Sequential Modeling and Structural Anomaly Analytics in Industrial Production Environments

Martin Atzmueller[1] and Andreas Schmidt[1] and David Arnu[2]

[1] University of Kassel, Research Center for Information System Design
{atzmueller, schmidt}@cs.uni-kassel.de
[2] RapidMiner GmbH, darnu@rapidminer.com

Abstract. The analysis of sequential data is a prominent research topic, e. g., for investigating actions, events or log entries. This demonstration paper presents an integrated approach for anomaly analytics in an industrial production scenario. Based on first-order Markov chain models, we analyse sequential trails relative to specific hypotheses in an industrial application context. We summarize the applied method and present its implementation in a data analytics process.

1 Introduction

In many industrial areas, production facilities have reached a high level of automation. Here, knowledge about the respective processes is crucial, e. g., targeting the topological structure of a plant, sequences of operator notifications (alarms), and unexpected (critical) situations. Then, the analysis of (exceptional) sequential patterns is an important task for obtaining insights into the process and for modelling predictive applications.

Context. The BMBF funded research project "Early detection and decision support for critical situations in production environments"[3] (short FEE) aims at detecting critical situations in production environments as early as possible and to support the facility operator in handling these situations, e. g., [9]. In abnormal situations, typically such a large number of notifications is generated, that it often cannot be physically assessed by the operator [31]. Therefore, appropriate abstractions and analytics methods are necessary in order to adapt and to change from a reactive to a proactive behaviour. The consortium of the FEE project consists of several partners also including application partners from the chemical industry. These partners provide the use cases for the project and background knowledge about the production process which is important for designing suitable analytical methods.

Objectives. This paper presents sequential modelling and anomaly analytics in an industrial application context. We present the implementation of a comprehensive modelling approach for comparing hypotheses with observed "reference" sequential patterns, based on methods for modelling and comparing networks and transition matrices, in particular the HypGraphs [12] and DASHTrails approaches [11]. Then, we aim to identify deviating (abnormal/anomalous) and conforming (normal) hypotheses. Implemented as a new RapidMiner operator and embedded in an analytical process, we demonstrate the application (cf., Section 4) of the proposed approach.

[3] http://www.fee-projekt.de

2 Related Work

The investigation of sequential patterns and sequential trails are interesting and challenging tasks in data mining and network science, in particular in graph mining and social network analysis, e. g., [4,7]. A general view on modeling and mining of ubiquitous and social multi-relational data is given in [5] focusing on social interaction networks. Here, dynamics and evolution of contacts patterns [8,16,20], for example, and their underlying mechanisms, e. g., [23] are analyzed. However, the analysis in these contexts focuses on aggregated sequential data. Navigational patterns, as sequential (link) patterns in online systems, have been analysed and modelled, e. g., in [25, 29]. In contrast to that, our approach focuses on modelling and comparing sequential patterns (hypothesis) in a graph-based network representation.

For comparing hypotheses and sequential trails, the HypTrails [28] algorithm has been proposed. In [11] we have presented the DASHTrails approach that incorporates probability distributions for deriving transitions utilizing HypTrails. Based on that, the HypGraphs framework [12] provides a more general modeling approach. Using general weight-attributed network representations, we can infer transition matrices as *graph interpretations*, while HypGraphs consequently also relies on first-order Markov chain modeling [19,29] and Bayesian inference [29,30].

Sequential pattern analysis has also been performed in the context of alarm management systems, where sequences are represented by the order of alarm notifications. Folmer et al. [14] proposed an algorithm for discovering temporal alarm dependencies based on conditional probabilities in an adjustable time window. To reduce the number of alarms in alarm floods Abele et al. [2] performed root cause analysis with a Bayesian network approach and compared different methods for learning the network probabilities. Vogel-Heuser et al. [31] proposed a pattern-based algorithm for identifying causal dependencies in the alarm logs, which can be used to aggregate alarm information and therefore reduce the load of information for the operator. In contrast to those approaches, the proposed approach is not only about detecting sequential patterns. We provide a systematic approach for the analysis of (derived) sequential transition matrices and its comparison relative to a set of hypotheses. Thus, similar to evidence networks in the context of social networks, e. g., [22], we model transitions assuming a certain interpretation of the data towards a sequential representation. Then, we can identify important influence factors

Process Mining [1] aims at the discovery of business process related events in a sequence log. The assumption is that event logs contain fingerprints of a business process, which can be identified by sequence analysis. One task of process mining is conformance checking [24, 27] which has been introduced to check the matching of an existing business process model with the a segmentation of the log entries. Compared to these approaches, we do not use any apriori knowledge about business processes to create our hypothesis. Furthermore, our hypothesis do not necessarily need to conform with an existing business process.

