Identifying Variation in Personal Daily Routine Through Process Mining: A Case Study

Gemma Di Federico[1][0000−0002−2487−1164], Carlos Fernández-Llatas[2][0000−0002−2819−5597], Zahra Ahmadi[3][0000−0002−2313−5392], Mohsen Shirali[4][0000−0003−0584−4360], and Andrea Burattin[1][0000−0002−0837−0183]

[1] Technical University of Denmark, Kgs. Lyngby, Denmark
[2] Universitat Politècnica de València, Valencia, Spain
[3] LIRIS, KU Leuven, Brussels, Belgium
[4] Shahid Beheshti University, Tehran, Iran
gdfe@dtu.dk

Abstract. The study of daily routines has gained substantial attention, especially in healthcare. Understanding the activities and behaviors of individuals, particularly older adults, has the potential to play a crucial role in providing effective care and support, for example, when it comes to spotting deviations from it automatically.

Process mining is a valuable tool for analyzing routine dynamics and identifying variations. However, human behavior is unstructured and characterized by variability, making it difficult to derive a process model representing only the control flow.

In this paper, we employ a multi-dimensional process discovery and conformance checking methodology to a real-world dataset representing a person's behavior in a smart environment. The derived model combines control flow and statistics on the data. The results, on the real-world data, highlight that the approach can identify variations in the inhabitant's behavior.

1 Introduction

The study of daily routines has gained substantial attention, especially within the healthcare field. Understanding the activities and behavior of individuals, particularly older adults, has the potential to play a crucial role in providing effective care and support [6]. There is evidence in the literature demonstrating the importance of deriving individual behavioral models that contribute to personalized healthcare [6, 13]. For example, an early diagnosis of dementia can be made through the detection of alterations in daily habits [5].

Process mining is a powerful methodology to analyze and extract information from event logs, allowing experts to gain insight into the dynamics of daily routines. In healthcare, process mining can contribute to a deeper comprehension of patients' activities, their patterns, and potential deviations from expected routines [11]. The study is made possible by the use of smart environments [23] in which sensors collect information about the activities carried out by the inhabitants. By applying process mining techniques to event data captured through sensor networks or wearable devices [3], a model representing the daily routine of the patient can be derived. The continuous monitoring of the behavior throughout the day can help identify variations and provide medical support to the patient.

The derivation of a model that faithfully represents and abstracts the individual daily behavior is a challenging task [9]. In fact, routines are not fixed as they are dependent on the context, and result in the variability of the behaviors. Variations can be an alternative way to accomplish a task or the introduction of a new behavior. Furthermore, although it might be possible to predict the set of activities that compose a routine, we cannot define with certainty their order of execution. All of these factors make it inadequate to derive a process model that represents only the control flow.

For this reason, Di Federico et al. [8] implemented an algorithm able to derive a multi-dimensional model composed not only by the control flow but also embedding the data dimension in the form of various statistics. The approach derives multiple process models, both declarative and imperative, together with statistics on the activities. The approach can also be used to identify variations in the execution of the routine itself by proposing a conformance checking algorithm that balances control-flow alignment and data.

In this paper, we demonstrate the capability of this approach in recognizing variations of behaviors through its application to a real-world dataset [12]. The dataset collects information regarding the daily routine of a person living in a smart environment. Differently from other publicly available datasets, in this case, we have evidence of events that impacted the routine of the inhabitant (i.e., the ground truth of the routine variation), and we are able to identify these changes.

The paper is structured as follows. Section 2 presents the state of the art of the analysis of routines. The algorithm used in the work is explained in Section 3, and its application is further elaborated in Section 4, i.e. the use case of this paper. The results are discussed in Section 5. Section 6 concludes the paper.

