Personalized App Service Composition based on Usage Behavior of Smartphone Owners in IoT Environment

Jongmo Kim, Minhwan Kim, and Mye Sohn*
Department of Industrial Engineering, Sungkyunkwan University,
Suwon, Republic of Korea
{dignityc, kmh3178, myesohn}@skku.edu

Abstract

Smartphone owners use many smartphone applications to request the services needed in their daily lives such as shopping, payment or entertainment. Despite the behavioral change of the human, there are very few research initiatives to analyze the behavioral patterns of the apps usage in order to improve the utility of apps. As a result, smartphone owners should make decisions to select the appropriate apps in their context, and even perform to compose the services among different apps when they need to interact with each other. Unfortunately, the average number of apps installed on smartphone is about 86 in Korea, Japan, and America. The cumbersome of apps usage may be severe in the future. To overcome this, we propose the App Service Composition (ASC) borrowing the concept of Web service. In ASC, we find the app usage patterns with the user context using the machine learning techniques and automatically discover and select the heterogeneous app services using the Linked Open Data (LOD) technology. To prove the superiority of the framework, we perform the experiments to evaluate performance of the framework that can predict the apps usage patterns of the users.

Keywords: Smartphone Application, App Service Composition, Sequence Clustering

1 Introduction

Smartphone applications (a. k. a., apps.) are becoming the replacements for cash, plastic cards, and paper vouchers, which have been people's daily necessities. For example, in a physical store, smartphone owners may use the QR code app to get the products information, use payments app to buy the products, and use the tracking app to track the delivery. Furthermore, s/he may use the cooking app to get the recipes and use the SNS app to share his/her dishes. In other words, the smartphone and its apps are changing the world and people's daily lives. Despite this change, there are very few research initiatives focused on analyzing the apps usage behavior of the smartphone owners for improving the apps' utility [13]. Although some researchers have defined the app chain, which is a group of application sessions within the predefined time interval in terms of human-computer interaction (HCI) [9], they were not interested in a composition of the apps that took into account the explicit and/or implicit needs of smartphone owners. After all, smartphone owners must find themselves on their smartphones which apps or app combinations they need in certain situations. Unfortunately, the number of apps installed on smartphones is about 102 in Koreans, about 80 in Japanese, and about 76 in Americans according to a survey [1]. So, it is very cumbersome for the smartphone owners to find the app or composition of them. To reduce the burden, this paper proposes App Service Discovery & Selection (ASDS) framework borrowing the concept of Web Service Composition (WSC), which is the mechanism for combining and reusing existing web services to create new web services [6]. In the ASDS framework, the apps in the

Research Briefs on Information & Communication Technology Evolution (ReBICTE), Vol. 5, Article No. 3 (November 15, 2019)
*Corresponding author: InfoScience Laboratory, Department of Systems Management Engineering, Sungkyunkwan University, 2066 Seobu-ro, Jangan-gu, Suwon 440-746, Republic of Korea, Tel: +82-(0)31-290-7605

smartphone correspond to existing web services. The apps are dynamically combined and reused to meet the contextual needs of the smartphone owners. At this time, it is assumed that the context of the smartphone owners can be inferred using IoT technology.

This paper is organized as follow. Section 2 review the related works of services discovery and selection. In Section 3, we illustrated the motivating scenario and research challenges to help understand our proposed idea. Section 4 offers the overall ap services discovery and selection framework and their details. Section 5 demonstrate the superiority of the proposed framework. Finally, Section 6 presents the conclusions and further research.