There are different definitions of an anomaly. According to the classical definition of [15], "an outlier is an observation that differs so much from other observations as to arouse suspicion that it was generated by a different mechanism". Then, interesting, important or exceptional groups [3, 26] can be identified. In contrast to approaches for anomaly detection that only provide a classification of anomalous and normal events, we can assess different anomaly hypotheses: Applying the proposed approach, we can then generate an anomaly indicator – as a potential kind of second opinion method for assessing the state of a production plant that can help for indicating explanations and traces of unusual alarm sequences in the plant. Furthermore, using the network representation, we can analyze anomalous episodes relative to structural (plant topology) as well as dynamic (alarm sequence) episodes.

3 Method

Our application context is given by (abstracted) alarm sequences in industrial production plants in an Industry 4.0 context, cf., [31]. Specifically, we consider the analysis of the plant topology and anomaly detection in alarm logs. We formulate the "reference behaviour" collecting normal episodes as sequences of normal situations, which is typically observed for long running processes, and can also be simply validated by a domain expert or notes from the operator journals. Then, we compare episodes of alarm sequences (formulated as hypotheses) in order to detect deviations, i. e., abnormal episodes, and conforming ones corresponding to the "normal behaviour". We map these sequences to transitions between functional units of an industrial plant, applying the modeling approach described below. The results can both be used for anomaly analytics as well as for diagnostics, by inspecting the transitions in detail.

3.1 Overview

Following [11, 12], we model transition matrices given a probability distribution of certain states. We assume a discrete set of such states Ω corresponding to the nodes of a network (without loss of generality $\Omega = \{1, \ldots, n\}$, $n \in \mathbb{N}, |\Omega| = n$). For modelling, we consider a sequential interpretation (according to the first-order Markov property) of the original data with respect to the obtained transition probabilities (Markov chain).

As shown in Figure 1, we perform three steps, discussed below in more detail:

1. Modeling: Determine a transition model given the respective weighted network using a *transition modeling function* $\tau : \Omega \times \Omega \rightarrow \mathbb{R}$. Transitions between sequential states $i, j \in \Omega$ are captured by the elements m_{ij} of the transition matrix M, i. e., $m_{ij} = \tau(i, j)$. Then, we collect sequential transition matrices for the given network (data) and hypotheses.
2. Estimation: Apply HypTrails, cf., [28] on the given data transition matrix and the respective hypotheses, and return the resulting evidence.
3. Analysis: Present the results for semi-automatic introspection and analysis, e. g., by visualizing the network as a heatmap or characteristic sequence of nodes.

Thus, using τ, we can model (derived) transition matrices corresponding to the *observed data*, e. g., given frequencies of alarms on measurement points, as well as hypotheses on sequences of alarms. For data transition matrices, we need to map the transitions into derived counts in relation to the data; for hypotheses we provide (normalized) transition probabilities. As a simple transformation for normalization, we can, e. g., directly convert the weighted network using the defined transition modeling function (i.e., we convert the obtained values to probabilities by row-normalization).

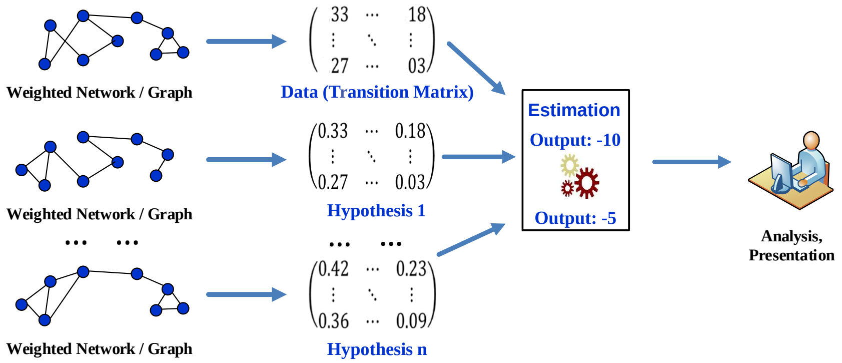


Fig. 1. Overview on the HypGraphs modeling and analysis process, cf., [12] for more details.

