@dbs.ifi.lmu.de (A. Maldonado); gabriel.tavares@unimi.it (G. M. Tavares); rafael.oyamada@unimi.it (R. Oyamada); paolo.ceravolo@unimi.it (P. Ceravolo); seidl@dbs.ifi.lmu.de (T. Seidl)


🌐 <http://github.com/andreamalhera> (A. Maldonado); <http://github.com/gbrltv> (G. M. Tavares);

<https://github.com/raseidi> (R. Oyamada); <https://ceravolo.di.unimi.it> (P. Ceravolo);

<https://www.dbs.ifi.lmu.de/cms/personen/professoren/seidl/index.html> (T. Seidl)



© 2023 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)



preprocessing capabilities, encompassing featurization, streamline the transition from raw event data to meaningful insights. By integrating these two domains, FEEED empowers analysts to harness the full potential of their data while enhancing the accuracy and utility of downstream analysis. E.g. by correlating event data behavior, in form of features, to algorithm performance, we can exploit data characteristics to gain knowledge of process mining and data mining tasks. This synergy between data mining and process mining holds immense promise for advancing the state-of-the-art in event data analysis.

FEEED's key innovation lies in its compilation of features gathered from the literature. While state-of-the-art feature extraction is performed singularly before a specific task, our library centralizes feature extraction. Thus, data scientists can develop, compare and assess process and data mining pipelines on event data using a common feature set. FEEED offers a set of features encapsulating various aspects of event data behaviors, from straight-forward, as e.g. number of events, to complex, as e.g. entropies, for activity, trace and event-log levels. This rich, comprehensible feature set enhances the depth and breadth of insights derived from the data, making FEEED a valuable asset in data-driven decision-making processes.

To correctly capture data behavior, we rely on a set of features proposed in the literature covering different aspects of event data [4, 5, 6] and considering different granularity levels (activity, trace, and log). The significance of the chosen features for FEEED was extensively demonstrated by analyzing the correlation between them and algorithm performance for multiple process mining tasks: E.g. Trace Clustering [4], Process Discovery [6, 3]. For trace-level descriptors, trace lengths and variants are used as the basis for statistical-based metrics. For trace lengths, we compute data distribution with profiles including kurtosis and skewness coefficients, mean, standard deviation, the 25th and 75th percentile of data, interquartile range, and geometric and harmonic mean. Trace variants analysis can enlighten the process flow behavior by extending statistical features to ratios, such as the ratio of the most common variant compared to all variants. The activity-based features are subdivided into three groups: activities, start activities, and end activities. We extract 12 statistical-based features for each group, similar to those used for trace profiling. On the log level, we extract four features: The number of events, traces, unique traces, and their ratio. We enhance statistical descriptors by adding complexity-based metrics, i.e., entropies [5]. The entropy measures are further divided into four groups: in-trace frequency, language-inspired, dynamic systems, and molecular structural analysis. These metrics capture log structure and variability across activity, trace, and event-log regardless of the logs complexity. Lastly, process complexity metrics[6] are based on graph entropy and capture complexity in multiple perspectives. The authors demonstrate how such measures successfully depict complexity correlated to a task, in this case, process discovery.

The library also offers configurable feature extraction, such as extracting features from a single type or selecting a specific set of features. It allows users to tailor the feature selection process to their specific needs. This adaptability ensures that FEEED can seamlessly integrate with diverse data sources and cater to a wide spectrum of analysis objectives. Finally, FEEED's extendibility is a cornerstone of its utility. It provides a platform that can be extended with additional features or custom encoding techniques, making it adaptable to evolving data analysis requirements. This feature ensures that FEEED remains a valuable library in the long term, capable of addressing new challenges and opportunities in event data behavior analysis.

