Extracting Service Process Models from Location Data

Ye Zhang, Olli Martikainen, Riku Saikkonen, Eljas Soisalon-Soininen

HAL Id: hal-01769760
https://inria.hal.science/hal-01769760

Submitted on 18 Apr 2018

Extracting Service Process Models from Location Data

Ye Zhang, Olli Martikainen, Riku Saikkonen, and Eljas Soisalon-Soininen

Aalto University, Finland
{ye.zhang, riku.saikkonen, eljas.soisalon-soininen}@aalto.fi,
olli.martikainen@pfu.fi

Abstract. Services are today over 70% of the Gross National Product in most developed countries. The productivity improvement of services is increasingly important and it relies heavily on a deep understanding of the service processes. However, how to collect data from services has been a problem and service data is largely missing in national statistics, which brings challenges to service process modelling.

This work aims to simplify the procedure of automated process modelling, and focuses on modelling generic service processes that are location-aware. An approach based on wireless indoor positioning is developed to acquire the minimum amount of location-based process data that can be used to automatically extract the process models.

The extracted models can be further used to analyse the possible improvements of the service processes. This approach has been tested and used in dental care clinics. Besides, the automated modelling approach can be used to greatly improve the traditional process modelling in various other service industries.

Keywords: process modelling, service process, location-based, automated

1 Introduction

Services are increasingly important to the economy: efficiently and effectively running service operation is the key for gaining a competitive edge in almost every industry. The productivity improvements explain the changes in the profitability of industries and in the welfare of societies. Especially, the impacts of technology and innovations can be analysed by Total Factor Productivity (TFP) improvement [4, 30]. We have developed the analysis of productivity improvements of services by calculating the effects of process changes in the service performance [11]. This is based on the comparison of the process models before and after the service improvement. To speed up the modelling of processes wireless measurement has been developed [34]. This article shows how the process model of one person can be extracted automatically from the wireless measurement data. The extraction of the process models with interacting teams of people is still under study.

The improvements of services rely heavily on a deep understanding of the service process, it is inevitable that we should consider service innovation from

the process model viewpoint. A service process model prescribes how a particular service is carried out. Partington et al. [23] demonstrated that through analysing the processes, detailed insights into clinical (quality of patient health) and fiscal (hospital budget) pressures in health care practice were provided. Rovani et al. [26] also testified that modelling the healthcare service process mediates between event data reflecting the clinical reality and clinical guidelines describing best practices in medicine.

However, how to collect process data from services has been a problem. Halonen et al. [11] documented the healthcare process of the acute neurology ward in Helsinki University Hospital based on interview data. Constructing process models from scratch with traditional approaches often requires the involvement of field experts, which is difficult, expensive and time-consuming. An alternative way applies process mining techniques to extract process models from event logs. This technique has been widely explored in the healthcare sector: Rebuge et al. [25] analysed the hospital emergency service, Mans et al. [17] studied the gynecological oncology process, Blum et al. [5] mined the laparoscopic surgery workflow. However, one limitation is that the quality of a mined model depends on whether the log covers all the dependencies between activities. The data quality of real-life logs usually is far from ideal: they are usually noisy, incomplete and imprecise [6]. Besides, the goal of most of the process mining research is to extract domain-specific workflows [18], but approaches that are capable of picturing more generic processes are still lacking. Additionally, while the Big Data era opens new prospects for enabling service intelligence, it also proposes challenges to process mining [19, 31]. The integration of high volume data from various sources complicates the operation of process mining. Therefore, in this work, we tackle these challenges and aim to:

- Simplify the procedure of automated process modelling. Instead of using a mass of redundant event logs as input data, we aim to collect the minimum amount of process data that is required in the modelling of a process.

- Ensure the quality of mined models, especially model completeness. The input data used for the process modelling should be precise and cover all possible ways of executing a process.

- Develop approaches for modelling more generic services than domain-specific service processes.

Quite often, activities in the generic services have the property of being location-aware, which means that a particular activity happens in a specific location. Thus, location information can be used to infer the happenings of activities [34].

The development of inexpensive and unobtrusive sensors has enabled real-time activity recognition and automated process modelling [21, 24, 32]. Different wireless indoor localization technologies offer various levels of cost, accuracy and applicability. These include technologies based on radio-frequency (RF) [2, 20], WiFi networks [12, 14, 28, 29, 33], ambience fingerprinting using the camera and microphone of a mobile phone [1], and using the accelerometer and compass of

a phone [7]. Many of these are based on employing hardware that is built into a smartphone.

When modelling location-based service processes, the accuracy of localization is not the primary requirement. Therefore, we decided to use an easy and inexpensive solution that is accurate enough for our purposes: indoor localization based on Bluetooth sensors and the Bluetooth hardware of smartphones. Bluetooth-based localization is low-cost, highly ubiquitous, very easy to implement and deploy, and has low power consumption and allows ad-hoc connections with room-wise accuracy [8, 13].

Based on this inexpensive indoor localization, this article presents an approach for automated process modelling. Our light-weight process data acquisition system collects the minimum amount of service process data needed for process modelling. This article describes our approach for automatically mining a process model. The obtained process model could then be used by tools such as 3VPM [15] to analyse and optimise the process performance.

The rest of the paper is organised as follows: Section 2 introduces the conceptual framework of the service process model. Section 3 illustrates the automated process modelling system. Section 4 discusses the analytical approach used in process model extraction. We describe how the system was evaluated in a laboratory case study in Section 5, and in a dental care clinic in Section 6. Section 7 concludes the paper with the limitations of the current system and directions for future work.

2   Conceptual Framework of Service Process Model

This section discusses the conceptual framework of the service process model that is used in this study: the concepts of service process and service process model, and block-structured model representation. Besides, this section defines generic service processes as the focus of this study, and distinguishes them from domain-specific processes.