2 State of the art

Human behavior can be defined as a combination of intentions, perceptions and states. The execution of a behavior, and its purpose, is influenced by all of these factors [9]. Human beings establish routines and perform them repetitively [2]. However, routines are not static; rather, they exhibit variations over time and are greatly influenced by situational factors, reflecting the unpredictable nature of human behavior. The main challenge when analyzing human behavior is whether the behavior is structured enough to be represented using a process model. In fact, behavioral models usually result in spaghetti-like models [17]. As a consequence, the identification of variations becomes even more challenging. A variation in the daily behavior can be defined as an unexpected but significant irregularity in otherwise normal data, which could be indicative of an adverse condition [23].

Several approaches in the literature tried to discover a behavioral model for daily routines. Palermo et al. [21] propose an approach to analyze the risk of agitation in people living with Dementia. Unusual patterns of activities could be an indication of an agitation episode. The authors construct a Long-Short Term Memory (LSTM) network from the behavior of 46 patients to characterize their daily routine. The approach predicts whether the behavior of an individual is classified as agitated or not agitated. Nevertheless, the behavioral model is neither personalized nor easily interpretable. Specifically, it only returns the predicted class, and no additional information can be provided.

Dogan et al. [11] apply process mining techniques to discover and classify human behavior patterns. The authors use a discovery algorithm to identify trajectories in the smart environment, then a clustering technique is applied to identify behavioral patterns, subsequently represented using a calendar view. It is noteworthy that the resulting models are not individual, but they represent groups of people, and then each participant is assigned to a group. The approach classifies deviations as a difference between the participant's behavior and the modeled one. However, the reference model is a generalization and doesn't take into account the specific person in detail. What is more, the approach only considers the process flow perspective. When representing human behaviors, other dimensions should be considered, e.g., the excessive repetition of an activity cannot be captured by the control flow alone.

Di Federico et al. [8] propose an approach to derive a personalized behavioral model which combines control flow and data dimensions. The authors argue the importance of considering statistics on the data while deriving a model. What is more, they propose both imperative and declarative modeling languages to represent the control flow, as there is not yet a conclusive solution capable of dealing with routine behavior. However, so far, the approach has only been tested on simulated data [10].

In the context of this paper, we apply the algorithm proposed by Di Federico et al. [8] on a real dataset. Before delving into the use case description, we describe the approach in the following background section.

3 Background

The approach used in this paper is a multi-dimension algorithm described in [8]. The algorithm proposes the discovery of both control flow models and the computation of statistics and it also verifies the conformance between such multi-dimensional model and new instances of the process. In the following sections, the two stages are explained in detail.

3.1 Process discovery

The discovery consists of deriving both the control flow and the data perspectives. Observing the process from different points of view allows us to consider all the aspects characterizing human behaviors. The control flow is composed of the discovery of three models, i.e., a Petri Net [19], a Declare model [22], and a DCR graph [15]. The Inductive Miner [16] is used to obtain the Petri Net, while for the declarative models the discovery algorithms of the corresponding languages are used [14, 18, 20]. On one hand, a declarative language represents the process in the form of constraints, therefore more suitable to abstract from the problem of variability. On the other hand, imperative languages have a more structured and clear representation of the process.

The data dimension focuses on deriving significant statistics related to the activities' frequency, duration, and occurrence time. These aspects are deemed relevant as they allow the monitoring of the repetition of activities and position them on a timeline. The activity frequency is determined by counting the occurrence of each unique activity identifier in each trace. Then, the values are aggregated to the entire event log, and mean, standard deviation, median, minimum, and maximum are computed. The activity duration is assessed through the average duration per activity identifier, with statistics like mean, median, minimum, and maximum duration computed across traces. The absolute time dimension is derived by using histograms to depict activity occurrence frequency within specific time intervals. The last dimension is the trace length, and min, max, mean, and standard deviation are derived. All these insights offer a comprehensive understanding of process behavior, unveiling recurring patterns, timing delays, and variations in activity execution.