3.2 Modeling

For explicitly observed sequences we can simply construct transition matrices counting the transitions between the individual states, e. g., corresponding to the set of alarms (or according abstractions for aggregating sets of alarms). Then, $\tau(i, j) = |suc(i, j)|$, where $suc(i, j)$ denotes the successive sequences from state i to state j contained in the sequence. For a derived data matrix, e. g., given by calculating similarities between log entries, we typically normalize the obtained transition values. In the (general) continuous case, for example, this can be achieved by a suitable transformation on the values, e. g., simply dividing by the minimum value, or by interpreting the obtained values according to some prior distribution, e. g., to the static distribution on the (static) network (for more complex processing see [11]).

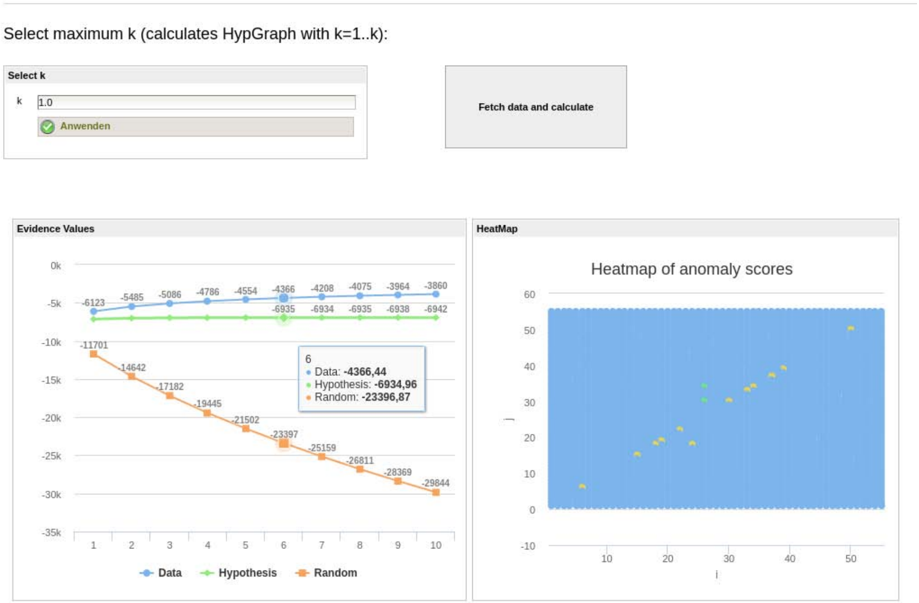
For assessing a set of hypotheses that consider different transition probabilities between the respective states, we apply the core Bayesian estimation step of HypTrails [28] for comparing a set of hypotheses representing beliefs about transitions between states. In summary, we utilize Bayesian inference on a first-order Markov chain model. As an input, we provide a (data) matrix, containing the transitional information (frequencies) of transition between the respective states, according to the (observed) data. In addition, we utilize a set of hypotheses given by (row-normalized) stochastic matrices, modelling the given hypotheses. The estimation method outputs an evidence value, for each hypothesis, that can be used for ranking. Also, using the evidence values, we can compare the hypotheses in terms of their significance. We refer to [11, 12, 28] for more details on modelling and inference, respectively.

As an alternative, we can apply the quadratic assignment procedure [18] (QAP) as a frequentist approach for comparing network structures. For comparing two graphs G_1 and G_2, it estimates the correlation of the respective adjacency matrices [18] and tests a given graph level statistic, e. g., the graph covariance, against a QAP null hypothesis. QAP compares the observed graph correlation of (G_1, G_2) to the distribution of the respective resulting correlation scores obtained on repeated random row and column permutations of the adjacency matrix of G_2. As a result, we obtain a correlation value and a statistical significance level according to the randomized distribution scores.