2.1   Process Model Representation

A service process shows how a particular service is carried out, it pictures all the possible ways to execute a service process. A service process includes a sequence of activities, and each activity is a major unit of work. By performing the sequence of activities, the service process outputs a service as a result.

The process model representation used in this study is based on a block-structured model that uses boxes and arrows. One box denotes an activity, whose occurrence usually lasts a certain service time. Arrows between activities define the partial order of activities. If there are several arrows leaving a box, then each arrow is attached by the probability of transition from the current box to the box pointed by the arrow. The sum of the probabilities of arrows leaving one box is 1.

The block-structured models can be represented in form of trees [35], or terms, or diagrams. Block-structured models have advantages of not containing any deadlocks or other anomalies, moreover, the models captured are well-formed and always sound [27]. Figure 1 illustrates the process model representation in the form of a diagram, which describes an example of a healthcare service. In the example, we use oval-shaped boxes to represent activities.
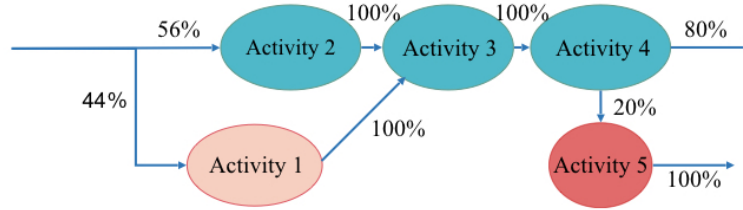

Fig. 1: Example of a service process model in the form of a diagram

In general, there are three basic types of transitions between activities:

– A sequential transition indicates that activities are performed in a sequential order. For example, three sequential activities in Figure 1: Activity 2, Activity 3, Activity 4. A single arrow between two activities a_1 and a_2 means that a_1 must precede a_2 and a_2 follows a_1 with probability 1.
– An alternative transition indicates the occurrence of exactly one activity out of all alternative activities at a time. In the example model, Activity 2 and Activity 1 are alternative activities, which means for a particular execution of the service, the process will proceed with either Activity 2 or Activity 1.
– A parallel transition means that the activities can be performed at the same time in parallel.

Sequential and alternative transitions are used in this research to model the processes of a single actor. Parallel transitions are applied in multi-actors collaboration processes, in which activities can be performed by different team actors in parallel. Our process models inherently include parallelism when several persons or customers are moving between the activities, which is similar to queuing networks. In the automatic measurement, we do not obtain more information on activity ordering than transaction probabilities.

2.2   Generic Service Process and Domain-specific Process

In the healthcare sector, most research so far has focused on mining domain-specific processes. Here domain-specific refers to treatment-specific, such as a

laparoscopic surgery process [5] or a gynecological oncology process [25]. Such a process can be extracted from event logs of existing medical treatment systems. However, there are more generic activities in the service process that have no trails in any administrative system or medical device. This gap requires approaches that are capable of picturing more generic service processes.

According to the process definition given by Davenport [9], treatment-specific processes can be understood as small processes, and generic service processes are larger processes that serve an entire set of service activities. Davenport reckons that larger processes have greater potential for radical benefits, and a key aspect of process innovation is to focus on broad processes. Generic activities are often location-aware, in the sense that a particular activity is performed at a specific location. Therefore, we abstracted generic processes to location-based ones. An indoor positioning technique such as Bluetooth was considered. Bluetooth is a mature research field and has been widely studied [3, 22]. This study integrates Bluetooth indoor positioning and Internet of Things (IoT) to collect service activity data and construct service process models.

3   Automated Process Modelling System

In the previous sections, we discussed the target process group of this study and introduced the process model representation applied in it. In this section, we present an automated process modelling system, Figure 2 illustrates our approach, which has 4 steps:

- *Step one: planning.* First, determine targeted process activities and locations where these activities are performed. Then place Bluetooth sensors in these activity locations.

- *Step two: process calibration.* For process p, train a process-specific subset of Bluetooth sensors $S(p)$ on the mobile device-side. Meanwhile, it also collects training data and transfers it to the server-side to compute activity patterns and other parameters, which are used in process measurement for activity recognition.

- *Step three: process measurement.* Collect process data and synchronise them to the server continuously for activity recognition.

- *Step four: process modelling.* Based on the information of recognised activities, model the entire process on the server-side.

3.1   Input Data of the Process Modelling System

Bluetooth data is collected to analyse indoor location information, and furthermore be used to infer location-based activities. The system requires information of the activities that compose the process, which is defined as follows:
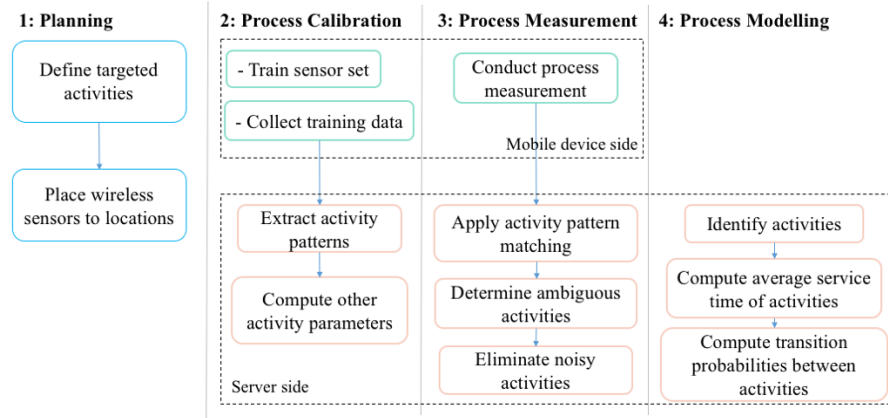
Fig. 2: The approach used for automated process modelling

Definition 1. *Let $a = (id, name, location, S_a \subseteq S)$ denote the basic input information of an activity: an activity identifier, a name, a location where it is performed, and a set of Bluetooth sensors that are placed to the location. Then for a service process p with n activities, $A(p) = \{a_1, \ldots, a_n\}$ denotes the set of activities in the process.*

In addition, the system also requires information of the Bluetooth devices that are involved in the process measurement. Let $D = S \cup U$ be a set of Bluetooth devices that includes Bluetooth sensors (denoted by $S = \{s_1, \ldots, s_q\}$) and mobile devices (denoted by $U = \{u_1, \ldots, u_u\}$). The minimum information that is required to identify a Bluetooth sensor or a mobile device is the unique Bluetooth MAC address.