3.2 Conformance checking

The model derived in the discovery phase can be used as a reference model in the conformance checking. The conformance algorithm presented includes the verification of all the mentioned perspectives, and returns an enriched fitness value which is an aggregation of all of them. In particular, the conformance checking between the Petri Net and new instances of the process is computed using alignments [1] (we'll refer to this value as CCInd). Additionally, the precision value computed using the alignments is also considered. For the Declare model, the algorithm implemented by Burattin et al. [4] is used. In this case, the algorithm returns the number of activations, fulfillment, and violations of each constraint in the input model and for each trace in the log, and we computed the fitness value (CCDeclare). For the conformance of the DCR graph (CCDcr), a rule checker algorithm is deployed [7]. For the verification of the data flow dimensions, comparison functions are used. For the frequency of the activities (CCFreq), the assumption is that the average value serves as a benchmark. The probability density function lets us measure how likely it is that the mean frequency value for each activity identifier in the trace is close to this reference. The same approach is adopted to obtain a fitness value for the duration (CCDur) of the activities and for the trace length (CCLen).

For the absolute time dimension (CCTime), the reference is the histogram of the frequencies of each activity over time intervals. The conformance of a new absolute time is then computed as the normalized frequency for the time interval the new absolute time belongs to.

The resulting enriched fitness value is an aggregation of the four values obtained from the conformance of the control flow dimension and the four values obtained from the conformance of the data flow dimension.

4 Application

The objective of the paper is to demonstrate that the daily routine can be modeled using a multi-dimension model and that behavioral changes can be identified. In particular, the model is personalized for an individual, therefore it is highly representative of the habits and characteristics of that person. It is important to mention that the daily routine evolves over time; hence the reference model should be updated consequently. The case study presented in this paper is the daily routine of a person living in a smart environment.

4.1 Dataset description

The dataset used in this case study refers to the behavior of a 60-year-old woman living in her private house [12]. The components of the setting are a smart home, a smartphone, and a smart wristband. The smart home is equipped with sensors used to perceive the environment and the activities performed by the resident during the day. The installed devices are 15 binary sensors positioned on furnishing elements, appliances, and doors.

With the aid of a mobile application, smartphone usage information is collected. The wristband is instead used to obtain sleeping data and to collect the number of daily steps. The data were collected in the timeframe 08/01/2020 - 02/06/2020. During the collection period, various events happened that influenced the behavior of the resident. From the 40th day from the beginning of the collection process, the Covid-19 pandemic was prevalent in the region, affecting the habits of the participant. From the 109th day to the 137th day was the Ramadan month; therefore, the resident changed their habits such as meal times, praying time and duration, and sleeping intervals.

4.2 Pre-processing

Since the objective is to monitor the behavior of the subject at the level of activities executed during the day, the dataset requires some abstraction. In fact, the dataset includes sensors' raw readings at a low level of abstraction representing the presence of the subject near the appliances, the open or closed state of doors and the use of the stove. We pre-process the raw data to extract information about the subject's presence in different locations of the house (e.g. bedroom, kitchen) by looking at the locations in which sensors are mounted. In this way, we are able to detect events related to the visited home locations for the entire dataset period. In addition, the executed activities at a higher level (for instance, sleeping, eating, watching TV) are also determined based on the sensor states. Then, for each detected location or activity, a record is added to the event log with the labels in the form of location-activity (e.g., *LivingRoom-Mobile-Cooking*, meaning that the person was in the living room, they were using the mobile phone, and something was left cooking on the stove). However, we decided to simplify the labels and rename them with the location only (e.g., the previous label is replaced by *LivingRoom*). This choice is driven by the presence of activities that occur without direct involvement from the participant, as seen in the cooking activity from the previous instance. If the person is in the living room at that moment, we are interested in that specific event. What is more, all the mobile events have an additional corresponding entry *Mobile*. All events related to the *corridor* location are removed, as the corridor can be considered a hub and not a place that actually contributes to the execution of an activity. We also removed all the events in the *bathroom* location, referring to the usage of the toilet to ensure the privacy of the resident, while we considered only those events located in the anteroom. To divide the log in cases, we group them by day. At this point, we obtained 43 278 events organized in 146 cases. What is more, it is good practice to focus on a specific routine, therefore we decided to investigate the day routine without considering the evening and the night routine. Hence, we filter the event log to preserve only events in the timeframe 10am to 6pm. Last, because of the renaming of the labels, we could observe a lot of consecutive events with the same event name, thus we merged them considering the start time as the timestamp of the first event, and the end time as the timestamp of the last consecutive event. To sum up, we obtain an event log with 146 cases, 9237 events, and 14 unique activity identifiers.

