Convenience-Based Periodic Composition of IoT Services

Bing Huang(B), Athman Bouguettaya, and Azadeh Ghari Neiat
School of Information Technologies, The University of Sydney, Sydney, Australia

Summarized by: Siyu Yang

1. Problem Statement

This paper focus on providing convenience by discovering periodic composite IoT services. They propose a novel framework for the transient composition of spatio-temporal IoT service to suit the smart home occupant's convenience needs. They design a significance model is proposed to prune insignificant composite IoT services and describe a spatio-temporal proximity technique to prune loosely correlated composite IoT services. A periodic composite IoT service model is proposed to model the regularity of composite IoT services occurring at a certain location in a given time interval.

2. Motivation

An application domain for IoT is the smart home. To empower such high-level intelligence in smart homes, a key task is to discover periodic composition of IoT services which can represent periodic human activities. It is important to discover periodic composite IoT services. Periodic composition of IoT services can provide an insightful and concise explanation of IoT service usage patterns. These patterns can be used to design intelligent control of IoT services in smart homes to reduce residents' interactions with IoT services. Reducing those interactions provides more convenience for residents. Such periodic composite IoT services are also useful for human activity prediction.

There are three key challenges:

- The set of IoT services are not known. We refer such set of IoT services as composite IoT services.
- Many of the spatio-temporal relationships may be insignificant and loosely correlated so we need to prune insignificant and loosely correlated composite IoT services.
- The associated time interval and location for the periodic composite IoT services are not known. Therefore, there is a need to estimate the associated time interval and location.

The discovered periodic composite IoT services can be applied to build an intelligent system for providing convenience.

3. Overview of the Proposed Solution

A. Discovering composite IoT services.

Definition 1: IoT Service

The representation of an IoT service S_i is simplified as $< (s_i^+, st_i, sl_i), (s_i^-, et_i, el_i) >$. For example, a light service like $< \left(light^+, 7pm, (1,2)\right), \left(light^-, 9pm, (1,2)\right) >$ which is described as lighting from 7pm to 9pm in the bedroom where (1,2) is the GPS point in the bedroom and 7pm (resp. 9pm) is the start time (resp. end time).

Definition 2: Composite IoT Service

A composite IoT service CS is a collection of IoT services, $CS = < S, sup(S) >$ where $S = \{< (s_1^+, st_1, sl_1), (s_1^-, et_1, el_1) >, \ldots, < (s_n^+, st_n, sl_n), (s_n^-, et_n, el_n) >\}$ represents n component IoT services, $sup(S) = |\{(sid, S) \in DB | S \sqsubseteq S'\}|$ is the support of S (*sid* is a sequence ID and $S'$is the composite IoT service). *sup(S)* is the number of tuples containing *S*.

B. Employ significance and proximity to prune composite IoT services.

Definition 3: Significance

Significance is used for evaluating statistic importance of *CS*. According to Bernoulli distribution, the occurrence expect(S) is:

$$significance(S) = \frac{\sqrt{expect(S)}}{sup(S) - expect(S)}; \ \ expect(S) = P(S) \cdot |DB_{ri}| = \prod_{\forall s_i^+ \in \text{Seq}} P(s_i^*)$$

where$|DB_{ri}|$ is number of IoT service events in region r_i.

Definition 4: Proximity

Given a composite IoT service $CS = < S, sup(S) >$, its proximity function is defined as: $U = w1 \cdot spatial\ proximity + w2 \cdot temporal\ proximity$, where w_i is a weight such that $w_i \in [0,1]$ and $w_1 + w_2 = 1$.