3.2 Calibration of the Process Measurement

In order to position activity locations, we collect Bluetooth Radio Signal Strength Indications(RSSI) data, which can be used to infer the location information of activities. However, the radio propagation of Bluetooth sensor is extremely complex and unstable. We compared the performance of Bluetooth sensors by measuring their RSSI values from the same distance away, but the radio signal strengths obtained varied dramatically. As a result the RSSI values cannot be used directly, we had to include an essential step: *calibration* and introduce an analytical approach that converts RSSI values at each time stamp into a numeric vector.

Figure 3 demonstrates three independent calibrations for different processes. For a particular process p_i, we train a process-specific subset of Bluetooth sensors

$S(p_i)$ from the Bluetooth sensor set S, $S(p_i) \subseteq S$. Different processes can consist of activities that are performed at the same location, which also means one sensor placed in one location can be involved in the measurements of multiple processes. For example, in Figure 3, sensor s_{10} and sensor s_{13} are used in the measurements of Process A and Process B, sensor s_7 is involved in the measurements of Process B and Process C.
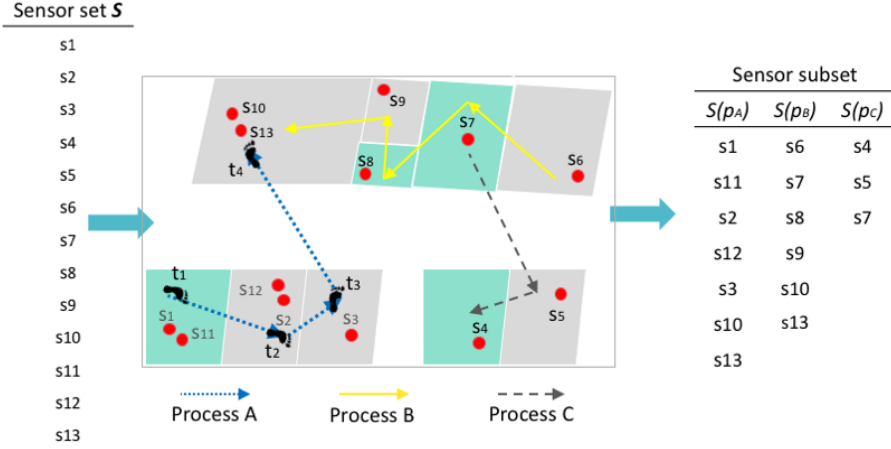

Fig. 3: Three calibration operations train different subsets of Bluetooth sensors ($S(p_A)$, $S(p_B)$, $S(p_C)$) for different processes (p_A, p_B, p_C).

During this, we also collect a training data set $E' = \{e'_1, \ldots, e'_t\}$ with a sampling rate f, so that the timestamp of a record is $t = 0 + n \cdot f$, $n \in \mathbb{N}$. Training data is defined as follows.

Definition 2. *At time point t, a calibration event record for mobile device u is $e' = (t, R_t)$, where R_t is a set containing one RSSI value r_s for each sensor $s \in S$. Here $r_s = 0$, if sensor s is not in the radio detecting range of u, otherwise, r_s equals real-time RSSI value of the sensor.*

Figure 4 shows the example of training data that is collected from the calibration of Process A. In the first event tuple, at time point t_1, RSSI values of sensor s_1 and s_{11} are -89 and -85, other sensors are out of range, hence RSSIs are 0.

In addition, calibration synchronises the device's local time with the remote server time, to maintain the timestamps of records that are collected from different devices consistent. This is an essential step for multi-user collaborative activity recognition. In practice, calibration only requires an administrator to carry all the user devices in a basket and walk through all the locations once, letting the devices measure a few data points of RSSI information at each location. Calibration is subject-independent, it keeps general users away from the

Time	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13
t1	-89	0	0	0	0	0	0	0	0	0	-85	0	0
t2	0	-79	0	0	0	0	0	0	0	0	0	-80	0
t3	0	0	-95	0	0	0	0	0	0	0	0	0	0
t4	0	0	0	0	0	0	0	0	0	-75	0	0	-77

Fig. 4: Example of Process A's calibration event record: $e' = (t, R_t)$. At time point t_1, calibrate the sensors (s_1, s_{11}) of activity a_1, and at t_2, calibrate the sensors (s_2, s_{12}) of activity a_2, and so on.

burden of the training phase, and provides users with a ready-to-use application, which will facilitate the application of this system.

3.3 Process Measurement

Process measurement collects process data $E = \{e_1, \ldots, e_t, \ldots\}$ and synchronises them to the server continuously for activity recognition. With the objectives of simplifying the procedure of automated process modelling, we only collect the minimum amount of data that is required to model a process: activities, location of activities, activity begin and end time, and the actor who performs the process.

This work studies processes performed by a single user, instead of analysing collaborative processes that involve multiple users. Hence, a process measured is user-specific, and the measurement record has an additional user identifier in comparison with calibration records. We use Bluetooth MAC address of a mobile device as the user identifier, and define measurement data as follows:

Definition 3. *For process p, a measurement event record collected for user u_i at time point t is $e = (t, u_i, R(p)_t)$, where $R(p)_t$ is a set containing one RSSI value r_s for each sensor $s \in S(p)$. Here $r_s = 0$, if sensor s is not in the radio detecting range of u_i, otherwise r_s equals real-time RSSI value of the sensor.*

Figure 5 shows the example of measurement data that is collected from the measurement of Process A. Measurement data is similar to training data in Figure 4. Besides the additional user identifier, measurement records are also process-specific, which means it only measures the RSSI values $R(p)$ of process related Bluetooth sensors $S(p)$.