4 Process Model & Implementation

In the case of the FEE project and large-scale application in production, a distributed storage and computation system can handle the requirements of evaluating several years of production data. The RapidMiner [21] platform, e. g., offers with Radoop a simple integration to Hadoop systems and is able to build preprocessing and analytical processes on a local machine and transfers them to a big data environment. It is also Open Source and its functionality can be extended with self written code.

Overall, in the context of the FEE project our goal is to build a two layered computation architecture. Long running and computational expensive processes will run in the Hadoop infrastructure and either the prepared data or the final models, in this case the set of transition matrices, can be applied on a local machine. By running the computation on Spark/MapReduce [13], for example, and orchestrating the data with RapidMiner, the deployment process is speed up even further. Also the results can easily be visualized for the process engineer, for example as a heatmap of anomaly scores and be embedded in a dashboard, cf., Figure 2.



Select maximum k (calculates HypGraph with k=1..k):

Select k

k 1.0

Fig. 2. RapidMiner Server Dashboard: By adjusting the parameter k we can tune our belief in the hypotheses, cf., [11]. Then, the resulting *evidence* values (depending on the parameter k, x-axis) allows for a ranking of these compared to the *data* and to a randomized hypothesis (null model).

For the process we can load the data from local storage, but chunks of the live data can easily be accessed with a database query. Figure 3 shows examples of sequential (abnormal and normal) transitions in a network visualization. Such a visualization can be used for data exploration or be embedded into the dashboard discussed above.

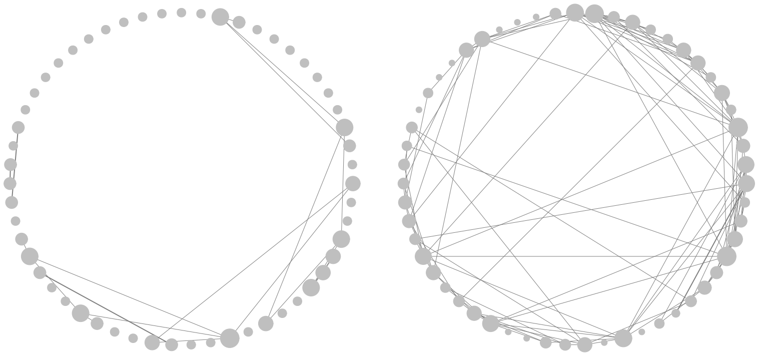


Fig. 3. Examples of transition network visualizations: Anomaly (left) vs. normal state (right). The nodes of the network denote aggregations of different alarm sources, where the size of a node denotes the shares of outgoing alarm notifications, and an edges denotes at least one transition (without self loops). These can also be filtered according to subsets of functional plant units. The figure shows such a case, where normal and abnormal situations show different characteristics and can be clearly distinguished.

Starting with process data from a streaming data source (or also historic data, e. g., as flat files) Figure 4 illustrates the inner loop of the data flow in the RapidMiner process. The calculated evidence values are collected in a table for further processing, for example in an interactive dashboard as shown in Figure 2. In the industrial context such a dashboard can be used as a standalone tool for analysing the production process online, or for evaluating historic data in order to optimize the process offline.

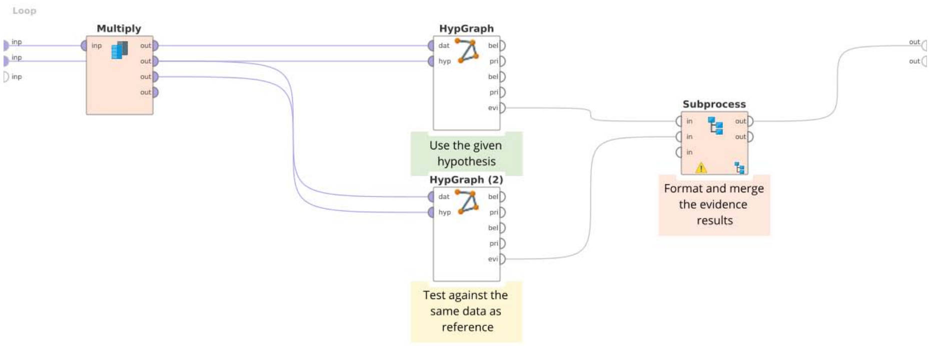


Fig. 4. Exemplary RapidMiner Process

Thus, a predefined RapidMiner process can simply be started by an engineer in order to get feedback of the current production state that can then be easily interpreted by the dashboard visualization.