The spatial proximity measures the average location proximity of all composite IoT service instances. *spatial_proximity* is the average of *Spa.*

$$Spa = \sum_{i=1}^{n} \frac{1}{|x_i - x_{i+1}| + |y_i - y_{i+1}|} \qquad spatial\_proximity = \frac{\sum_{j=1}^{sup} Spa_j}{sup}$$

The temporal proximity measures the average temporal proximity of all composite IoT service instances. *temporal_proximity* is the average of *Temp.* $f_i(t)$is 0 or 1.

$$Temp = \frac{\int_{t_1}^{t_{2n}} \sum_{i=1}^{n} f_i(t)\, dt}{(t_{2n} - t_1) \cdot n} \qquad temporal\_proximity = \frac{\sum_{j=1}^{sup} Temp_j}{sup}$$

C. Estimate associated time interval and location for the candidates.

Definition 5: Periodic composite IoT service

A periodic composite IoT service *PC* is defined as the repeating composite IoT services at certain locations with regular time intervals. It is denoted by a tuple $PC = < CS, T, L, P >$, where *CS* is a composite IoT service, $T = < Ts, Te >$ is a representative time interval associated with *CS, L* is the region location of *CS, P* is the probability of *CS* occurring around time interval *T* at location *L.*

$$Dis(T, \tau) = \sum_{i=1}^{m} |T_s - st_i| + |T_e - et_i| \qquad P = \frac{Num}{TNum}$$

D. Measure how much convenience can be obtained by applying this.

Definition 6: Convenience

The IoT service events involved in PC_{m+1} is $\{b_1, b_2 \ldots b_m\}$. Suppose the actual event set occurs next is $\{c_1, c_2 \ldots c_k\}$, the amount of convenience can be quantified by:

$$convenience = \frac{|\{b_1, b_2 ... b_m\} \cap \{c_1, c_2 ... c_k\}|}{|\{c_1, c_2 ... c_k\}|}$$

E. PCMiner algorithm

The algorithm consists of four phases:

- The mining process starts with *dividing the search space*.
- PCMiner *searches all composite IoT services* in a determined space.
- PCMiner applies the *significance and proximity strategies* to remove non-promising composite IoT services.
- PCMiner collects time information and location information for candidates generated in the third phase.

Phase I: Dividing Search Space

The layout of a smart home consists of multiple regions such as a bedroom and a kitchen. Each IoT service event is associated with a region.

Phase II: Searching Event Patterns

PCMiner employs a divide-and-conquer, pattern-growth principle from Prefixspan.

Phase III: Calculate Significance and Proximity for Event Patterns.

Collect the time information and location information from the event sequences and calculate the statistic significance for each event pattern by Definition3. Then discard insignificant ones if its significance is less than the significance threshold *minsig*. Given proximity threshold *minpro*, we calculate average proximity for each event pattern by Definition4 and filter out those patterns whose proximity are less than *minpro*.

Phase IV: Generating Periodic Event Patterns

Definition 7: Projected database

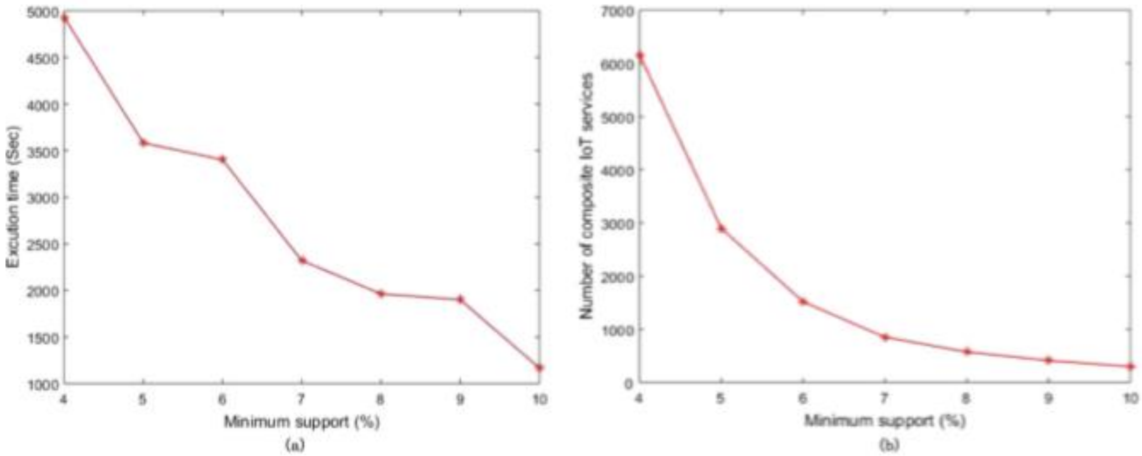
Let *p* be an event pattern in a database DB. The p-projected database is the collection of suffixes of event sequences in *DB* with regard to the prefix *p*. The searching process consists of three sub-phases:

- Find the set of 1-length event patterns L1
- Construct projected databases for each 1-length event pattern.
- k-length event pattern α is grown to the (k+1)-length event pattern α through searching the projected database DB|α corresponding to α (k ≥ 1).