When a particular process p is selected to be measured, information about the corresponding set of Bluetooth sensors $S(p)$ will be synchronised from the remote server. Subsequently, a background service is started and it applies asynchronous *Broadcast Receiver* schema to detect RSSI vectors periodically. The asynchronous broadcast receiver schema is basically a broadcast receiver that keeps listening to two actions: *Action one*, a remote Bluetooth sensor found; *Action two*, one Bluetooth inquiry finished. Action one is triggered when the mobile

Time	User	s1	s11	s2	s12	s3	s10	s13
t1	u1	-66	-65	-95	0	0	0	0
t1	u2	0	0	0	0	-68	0	0
t2	u1	0	-93	-75	-93	-89	0	0
t2	u2	0	0	0	0	0	-69	-73
t3	u1	0	0	0	0	-78	0	0
t3	u2	-87	-85	-73	-69	0	0	0
t4	u1	0	0	0	0	0	-71	-74
t4	u2	-97	-94	-77	-71	-89	0	0

Fig. 5: Measurement data samples of Process A: $e = (t, u_i, R(p_A)_t)$. At time point t_1, activity a_1 is performed, RSSI values of sensors (s_1, s_{11}) are received, due to the signal overlapping, we also receive the RSSI value of sensor s_2 that is placed in another location.

device enters the radio proximity of a fixed Bluetooth sensor, in the meantime, the system collects real-time Bluetooth RSSIs. Action Two is triggered when one Bluetooth inquiry duration ends, which is about 12 seconds and is the same as the system's sampling rate. Thereafter, a new Bluetooth discovery will start.

Upon this architecture, the integration of IoT enables automated process modelling: the system collects tuples continuously, meanwhile, the mobile device synchronises tuples to the remote server through Wi-Fi periodically. The synchronising rate is adjustable for different measurement needs.

The system applies Google *Volley* networking framework[1] to stream data between the server and the mobile devices. Server side applies activity pattern matching and sensor performance parameters to recognise explicit activities as well as ambiguous activities. Ultimately, window size is used over incoming tuples to eliminate noisy activity detections.

4 Process Model Extraction

In this section, we discuss the analytical approach used to identify process activities and extract process models. To simplify the following discussion, we assume that there is only one person whose activities are being measured.

4.1 Computing Activity Patterns and Other Parameters

During the calibration, the mobile device executes sensor training $S(p)$ and collects training data E'. Mobile devices are taken to walk through all the locations of activities and to measure a few data tuples at each location. The calibration process is based on activity information $A(p)$ defined in section 3.1, and each training tuple is labeled according to $A(p)$.

---

[1] https://developer.android.com/training/volley/index.html

Then, the server side's analytical process uses training tuples to compute activity patterns and other activity related parameters. Figure 6 illustrates the activity pattern extraction process. Pattern matching approaches for activity recognition were used in other research as well: Faragher and Harle [10] that used fingerprint techniques, and Martikainen [16]. The activity pattern used in this study is defined as follows.
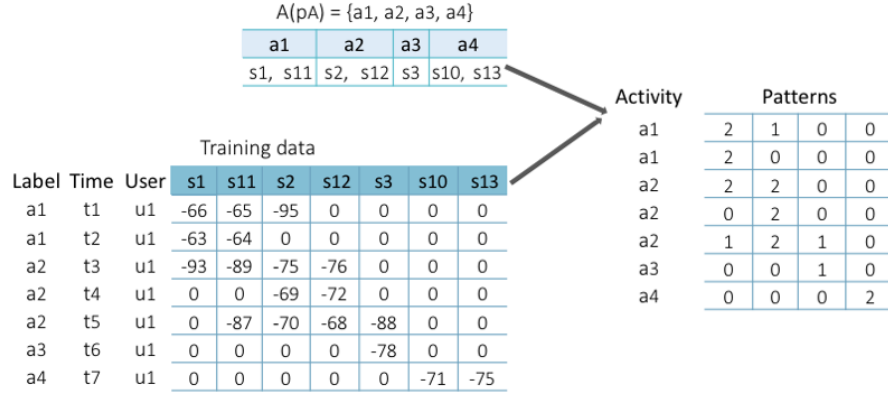

Fig. 6: Computing Activity Patterns from Calibration Data. At time point t_1, mobile device receives RSSI values from two sensors (s_1, s_{11}) that are placed in activity a_1, as well as one sensor (s_2) that are placed in activity a_2. No signal received from activity a_3 and a_4. According to the definition of a activity pattern, pattern here is [2, 1, 0, 0].

Definition 4. *For an activity a, we define an activity pattern $m(a)$ as the set containing one element $k_{a'}$ for each $a' \in A(p)$. Here $k_{a'}$ denotes the number of Bluetooth sensors at the location of activity a' that are within the radio detection range of the measurement device when performing activity a. Then the set of activity patterns of the process p is $M(p) = \{m(a)|a \in A(p)\}$.*

Figure 6 also shows activity pattern examples. The activity pattern $m(a_1) = [2, 0, 0, 0]$ is an ideal case: when performing the activity a_1, only the two Bluetooth sensors at the location of activity a_1 are within the mobile device's radio detection range. Sensors at locations of other activities are not in the detecting range, denoted by 0. However, in practice there are often overlapping signals. For example, the activity pattern $m(a_2) = [1, 2, 1, 0]$ means that while we are performing activity a_2, besides the two sensors of activity a_2, one sensor at the location of activity a_1 and one sensor at the location of activity a_3 are in the detection range as well.

In addition, for each activity, all training data that is labeled with this activity is used to compute other parameters like the activity-specific RSSI mean, average performance and tolerance range.