4.3 Method

For the analysis of the dataset we use the approach reported in Section 3. An important factor to consider while analyzing human behavior is that the routine evolves over time. For this reason, we have applied a sequential approach, in which we consider two weeks to derive the reference model, and one week for the conformance, with a sliding of 1 week for the subsequent training/testing. We decided on this strategy to accommodate the natural evolution of routines that might naturally happen over the course of time. In total, we obtained 18 logs for training and 18 for testing. Once we applied the algorithm, we obtained results for 18 data points, i.e., a result every other week. In the rest of the paper, we will refer to the discovery phase as *training*, while to the conformance checking phase as *testing*.

5 Results

In this section, we present the results obtained by the application of the multi-dimension discovery and conformance approach to the dataset under examination. In all the graphs, we'll highlight the two events that probably affect the routine (i.e., Covid-19, Ramadan) as horizontal bars in blue shades.

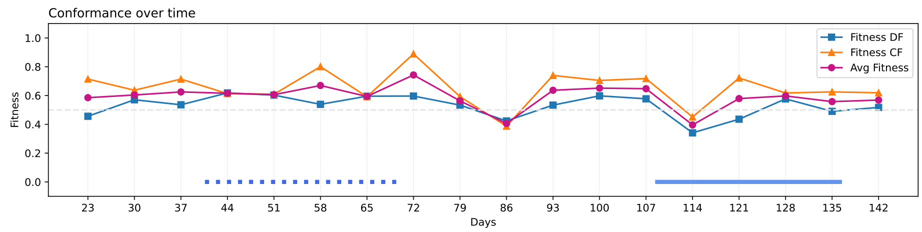

Fig. 1. Fitness value given by the average between the control flow perspective (CCInd, CCDcr and CCDeclare) and the data flow perspective (all)

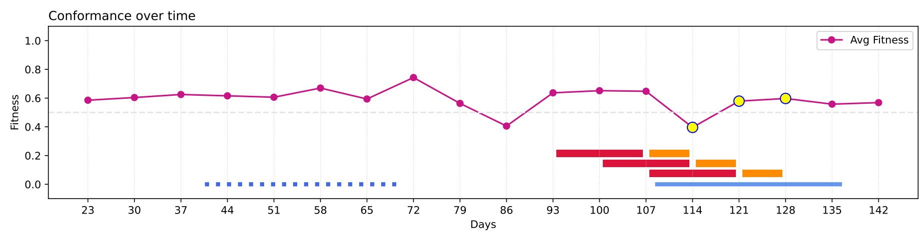

Fig. 2. Propagation of the variations on the results. The red/orange bars refer to training/test sets, while the yellow dots the corresponding fitness result

Prior to delving into the specifics of how to balance the different dimensions, it's important to grasp the overall direction of the analysis. For this reason, the first result we analyze is a fitness value obtained by averaging the control flow and the data flow perspectives. The values considered are the average between CCInd, CCDcr, and CCDeclare for the control flow, and the average between CCFreq, CCDur, CCTime, and CCLen for the data flow. The result is plotted in Figure 1. The plot highlights two negative trends: from day 72 to 86 and from day 107 to 114. In general, control flow and data flow follow the same trend, especially in the second half of the graph.