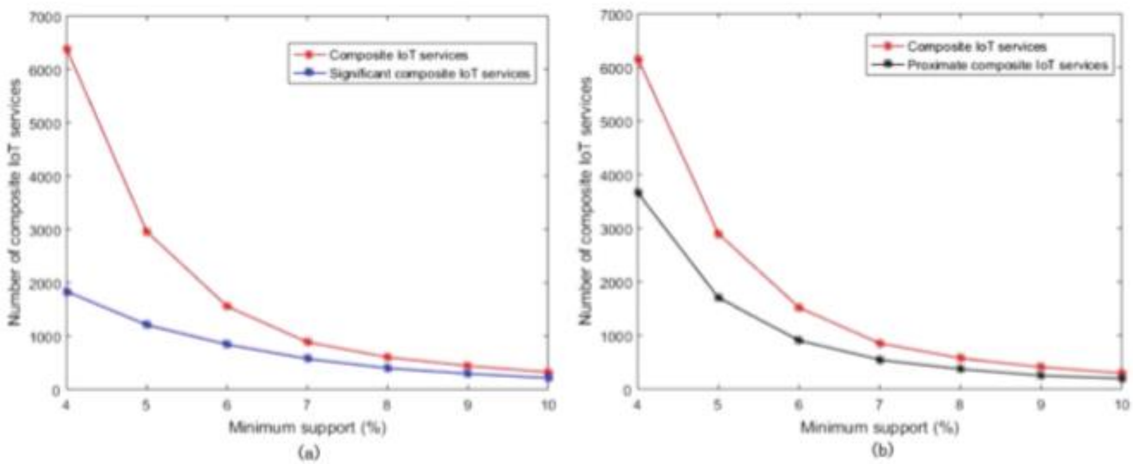
4. Validation

Data1 and Data2 are from CASAS datasets, which are collected in smart home environment. For Data3, 23 activities are recorded and annotated.

- The first set of experiments is conducted on a dataset which is a combination of three datasets. We vary the support threshold sup from 4% to 10%. Figure(a) shows the execution time of PCMiner decreases by increasing the support threshold. Figure(b) illustrates that the number of discovered event patterns decreases by increasing the support.



- In the second set of experiments, we assess the effectiveness of significance and proximity in pruning non-promising event patterns. We set the significance to be 0.01. We test the effectiveness of significance in reducing insignificant IoT service event patterns while varying different support threshold. Figure depicts the number of discovered patterns and significant patterns at different support threshold. The results show that the significance strategy performs effectively in pruning insignificant event patterns, which is an expected results.



- We perform the third set of experiments on Data3 to evaluate the applicability of our proposed approach. Table1 shows the primary discovered composite IoT services. Some of the composite IoT services are indeed difficult to be discovered because they are less frequent.

We conduct the fourth set of experiments on Data3 to measure how much convenience can be obtained by applying the discovered results in Table1. We showcase some preliminary results.

Periodic composite IoT services	Representative time intervals, location, probability
Taking medication	(1:54–2:00, in the kitchen, 0.5), (19:23–19:38, in the kitchen, 0.29)
Preparing breakfast	(5:45–6:20, in the kitchen, 0.75)
Preparing lunch	(11:15–12:05, in the kitchen, 0.64)
Preparing dinner	(18:15–18:48, in the kitchen, 0.5)
Going out for shopping	(7:52–8:05, in the hallway, 0.67)
Watching TV	(7:00–7:23, in the living room, 0.27), (14:27–15:22, in the living room, 0.33)