4.2 Analytical Approach to Extract Process Model

After we obtain the set of activity patterns $M(p)$ from the calibration, measurement can be started. During the measurement, the server's analytical process applies activity pattern matching on each measurement tuple that is transferred from the mobile devices. Each measurement tuple will be firstly converted into an activity pattern m_i that is defined in Section 4.1, then be compared with $M(p)$ to determine the activity. For those ambiguous tuples that match more than one activity class, extra parameters computed in the calibration are used to determine the best matching activity class for the tuple.

When measurement is finished, we will get a list of determined activities A_D and further analysis can be conducted (as shown in Figure 7) to extract process models. Firstly, the server applies a window over a few successive activities to get the most frequently occurring activity. This eliminates noisy activity detections and gets the list of calibrated activities A_C. Very often noisy activities are recognised when switching to a new activity momentarily, and back to the original activity in the next data point, for example the determined activity 3 at time point t_7 in Figure 7. Here, we assume this type of noise contains less than two tuples. Secondly, based on calibrated activity list A_C, we determine the beginning and ending of the activities and output changes of activities A_B: 1 or 2 in this example indicate the beginning of activity 1 and activity 2, 0 means no change, -1 indicates no activities or constant change between activities.

i	Time	Determined Activity $A_D[i]$		Calibrated Activity $A_C[i]$		Activity begin & end $A_B[i]$	Service Time
1	t1	1		1		1	
2	t2	1		1		0	t2−t1
3	t3	2		2	<2>	2	
4	t4	2		2		0	
5	t6	2		2		0	t8 - t3
6	t7	3	<1>	2		0	
7	t8	2		2		0	

Fig. 7: Analytical approach for process model extraction. Step <1>: get the most frequently occurring activity in the three successive activities. Step <2>: determine the beginning and ending of the activities.

At last, we computed average service times and transition probabilities to model the process. For an activity $i \in \{1, \ldots, n\}$ that occurred m_i times in the

data, the average service time is:

$$S_i = 1/m_i \sum_{j=1}^{m_i} d_{i,j} \tag{1}$$

where $d_{i,j}$ is the duration of the jth occurrence of activity i. We then compute a matrix of how many (directed) transitions occurred between the activities: $T_{i,j}$ = number of transitions from activity i to activity j. From this we can calculate transition probabilities by scaling them with the total number of outgoing transitions from an activity. That is, the transition probability $P_{i,j}$ from i to j is:

$$P_{i,j} = T_{i,j} / \sum_{k=1}^{n} T_{i,k} \tag{2}$$

4.3   Accuracy

The accuracy of the extracted process depends on the length of the measurement (amount of data available), as well as the accuracy of the indoor localization method. As mentioned, we are interested in location-based service processes where the locations are not very close together (typically in different rooms) and the service times are reasonably large (several minutes or more).

Our inexpensive localization method is limited to discrete locations at least about 2 meters apart, and the service times can have an error of up to 24 seconds (both entry and exit from a location can be detected up to 24 seconds too late, as mentioned above). In practice this localization method appears to be fairly good at detecting every activity (if they are minutes long) and not detecting any spurious wrong activities; thus usually it does not cause errors in the transition probabilities.

The estimated service time for each activity is simply the average of all measured service times (including the up to 24-second error). If there is a lot of variation in the actual service times, our process model hides this and the average may not be representative. However, it would also be possible to examine the distribution of the measured service times at each activity.

The amount of variation in the actual process also affects the estimated transition probabilities, especially if the measurement is short. For instance, rare transitions (say, a special case of a service process that is encountered rarely) might not be observed at all in a short measurement. However, this might not be important if the purpose of process extraction is to analyse or optimise the common cases. The most important factor in this accuracy is how many times each transition was observed in the data.

5   Laboratory Case Study

We implemented the data acquisition system for Android smartphones, and evaluated the system and the analytical approach for process model extraction in a

laboratory case study. We put 17 Bluetooth sensors in 8 locations in the computer science building in Aalto University to represent 9 activities. The Bluetooth sensors used in this study were built with JY-MCU Bluetooth wireless serial port modules[2].

Figure 8 shows the placement of the sensors in the process measurement. As discussed in previous sections, the performances of sensors vary and they are neither stable nor consistent. In order to find the optimal placement of the sensors, we conducted several experiments and found out that, the usage of two sensors to represent one activity helps improve the process measurement results. In the case study, we wrote down the actual process on paper by hand so that we could compare results. The process measurement results are presented in Figure 9a and Figure 9b.
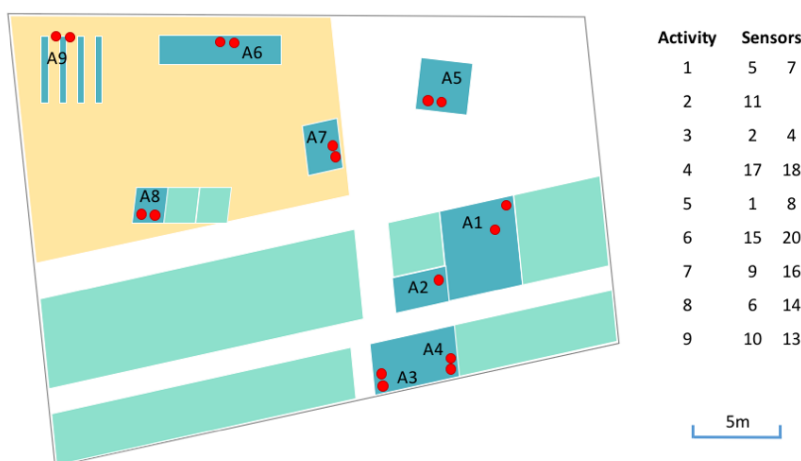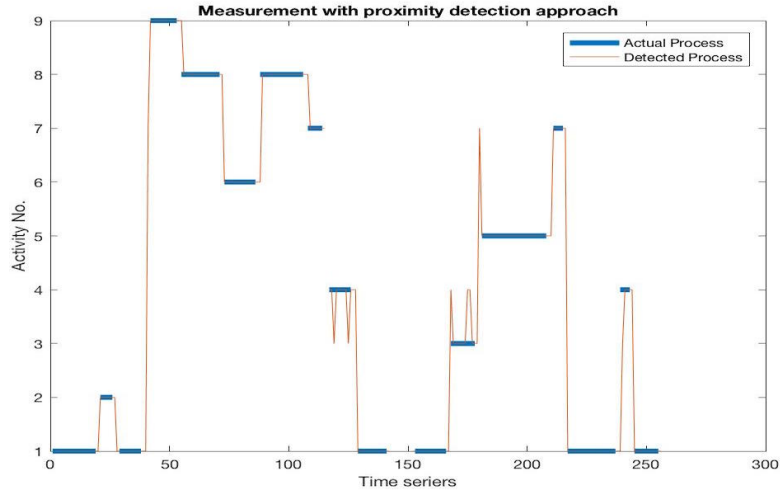