5 Conclusions

This paper presented a sequential modelling and anomaly analytics approach in an industrial application context. Based on first order Markov chain models and methods for modelling and comparing networks and transition matrices [11, 12, 28], we sketched an approach for comparing hypotheses with observed "reference" sequential patterns. In that way, we can identify deviating (abnormal) and conforming (normal) hypotheses, in order to support anomaly analytics and diagnostics. Finally, we demonstrated the application of the proposed approach implemented as a RapidMiner operator.

For future work, we aim at extending the visualization options in order to support further introspective analytics options. For enabling (semi-automatic) techniques for detecting exceptional sequential trails and patterns, e. g., [6, 10], the according adaptation and extension of the presented approach in order to enable integrated anomaly detection and analysis seems promising. Furthermore, the development of comprehensive Big Data software architectures for plant operator support, e. g., [17] is another interesting direction for future research.

Acknowledgements

This work was funded by the BMBF project FEE under grant number 01IS14006E. We wish to thank Florian Lemmerich (GESIS, Cologne) for discussions on HypTrails [28].

References

1. Aalst, W.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer-Verlag, Berlin (2011)
2. Abele, L., Anic, M., Gutmann, T., Folmer, J., Kleinsteuber, M., Vogel-Heuser, B.: Combining knowledge modeling and machine learning for alarm root cause analysis. In: MIM. pp. 1843–1848. International Federation of Automatic Control (2013)
3. Akoglu, L., Tong, H., Koutra, D.: Graph Based Anomaly Detection and Description. Data Min Knowl Disc 29(3), 626–688 (May 2015)
4. Atzmueller, M.: Analyzing and Grounding Social Interaction in Online and Offline Networks. In: Proc. ECML-PKDD. LNCS, vol. 8726, pp. 485–488. Springer, Heidelberg, Germany (2014)
5. Atzmueller, M.: Data Mining on Social Interaction Networks. Journal of Data Mining and Digital Humanities 1 (June 2014)
6. Atzmueller, M.: Detecting Community Patterns Capturing Exceptional Link Trails. In: Proc. IEEE/ACM ASONAM. IEEE Press, Boston, MA, USA (2016)
7. Atzmueller, M.: Local Exceptionality Detection on Social Interaction Networks. In: Proc. ECML-PKDD 2016: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases. Springer, Heidelberg, Germany (2016)
8. Atzmueller, M., Doerfel, S., Hotho, A., Mitzlaff, F., Stumme, G.: Face-to-Face Contacts at a Conference: Dynamics of Communities and Roles. In: Modeling and Mining Ubiquitous Social Media, LNAI, vol. 7472. Springer, Heidelberg, Germany (2012)
9. Atzmueller, M., Kloepper, B., Mawla, H.A., Jäschke, B., Hollender, M., Graube, M., Arnu, D., Schmidt, A., Heinze, S., Schorer, L., Kroll, A., Stumme, G., Urbas, L.: Big Data Analytics for Proactive Industrial Decision Support: Approaches First Experiences in the Context of the FEE Project. atp edition 58(9) (2016)