In order to understand these downward spikes, we have to understand the propagation of events, specifically the time required for the reference model to incorporate the new routine. Let's consider as an example Figure 2 and the Ramadan event, starting from day 109. In the figure we indicate the training period using a red bar, and the testing period using a orange bar. Taking the interval of days 94 to 106 for model training (prior to Ramadan), and days 107 to 113 for testing (the initial week of Ramadan), a divergence emerges between the expected behavior and the observed one. In fact, the Fitness value drops below 0.5 on the 114th day. Progressing to the subsequent application (the second set of red/orange horizontal bars), we note that the new routine is gradually integrated into the training set, hence it is becoming more representative of the Ramadan behavior, and this is the reason why the fitness value starts increasing again. In the following iteration, marked by the final pair of red/orange horizontal bars,

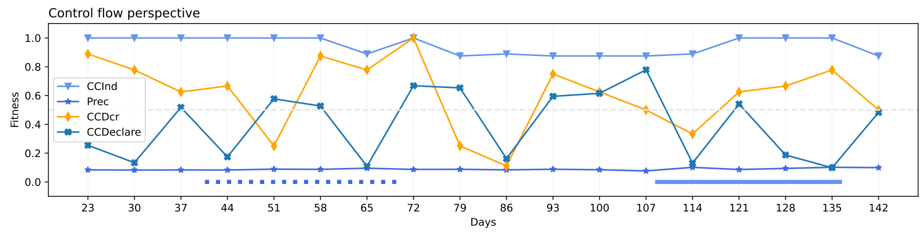

Fig. 3. All the control flow dimensions: fitness values obtained by CCInd, CCDcr, CCDeclare, and the Precision value

the training set fully incorporates the Ramadan routine. As a result, the observed behavior aligns perfectly, causing the fitness value to rise and stabilize.

Before analyzing the motivations behind the negative spikes, we have to investigate how to balance the different components of the approach. In fact, especially in the case of the control flow, we consider three different modeling languages and discovery/conformance algorithms, which all have different characteristics. For a faithful representation of human behavior, the modeling language must allow a certain level of flexibility in the execution flow. However, when deriving the Petri Nets by using the Inductive Miner, the models exhibit low precision and have a propensity to underfit the behavior. On the declarative side, Declare has demonstrated a higher degree of strictness by deriving notably more constraints when compared to DCR. The Fitness values and the precision are plotted in Figure 3. The graph illustrates a significant difference between the CCInd Fitness values and the Precision. The Fitness values are consistently high, indicating a strong match between the model and the observed behavior. However, Precision values remain close to zero throughout the graph. The divergence between the two metrics underscores a potential gap between overall alignment with the observed behavior and the model's capacity to generalize the behavior. Yet, as previously mentioned, relying solely on the control flow is insufficient for comprehending human behavior; hence, the viewpoint of data flow also demands consideration. Figure 4 plots the Fitness values of the data flow. It can be noticed that CCDur, CCFreq, and CCTime roughly follow the same trend, whereas CCLen shows two negative spikes, followed by a higher positive one, on day 86 and on day 114.

To find a balance between all the dimensions, we make use of the moving average so that we can identify peaks more accurately. We compute different configurations to obtain the total fitness value, which is then used to compute the moving average and the corresponding signal for the spike. If the difference in the signal between two data points is greater than 0.1, it is labeled as a spike.

Figure 5 illustrates the spike detector across three distinct configurations. In the first run, a balanced configuration was employed, allocating equal weight to all dimensions, that is the same weight we gave to the dimensions to compute the