Fig. 8: Case study floor plan: sensor placement for process measurement

Figure 9a shows the results of using a proximity detection approach and without calibration. It has two problems: first, when the locations of two activities are relatively close to each other, this approach will lead to noisy fluctuations; second, when there is only a very short interval between two activities, it won't be accurate enough to determine the interval. Figure 9b demonstrates the application of the analytical approach illustrated in Section 4.

By comparing with the actual process, the result shows that the analytical approach for process model extraction detected the correct activity in 93% of the data points. It other words, the system fulfils the demands of collecting precise process data for accurate process modelling.

---

[2] https://core-electronics.com.au/attachments/guides/Product-User-Guide-JY-MCU-Bluetooth-UART-R1-0.pdf

(a) Activity recognition results without calibration. Generated activity sequence example is: 11111 22222 11111 99999 8888, which has no intervals between activities.
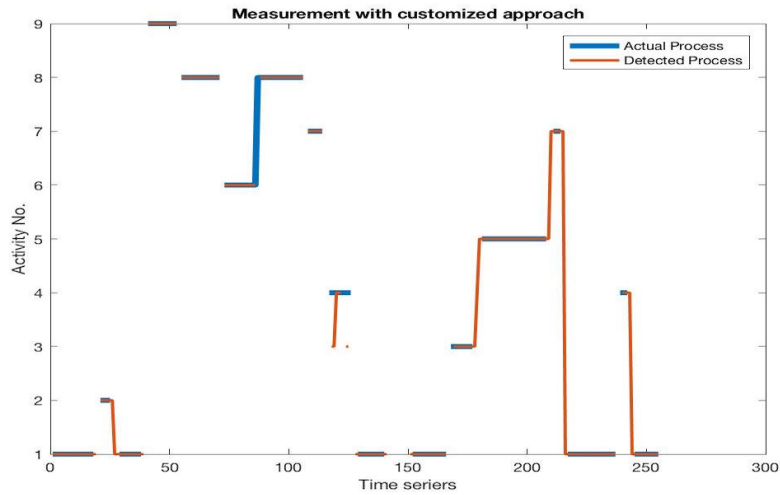
(b) Activity recognition results with calibration. Generated activity sequence example is: 11110 22222 11110 99990 8888, which recognises the intervals between activities.

Fig. 9: Evaluation of automated process measurement results. Actual activity sequence in this example is: 11100 22200 11100 99990 8888.

Figure 10 presents the process model captured from the case study. The average service times and transition probabilities are calculated from the analysed data (the beginning and end of each occurrence of an activity) with Equation (1).
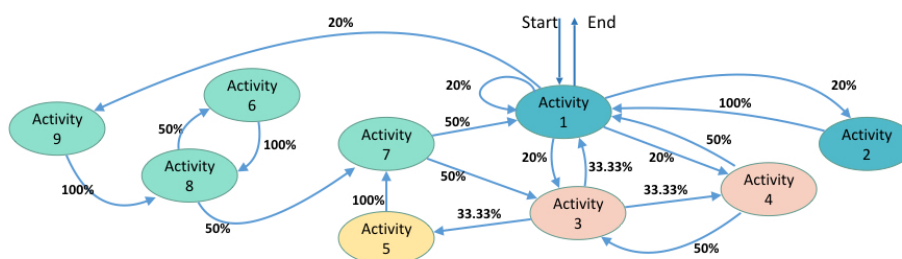

Fig. 10: Model of case study's process, which is extracted by using the automated process modelling system

6   Empirical Attestation

In a children's dental clinic (YoungTeeth) in Tampere, Finland, we evaluated an earlier version of our system, which did not send the location data to a centralised server: instead we needed to collect the data manually from each mobile device. However, the analysis of the data to extract the process model was essentially the same as given in Section 4. The dental care service process included patients (children and adolescents) and four professional groups (dentists, dental hygienists, dental nurses, and receptionists). Figure 11 presents the original service model generated by the data collected from the patients, and the model generated by the data from the personnel differs in that the process loops to the start point. "Hygiene check" is the oral health check performed by a hygienist, "Dentist" is the diagnosis made by a dentist, "Wait" is the time the patient needs to wait at reception, and "Recall" means the calling system that invite children to the clinic.