10. Atzmueller, M., Mollenhauer, D., Schmidt, A.: Big Data Analytics Using Local Exceptionality Detection. In: Enterprise Big Data Engineering, Analytics, and Management. IGI Global, Hershey, PA, USA (2016)
11. Atzmueller, M., Schmidt, A., Kibanov, M.: DASHTrails: An Approach for Modeling and Analysis of Distribution-Adapted Sequential Hypotheses and Trails. In: Proc. WWW 2016 (Companion). IW3C2 / ACM, New York, NY, USA (2016)
12. Atzmueller, M., Schmidt, A., Kloepper, B., Arnu, D.: HypGraphs: An Approach for Modeling and Comparing Graph-Based and Sequential Hypotheses. In: Proc. ECML-PKDD Workshop on New Frontiers in Mining Complex Patterns (NFMCP). Riva del Garda, Italy (2016)
13. Becker, M., Mewes, H., Hotho, A., Dimitrov, D., Lemmerich, F., Strohmaier, M.: SparkTrails: A MapReduce Implementation of HypTrails for Comparing Hypotheses About Human Trails. In: Proc. WWW (Companion). ACM Press, New York, NY, USA (2016)
14. Folmer, J., Schuricht, F., Vogel-Heuser, B.: Detection of temporal dependencies in alarm time series of industrial plants. Proc. 19th IFAC World Congr pp. 24–29 (2014)
15. Hawkins, D.: Identification of Outliers. Chapman and Hall, London, UK (1980)
16. Kibanov, M., Atzmueller, M., Scholz, C., Stumme, G.: Temporal Evolution of Contacts and Communities in Networks of Face-to-Face Human Interactions. Science China Information Sciences 57 (March 2014)
17. Klöpper, B., Dix, M., Schorer, L., Ampofo, A., Atzmueller, M., Arnu, D., Klinkenberg, R.: Defining Software Architectures for Big Data Enabled Operator Support Systems. In: Proc. INDIN. IEEE Press, Boston, MA, USA (2016)
18. Krackhardt, D.: QAP Partialling as a Test of Spuriousness. Social Networks 9, 171–186 (1987)
19. Lempel, R., Moran, S.: The Stochastic Approach for Link-Structure Analysis (SALSA) and the TKC Effect. Computer Networks 33(1), 387–401 (2000)
20. Macek, B.E., Scholz, C., Atzmueller, M., Stumme, G.: Anatomy of a Conference. In: Proc. ACM Hypertext. pp. 245–254. ACM Press, New York, NY, USA (2012)
21. Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., Euler, T.: Yale: Rapid prototyping for complex data mining tasks. In: Proc. KDD. pp. 935–940. ACM, New York, NY, USA (2006)
22. Mitzlaff, F., Atzmueller, M., Benz, D., Hotho, A., Stumme, G.: Community Assessment using Evidence Networks. In: Analysis of Social Media and Ubiquitous Data. LNAI, vol. 6904 (2011)
23. Mitzlaff, F., Atzmueller, M., Hotho, A., Stumme, G.: The Social Distributional Hypothesis. Journal of Social Network Analysis and Mining 4(216) (2014)
24. Munoz-Gama, J., Carmona, J., van der Aalst, W.M.P.: Single-entry single-exit decomposed conformance checking. Inf. Syst. 46, 102–122 (2014)
25. Pirolli, P.L., Pitkow, J.E.: Distributions of Surfers' Paths Through the World Wide Web: Empirical Characterizations. World Wide Web 2(1-2) (1999)
26. Ranshous, S., Shen, S., Koutra, D., Harenberg, S., Faloutsos, C., Samatova, N.F.: Anomaly Detection in Dynamic Networks: A Survey. WIREs: Comput. Statistics 7(3), 223–247 (2015)
27. Rozinat, A., Aalst, W.: Conformance Checking of Processes Based on Monitoring Real Behavior. Information Systems 33(1), 64–95 (2008)
28. Singer, P., Helic, D., Hotho, A., Strohmaier, M.: Hyptrails: A Bayesian Approach for Comparing Hypotheses about Human Trails. In: Proc. WWW. ACM, New York, NY, USA (2015)
29. Singer, P., Helic, D., Taraghi, B., Strohmaier, M.: Memory and Structure in Human Navigation Patterns. PLoS ONE 9(7) (2014)
30. Strelioff, C.C., Crutchfield, J.P., Hübler, A.W.: Inferring Markov Chains: Bayesian Estimation, Model Comparison, Entropy Rate, and Out-of-Class Modeling. Physical Review E 76(1), 011106 (2007)
31. Vogel-Heuser, B., Schütz, D., Folmer, J.: Criteria-based alarm flood pattern recognition using historical data from automated production systems (aps). Mechatronics 31, 89–100 (2015)