Workaround Mining in Health Information Systems

Wouter van der Waal[1[0000−0001−6514−7570]], Inge van de Weerd[1[0000−0003−3420−6923]], Iris Beerepoot[1[0000−0002−6301−9329]], Niels Martin[2[0000−0003−3279−3853]], and Hajo A. Reijers[1[0000−0001−9634−5852]]

[1] Utrecht University, the Netherlands
{w.g.vanderwaal,g.c.vandeweerd,i.m.beerepoot,h.a.reijers}@uu.nl
[2] Hasselt University, Belgium niels.martin@uhasselt.be

Keywords: Health Information Systems · Workarounds · Machine Learning.

Introduction Workarounds are process variants that differ from set procedures, usually when a user perceives a block on the designed path [2]. For example, when a patient needs medication, a physician may have to prescribe it. If this has not happened in time, a nurse might solve the problem by prescribing one-time medication instead.

Knowledge about workarounds can be useful to prevent dangerous situations or optimize difficult procedures. Finding workarounds is easier said than done, however. Most workarounds have been discovered through interviews with experts. While this has resulted in valuable information, it takes a lot of time. More important, there may be workarounds that experts were unaware of or were not willing to share [9]. We propose an objective method to detect workarounds using existing event logs from health information systems.

The authors from [9] use a deep learning method to detect workarounds from an event log. However, the explainability of neural networks is generally low [8]. In [7], it is shown that we can detect workarounds using a rule-based system. For example, we can detect a certain workaround by defining a set of authorized users for an activity and searching the log for any deviations of the rule. While this method will find specified patterns, most rules cannot be generalized to detect new workarounds.

Approach To detect workarounds, we use data on process instances. Depending on the workaround type, we look at different process perspectives [1]. For example, the time spent on a process can be used to discover Time-workarounds.

Starting from this data, we can approach this as a classification problem. This means we can use classical machine learning methods, such as Naive Bayes [5] or Support Vector Machines [6]. These methods are especially powerful if the possible classes (workaround and normative) are dissimilar. For example, they would perform well if workarounds take longer to complete than normative process instances.

As an alternative, we will look at clustering algorithms, such as k-means [4] and DBSCAN [3]. These algorithms work very well if occurrences of the same class are very similar to each other. So if, for example, the normative process instances would all take about the same time, clustering algorithms would group them together. While results from this approach are more difficult to interpret, it does have the advantage that is does not require more expert input beforehand.

Expected Contributions This research will provide a novel approach for data-driven workaround discovery. We will determine which process instance dimensions are interesting for distinguishing between normative process variants and workarounds. Most likely, this will differ between different types of workarounds.

In addition, we can find new workarounds. It would be valuable to know why these workarounds were not discovered through interviews. Otherwise, they are still interesting for the hospital in the traditional sense.

In conclusion, we propose a more objective approach for workaround discovery. Aside from being less labour-intensive, it would give a more complete view of the workarounds that exist, allowing hospitals to address the problematic ones or adjust the procedures.

References

1. Beerepoot, I., Lu, X., Van De Weerd, I., Reijers, H.A.: Seeing the Signs of Workarounds: A Mixed-Methods Approach to the Detection of Nurses' Process Deviations. In: Proceedings of the 54th Hawaii International Conference on System Sciences (2020). https://doi.org/10.24251/HICSS.2021.456
2. Ejnefjäll, T., Ågerfalk, P.J.: Conceptualizing Workarounds: Meanings and Manifestations in Information Systems Research. Communications of the Association for Information Systems **45**(1), 340–363 (2019). https://doi.org/10.17705/1CAIS.04520
3. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: kdd. vol. 96, pp. 226–231 (1996)
4. Hartigan, J.A., Wong, M.A.: Algorithm AS 136: A K-Means Clustering Algorithm. Journal of the royal statistical society. series c (applied statistics) **28**(1), 100–108 (1979). https://doi.org/10.2307/2346830
5. Lewis, D.D.: Naive (bayes) at forty: The independence assumption in information retrieval. In: European conference on machine learning. pp. 4–15. Springer (1998). https://doi.org/10.1007/BFb0026666
6. Osuna, E., Freund, R., Girosi, F.: An improved training algorithm for support vector machines. In: Neural networks for signal processing VII. Proceedings of the 1997 IEEE signal processing society workshop. pp. 276–285. IEEE (1997). https://doi.org/10.1109/NNSP.1997.622408
7. Outmazgin, N., Soffer, P.: A process mining-based analysis of business process workarounds. Software & Systems Modeling **15**(2), 309–323 (May 2016). https://doi.org/10.1007/s10270-014-0420-6
8. Ras, G., van Gerven, M., Haselager, P.: Explanation Methods in Deep Learning: Users, Values, Concerns and Challenges. In: Explainable and Interpretable Models in Computer Vision and Machine Learning. pp. 19–36. Springer (2018)
9. Weinzierl, S., Wolf, V., Pauli, T., Beverungen, D., Matzner, M.: Detecting Workarounds in Business Processes-a Deep Learning method for Analyzing Event Logs. In: Proceedings of the 28th European Conference on Information Systems (2020)