After the original process model was created and calculated, possible improvements were analysed. In the original service process, patients visit hygiene check, if health issues like tooth decay are discovered, then patients have to visit the dentist twice for diagnostics and treatment. In order for patients to get an oral health diagnosis and treatment plan during one visit, a multi-room dental service model were proposed. The general idea of the optimised improvement is that dental hygienists consult dentists if special expertise is needed. This optimised process (see Fig. 12) was piloted 10 days in the same clinic to evaluate the improvements. Patients who do not need to consult the dentist visited Hygiene check 3, and others follow Dentist consultation procedure: hygienist performs
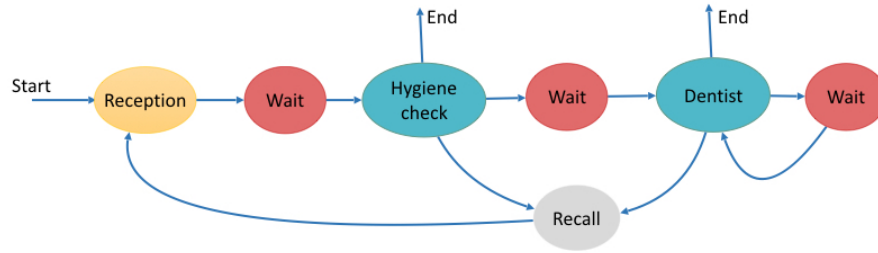
Fig. 11: The original service process of a children's dental clinic (YoungTeeth) in Tampere Finland. This model is generated from the data collected from the patients.

the oral health check (Hygiene check 1), the dentist is called if needed (Dentist), then hygienist finish the care (Hygiene check 2).
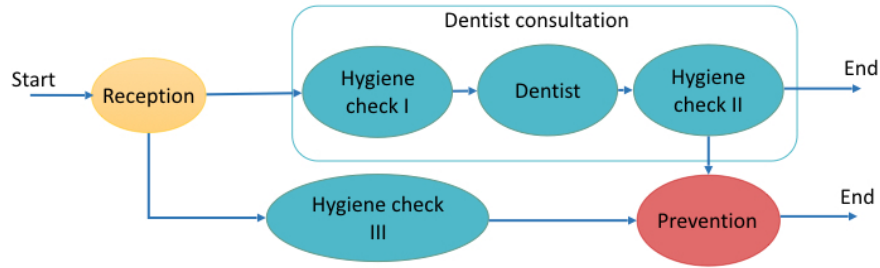

Fig. 12: The optimised process improvement proposed for a children's dental clinic (YoungTeeth) in Tampere Finland.

The optimised process was measured and data was collected and analysed for future use. In this case, we captured a stable service model from five complete measurements of the whole process. However, the required measurement times in different service processes vary with the complexity of the process (approximately 5-50). According to the analysis, the process performance was increased 24%.

The process model that was extracted in this case study also allowed a measurement of the maximum load of the professionals (work activity time divided by total time per person). In this case the load did not exceed 75%, which has been considered critical in other related research cases. Furthermore, this model was used as a basis of process measurement analysis in [11].

7   Conclusions

Process modelling is a critical factor in improving service productivity improvement and implementing service intelligence, and it is inevitable that we should consider service innovation from the process model viewpoint. However, how to collect data from services has been a problem: traditional approaches that construct process models from scratch often requires field experts' involvement, which is difficult, expensive and time-consuming. An alternative way that mines process models from event logs is suitable for modelling domain-specific workflows, but approaches that are capable of picturing more generic processes are lacking.

This work aims to simplify the procedure of automated process modelling and focuses on modelling generic service processes that are location-aware. We proposed an approach based on wireless indoor positioning, which collects the minimum amount of location based process data and models service processes. Bluetooth technique was selected because it is inexpensive, easy to implement and accurate enough for measuring service processes. The automated process modelling system has a mobile and a server side, the process data acquisition module of the mobile side is responsible for calibrating the process measurement and collecting real-time process event data. The analytical module of the server analyses calibration data, computes activity patterns and sensor performance parameters, and extracts service process models from acquired measurement data. Instead of using unstable RSSI values directly, our approach uses numeric vector of activity patterns that is converted from RSSI values. The approach is accurate enough for current objectives. In further research, we will study more accurate positioning techniques and explore whether very precise locations help improve location-based activity recognition.

In the case study, we examined the performance of the process data acquisition system and the analytical approach. The results of the case study demonstrates that the system fulfils the demand of collecting the minimum amount of process data for accurate process modelling. In addition, the presented approach has been tested and used in a children's dental care clinic in Tampere, Finland. This confirmed the feasibility of the approach for process modelling, and the extracted models were used in process performance optimisation.

Application status of the current system is limited to relatively ideal settings: one location represents only one activity. Besides, the system requires that two locations have a certain distance (minimum 2 meters) away from each other. As illustrated in our analytical approach, we eliminate noisy activities that have less than two tuples, hence, the shortest activity that can be detected has to have at least two tuples (about 24 seconds). However, this does not limit the application of our approach in service process measurement since activities of service process usually last more than 24 seconds. The current system is applicable for analysing the process of a single user, but does not have collaborative process modelling support yet. Therefore, our objective for future research is to implement automated process modelling for a team collaboration process, and

moreover, to improve the accuracy of process activity recognition with the help of additional data, for example, accelerometer data.