### 3. Availability and Usage

Our library is publicly available on GitHub<sup>1</sup> and as a PyPI<sup>2</sup> package, including installation instructions as well as an interactive tutorial with real data sets. Additionally, we include a tutorial video<sup>3</sup>. FEEED currently supports eXtensible Event Streams (XES) [7] as input, which is a commonly used format for event data. Furthermore using any csv-to-xes converter e.g. the one from pm4py [8], csv files can also be analysed using FEEED. Our library is easily extendable, as shown by our tutorial on “*Extending features*” with the example of “*time-based features*” on the aforementioned GitHub repository<sup>4</sup>. As an illustrative use case, we explore how FEEED can enhance the process of identifying similar logs within a log collection. This application holds particular relevance in numerous organizations where stakeholders frequently seek to group logs or discern related event logs. Our approach involved taking a collection of event logs and characterizing them using FEEED. Subsequently, we employed cosine similarity to calculate the degree of similarity between every pair of logs. The resulting visualization, shown in Figure 1, represents each log as a node, with edges connecting a log to its three most similar counterparts within the network. Interestingly, the results of this application show how processes originating from the same nature are closely connected, e.g., BPIC15 and BPIC17.

### 4. Conclusion

We introduced FEEED, a library developed for feature extraction from event data. Its aim is addressing the need for accurate and comprehensive event data analysis. By shifting from simplistic descriptors to advanced feature extraction, it enhances the performance of downstream tasks and supports effective decision-making across diverse domains. FEEED has been thoroughly implemented, tested, and is publicly available. Notably, opposed to deep learning encoding techniques, it provides human-interpretable features, facilitating deeper data insights for stakeholders and analysts. Additionally, it seamlessly integrates with data mining techniques, offering flexibility and adaptability for a variety of analytical tasks.

### References

- [1] F. Zandkarimi, J.-R. Rehse, Fig4pm: A library for calculating event log measures (extended abstract), 2021. URL: <https://api.semanticscholar.org/CorpusID:243858957>.
- [2] S. B. Jr., P. Ceravolo, R. S. Oyamada, G. M. Tavares, Trace encoding in process mining: a survey and benchmarking, *Engineering Applications of Artificial Intelligence* (2023).
- [3] S. Barbon Junior, P. Ceravolo, E. Damiani, G. Marques Tavares, Evaluating trace encoding methods in process mining, in: J. Bowles, G. Broccia, M. Nanni (Eds.), *From Data to Models and Back*, Springer International Publishing, Cham, 2021, pp. 174–189.

---

<sup>1</sup><https://github.com/lmu-dbs/feeed>

<sup>2</sup><https://pypi.org/project/feeed/>

<sup>3</sup><https://youtu.be/wS6n3ngRRd8>

<sup>4</sup><https://github.com/lmu-dbs/feeed#extending-features>

- [4] G. M. Tavares, S. Barbon Junior, E. Damiani, P. Ceravolo, Selecting optimal trace clustering pipelines with meta-learning, in: J. C. Xavier-Junior, R. A. Rios (Eds.), *Intelligent Systems*, Springer International Publishing, Cham, 2022, pp. 150–164.
- [5] C. O. Back, S. Debois, T. Slaats, Entropy as a measure of log variability, *Journal on Data Semantics* 8 (2019). doi:10.1007/s13740-019-00105-3.
- [6] A. Augusto, J. Mendling, M. Vidgof, B. Wurm, The connection between process complexity of event sequences and models discovered by process mining, *Information Sciences* 598 (2022) 196–215. doi:https://doi.org/10.1016/j.ins.2022.03.072.
- [7] Ieee standard for extensible event stream (xes) for achieving interoperability in event logs and event streams, *IEEE Std 1849-2016* (2016) 1–50. doi:10.1109/IEEESTD.2016.7740858.
- [8] A. Berti, S. J. van Zelst, W. M. P. van der Aalst, *Process mining for python (pm4py): Bridging the gap between process- and data science*, *CoRR abs/1905.06169* (2019). URL: <http://arxiv.org/abs/1905.06169>. arXiv:1905.06169.