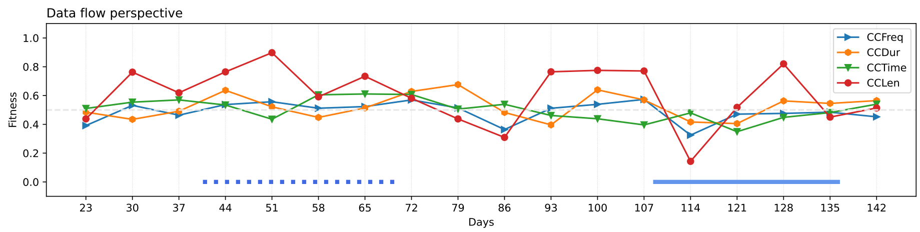

Fig. 4. All the data flow dimensions: fitness values obtained by CCFreq, CCDur, CCTime, CCLen

average fitness values plotted in Figure 1. The calculation highlights a spike on day 114, as expected. Contrary to what was observed before, the downward trend on day 86 is not identified as a peak. This is due to the gradual nature of the decrease, which spans over more than a week. As a result, the difference between the moving averages of consecutive data points does not satisfy the minimum threshold employed to confirm the occurrence of a spike. In the second and third configurations, we only consider Precision and CCDcr in the control flow, while for the data flow all the dimensions are considered. In the case of the second configuration, when calculating the average fitness value between the control flow and data flow, greater emphasis is given to the control flow dimension. Specifically, a higher significance is attributed to control flow by assigning it a weight of $4/5$, whereas data flow contributes with a weight of $1/5$. The resulting signal is plotted on the second graph in Figure 5. In this case, two spikes are identified, meaning that the control flow identifies a variation in the behavior on the data point at day 51.

The signal function depicted in the third graph adopts an inverse weighting compared to the prior setup. Here, the data flow has more importance, contributing for $4/5$ to the average fitness, while the control flow $1/5$. In this case, the spike at day 51 is not identified, meaning that the variation is not reflected in the data flow.

As the last result, we compare the behavior of the inhabitant at two different moments in time. In particular, we examine the behavior during Ramadan and the behavior preceding that period. When selecting the time frames, we consider the signal function in the first graph of Figure 5, opting for a date when the fitness fluctuation is low. The pre-Ramadan behavior is selected from day 44 to 64, while the period during Ramadan is from day 107 to 127. We use the same setup as before: two weeks for training and one for testing. Firstly we train and test a model for each timeframe independently to verify if the behavior is constant. The fitness value is set up according to the balanced configuration explained above. The result shows that the tested behaviors are actually the ones depicted by the corresponding training log, with a fitness value of 0.7 and 0.62 respectively. Delving into the differences, we notice that the *Eating* activity is not present in the log during Ramadan, in fact the mean value of the frequency statistic

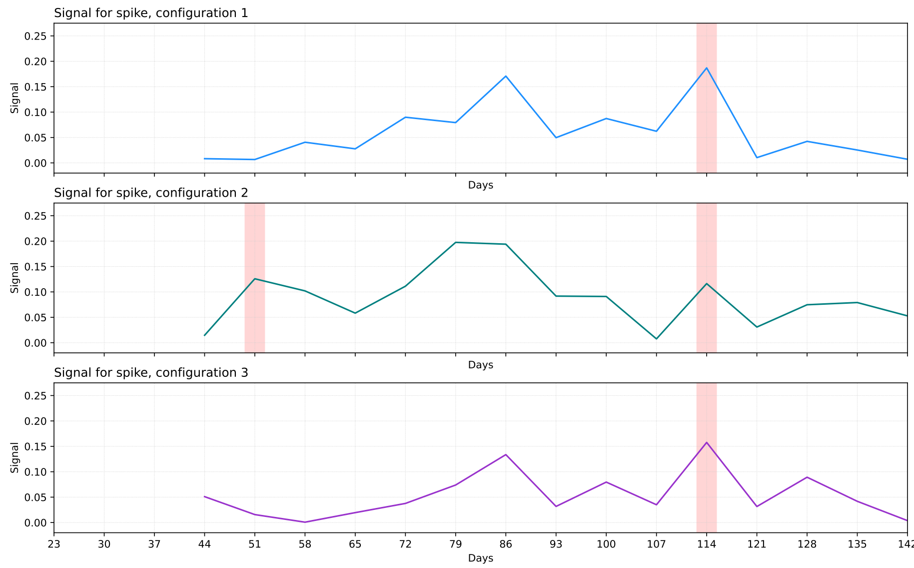

Fig. 5. Spike detection applied on three different configurations of the signal. A spike is identified when there is an increase of 0.1 in two subsequent points.