References

1. Azizyan, M., Constandache, I., Roy Choudhury, R.: SurroundSense: mobile phone localization via ambience fingerprinting. In Proceedings of the 15th annual international conference on Mobile computing and networking, ACM, 261-272 (2009)
2. Bahl, P., Padmanabhan, V. N.: RADAR: An in-building RF-based user location and tracking system. In INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings, Vol. 2, 775–784 (2000)
3. Baniukevic, A., Jensen, C. S., Lu, H.: Hybrid indoor positioning with wi-fi and bluetooth: Architecture and performance. In 2013 IEEE 14th International Conference on Mobile Data Management, Vol. 1, 207–216 (2013)
4. Basu, S., Pascali, L., Schiantarelli, F., Serven, L.: Productivity and the Welfare of Nations. NBER Working Paper No. 17971, 1-68 (2012)
5. Blum, T., Padoy, N., Feuβner, H., Navab, N.: Workflow mining for visualization and analysis of surgeries. In International Journal of Computer Assisted Radiology and Surgery, 3(5), 379–386 (2008)
6. Bose, R. J. C., Mans, R. S., van der Aalst, W. M.: Wanna improve process mining results? In Computational Intelligence and Data Mining (CIDM), 2013 IEEE Symposium on. IEEE, 127–134 (2013)
7. Constandache, I., Choudhury, R. R., Rhee, I.: Towards mobile phone localization without war-driving. In Infocom, 2010 proceedings IEEE 1-9 (2010)
8. Dardari, D., Closas, P., Djuri$\acute{c}$, P. M.: Indoor tracking: Theory, methods, and technologies. IEEE Transactions on Vehicular Technology, 64(4), 1263–1278 (2015)
9. Davenport, T. H.: Process innovation: reengineering work through information technology. Harvard Business Press, ISO 690 (2015)
10. Faragher, R., Harle, R.: Location fingerprinting with bluetooth low energy beacons. IEEE Journal on Selected Areas in Communications, 33(11), 2418–2428 (2015)
11. Halonen, R., Martikainen, O., Juntunen, K., Naumov, V.: Seeking efficiency and productivity in health care. In 20th Americas Conference on Information Systems. AMCIS-0251-2014.R1. (2014)
12. Lim, H., Kung, L. C., Hou, J. C., Luo, H.: Zero-configuration, robust indoor localization: Theory and experimentation. In Infocom (2006)
13. Liu, H., Darabi, H., Banerjee, P., Liu, J.: Survey of wireless indoor positioning techniques and systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37(6), 1067–1080 (2007)
14. Liu, H., Gan, Y., Yang, J., Sidhom, S., Wang, Y., Chen, Y., Ye, F.: Push the limit of WiFi based localization for smartphones. In Proceedings of the 18th annual international conference on Mobile computing and networking, ACM, 305-316 (2012)
15. Martikainen, O., Halonen, R.: Model for the Benefit Analysis of ICT. In 17th Americas Conference on Information Systems. AMCIS, 4-7 (2011)
16. Martikainen, O.: A method and a computer program product for controlling the execution of at least one application on or for a mobile electronic device, and a computer. Patent, EP2758874 (2011)

17. Mans, R. S., Schonenberg, M. H., Song, M., van der Aalst, W. M., Bakker, P. J.: Application of process mining in healthcarea case study in a dutch hospital. In International Joint Conference on Biomedical Engineering Systems and Technologies. Springer Berlin Heidelberg, 425–438 (2008)
18. Mans, R. S., van der Aalst, W. M., Vanwersch, R. J., Moleman, A. J.: Process mining in healthcare: Data challenges when answering frequently posed questions. In Process Support and Knowledge Representation in Health Care. Springer Berlin Heidelberg, 140–153 (2013)
19. Meng, S., Dou, W., Zhang, X., Chen, J.: Kasr: A keyword-aware service recommendation method on mapreduce for big data applications. IEEE Transactions on Parallel and Distributed Systems. 25(12), 3221-32 (2014)
20. Ni, L. M., Liu, Y., Lau, Y. C., Patil, A. P.: LANDMARC: indoor location sensing using active RFID. Wireless networks, 10(6), 701-710 (2004)
21. Okeyo, G., Chen, L., Wang, H., Sterritt, R.: Dynamic sensor data segmentation for real-time knowledge-driven activity recognition. Pervasive and Mobile Computing, 10, 155-172 (2014)
22. Palumbo, F., Barsocchi, P., Chessa, S., Augusto, J. C.: A stigmergic approach to indoor localization using bluetooth low energy beacons. Advanced Video and Signal Based Surveillance (AVSS), 12th IEEE International Conference on. IEEE, 1–6(2015)
23. Partington, A., Wynn, M., Suriadi, S., Ouyang, C., Karnon, J.: Process mining for clinical processes: a comparative analysis of four Australian hospitals. ACM Transactions on Management Information Systems (TMIS), 5(4), 19 (2015)
24. Pham, C., Pltz, T., Olivier, P.: A dynamic time warping approach to real-time activity recognition for food preparation. In International Joint Conference on Ambient Intelligence, Springer Berlin Heidelberg, 21-30 (2010)
25. Rebuge, Á., Ferreira, D. R.: Business process analysis in healthcare environments: A methodology based on process mining. Information Systems, 37(2), 99–116 (2012)
26. Rovani, M., Maggi, F. M., de Leoni, M., van der Aalst, W. M.: Declarative process mining in healthcare. Expert Systems with Applications, 42(23), 9236–9251 (2015)
27. Schimm, G.: Mining most specific workflow models from event-based data. International Conference on Business Process Management. Springer Berlin Heidelberg, 25-40 (2003)
28. Sen, S., Lee, J., Kim, K. H., Congdon, P.: Avoiding multipath to revive inbuilding WiFi localization. In Proceeding of the 11th annual international conference on Mobile systems, applications, and services, ACM, 249-262 (2013)
29. Sen, S., Radunovic, B., Choudhury, R. R., Minka, T.: You are facing the Mona Lisa: spot localization using PHY layer information. In Proceedings of the 10th international conference on Mobile systems, applications, and services, ACM, 183-196 (2012)
30. Solow, R.M.: Technical change and the aggregate production function. The review of Economics and Statistics, 312-320 (1957)
31. Vera-Baquero, A., Colomo-Palacios, R., Molloy, O.: Business process analytics using a big data approach. IT Professional, 15(6), 29–35 (2013)
32. Wan, J., OGrady, M. J., OHare, G. M.: Dynamic sensor event segmentation for real-time activity recognition in a smart home context. Personal and Ubiquitous Computing, 19(2), 287–301 (2015)
33. Youssef, M., Agrawala, A.: The Horus WLAN location determination system. In Proceedings of the 3rd international conference on Mobile systems, applications, and services, ACM, 205-218 (2005)

34. Zhang, Y., Martikainen, O., Pulli, P., Naumov, V.: Real-time process data acqui-
    sition with Bluetooth. In Proceedings of the 4th International Symposium on Ap-
    plied Sciences in Biomedical and Communication Technologies, Barcelona, Spain,
    Vol. 2629 (2011)
35. Zhang, L. L., Rodrigues, B.: A tree unification approach to constructing generic
    processes. IIE Transactions, 41(10), 916-929 (2009)