for this activity is 0.8 before Ramadan and 0.1 during Ramadan. The reason is that during Ramadan the *eating* activity is performed early in the morning or late in evening, so it is not executed during the timeframe we are considering. To conclude, we check if the behavior during Ramadan (days 121-128) can be represented by the model pre-Ramadan. In this case, the obtained fitness value is 0.4, indicating a difference between the behaviors.

Finally, we link the identified variation to the events which probably impacted the behavior of the participant. The first event is Covid-19 (from day 40). In the spike detector in Figure 5 the signal function starts at day 44, as 4 weeks are used as a subset to compute the first moving average value. Therefore, we cannot characterize the difference between the behavior before and during Covid-19 as we don't have enough data to analyze before this event. In addition, there is no end to the Covid-19 behavior in the recorded period, hence the only difference that can be identified is the beginning of the pandemic. The second event is the Ramadan period (from day 109 to 137). In this case, we have both the start and end of the event, but the recording period terminates only one week after, hence we do not have the time to propagate and recognize the end of Ramadan. However, the beginning of Ramadan is recognized by the approach across all configurations of the signal function, as well as by both the control flow and data flow independently.

6 Conclusions

In conclusion, the presented case study has demonstrated the effectiveness of the employed methodology in identifying significant variations within an individual's daily routine. By using an algorithm that integrates both the control flow and the data flow, the work derived a reference behavioral model representing the daily routine of the participant. Then, a conformance checking algorithm is applied to verify the discrepancies between the expected and the actual behavior. The model is updated every 7 days, to reflect changes in the routine. The findings demonstrated the effectiveness of the approach in identifying significant events that signify variations in the participant's behavior.

To sum up, this paper shows the capability of the approach presented in [8] which can automatically detect variations in the behavior of a subject during their time in a smart home. Proving the capability of such a system on real data paves the way to its application in contexts where the actual change in behavior is not known in advance and which could result in impactful in new disciplines (e.g., Dementia worsening, as discussed before). This also represents the main endeavor for the future: deploying the technique on new domains, where no information is known.

References

1. Van der Aalst, W., Adriansyah, A., Van Dongen, B.: Replaying history on process models for conformance checking and performance analysis. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **2**(2), 182–192 (2012)
2. Banovic, N., Buzali, T., Chevalier, F., Mankoff, J., Dey, A.K.: Modeling and understanding human routine behavior. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. pp. 248–260 (2016)
3. Bertrand, Y., Van den Abbeele, B., Veneruso, S., Leotta, F., Mecella, M., Serral, E.: A survey on the application of process discovery techniques to smart spaces data. Engineering Applications of Artificial Intelligence **126**, 106748 (2023)
4. Burattin, A., Maggi, F.M., Sperduti, A.: Conformance checking based on multi-perspective declarative process models. Expert systems with applications **65**, 194–211 (2016)
5. Chalmers, C., Fergus, P., Montanez, C.A.C., Sikdar, S., Ball, F., Kendall, B.: Detecting activities of daily living and routine behaviours in dementia patients living alone using smart meter load disaggregation. IEEE Transactions on Emerging Topics in Computing **10**(1), 157–169 (2020)
6. Chimamiwa, G., Giaretta, A., Alirezaie, M., Pecora, F., Loutfi, A.: Are smart homes adequate for older adults with dementia? Sensors **22**(11), 4254 (2022)
7. Debois, S., Hildebrandt, T.T., Laursen, P.H., Ulrik, K.R.: Declarative process mining for dcr graphs. In: Proceedings of the Symposium on Applied Computing. pp. 759–764 (2017)
8. Di Federico, G., Burattin, A.: Do you behave always the same? a process mining approach. In: International Conference on Process Mining. pp. 5–17. Springer (2022)
9. Di Federico, G., Burattin, A., Montali, M.: Human behavior as a process model: Which language to use? In: ITBPM@ BPM. pp. 18–25 (2021)

10. Di Federico, G., Nikolajsen, E.R., Azam, M., Burattin, A.: Linac: A smart environment simulator of human activities. In: International Conference on Process Mining. pp. 60–72. Springer International Publishing Cham (2021)
11. Dogan, O., Martinez-Millana, A., Rojas, E., Sepúlveda, M., Munoz-Gama, J., Traver, V., Fernandez-Llatas, C.: Individual behavior modeling with sensors using process mining. Electronics **8**(7), 766 (2019)
12. Falah Rad, M., Shakeri, M., Khoshhal Roudposhti, K., Shakerinia, I.: Probabilistic elderly person's mood analysis based on its activities of daily living using smart facilities. Pattern Analysis and Applications pp. 1–14 (2022)
13. Fernández-Llatas, C., Benedi, J.M., García-Gómez, J.M., Traver, V.: Process mining for individualized behavior modeling using wireless tracking in nursing homes. Sensors **13**(11), 15434–15451 (2013)
14. Hildebrandt, T., Mukkamala, R.R.: Declarative event-based workflow as distributed dynamic condition response graphs. PLACES **69** (10 2011). https://doi.org/10.4204/EPTCS.69.5
15. Hildebrandt, T.T., Mukkamala, R.R.: Declarative event-based workflow as distributed dynamic condition response graphs. In: Electronic Proceedings in Theoretical Computer Science 69 (2011). pp. 59—73 (2011)
16. Leemans, S.J., Fahland, D., Van Der Aalst, W.M.: Discovering block-structured process models from event logs-a constructive approach. In: Application and Theory of Petri Nets and Concurrency: 34th International Conference, PETRI NETS 2013, Milan, Italy, June 24-28, 2013. Proceedings 34. pp. 311–329. Springer (2013)
17. Leotta, F., Mecella, M., Mendling, J.: Applying process mining to smart spaces: Perspectives and research challenges. In: Advanced Information Systems Engineering Workshops: CAiSE 2015 International Workshops, Stockholm, Sweden, June 8-9, 2015, Proceedings 27. pp. 298–304. Springer (2015)
18. Maggi, F.M., Mooij, A.J., Van der Aalst, W.M.: User-guided discovery of declarative process models. In: 2011 IEEE symposium on computational intelligence and data mining (CIDM). pp. 192–199. IEEE (2011)
19. Murata, T.: Petri nets: Properties, analysis and applications. Proceedings of the IEEE **77**(4), 541–580 (1989)
20. Nekrasaite, V., Parli, A.T., Back, C.O., Slaats, T.: Discovering responsibilities with dynamic condition response graphs. In: International Conference on Advanced Information Systems Engineering. pp. 595–610. Springer (2019)
21. Palermo, F., Kouchaki, S., Li, H., Capstick, A., Fletcher-Lloyd, N., Zhao, Y., Nilforooshan, R., Sharp, D., Barnaghi, P.: Designing a clinically applicable deep recurrent model to identify neuropsychiatric symptoms in people living with dementia using in-home monitoring data. In: 35th Conference on Neural Information Processing Systems (NeurIPS 2021) (2021)
22. Pesic, M., Schonenberg, H., Van der Aalst, W.M.: Declare: Full support for loosely-structured processes. In: 11th IEEE international enterprise distributed object computing conference (EDOC 2007). pp. 287–287. IEEE (2007)
23. Rashidi, P., Mihailidis, A.: A survey on ambient-assisted living tools for older adults. IEEE journal of biomedical and health informatics **17**(3), 579–590 (2012)