2 Related Works

The service composition research in the IoT environment is very early stage. Most research exploit the idea of Service-Oriented Architecture (SOA)-based web service composition to resolve the various issues founded in service composition process [4]. The Quality of Service (QoS) is a significant information to evaluate and select the adequate service, so some studies proposed the QoS-aware service composition method using the optimization approach [3, 7]. However, the QoS information processing and their optimization solution commonly require the high computational cost, and it is not suitable to constrained IoT environment. To overcome the problem of constrained resources, some studies adopt cloud systems and utilize their resources [12, 2]. Recently, the research of service composition attempt to develop the methodology that is lighter than SOA and decentralized to avoid the heavy workload of specific devices such as mobile edge computing (MEC) [8, 5]. Although the various methodologies were proposed, most service composition studies focus on the system perspective such as the constrained resources or interoperability among services. There are few studies of service composition to improve the users' satisfaction on their context. A. Urbieta [10] proposed the adaptive and context-aware service composition method to catch the dynamic daily tasks of the user and service behaviors. To do this, they develop the dynamic reasoning framework with an abstract service model including service specification and context model. However, the reasoning method is difficult to reflect various service usage patterns for each individual and has a limitation in lack of flexibility to cover a huge service pool in IoT environment. To overcome limitations, we proposed the method of data-driven user service usage patterns recognition and LOD-based service model representation to ensure the flexibility.

3 Motivating scenario and research challenges

The motivating scenario is inspired by the daily life of the heavy users of smartphone apps. John, the movie-holic, visits a theater every weekend. He routinely performs all the steps necessary to watch a movie, from booking to ticketing, using a series of smartphone apps. First, He gets movie information such as showtimes, ratings, reviews, and synopsis through the Movie App. As a next, he uses the Movie Theater App to book a movie to play at the time he wants, as well as select seats. After this, he opens the Mobile Wallet App to pay for the tickets. At this time, he can choose the payment methods like Near Field Communication (NFC) or online payment. On the day of the movie screening, he visits the theater and buys drinks and snacks using the Mobile Wallet App. He routinely places orders at snack corner in the theater, pays by QR code by the Mobile Wallet App and picks the products, but sometimes uses the kiosks. After watching the movie, he also performs SNS activities such as uploading reviews. Depending on the context, he can watch different movies, visit to different theaters, and use different payment methods, but John would routinely carry out the above process to watch the movie. In this paper, it is called an App Usage Pattern (AUP). An illustrative AUP is depicted in Figure 5.

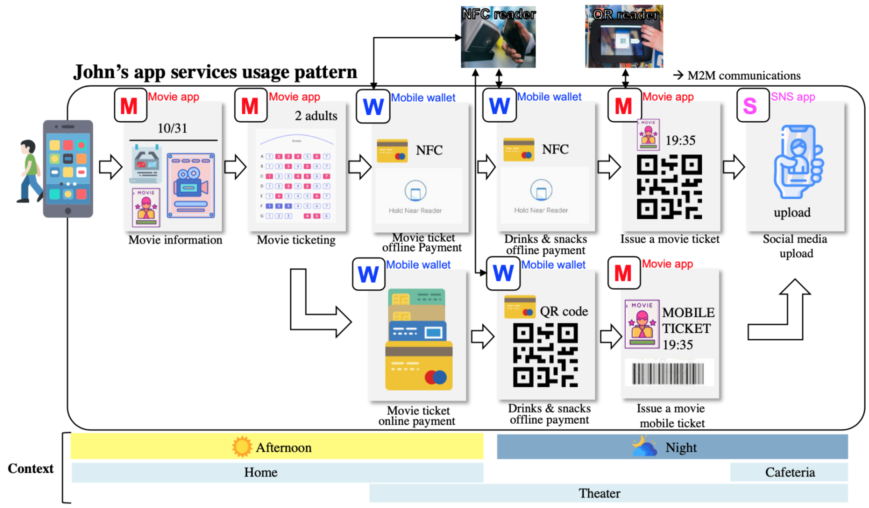

Figure 1: An illustrative example of Apps Usage Pattern for movie ticketing

4 Automated App Service Discovery & Selection (ASDS) Framework

The ASDS framework consists of two modules as shown in Figure 5: App Usage Patterns (AUP) recognition module and Proactive App services Discovery & Selection (PADS) module. The AUP recognition module finds the AUPs for smartphone users using the context-augmented and light-weighted machine learning methods. It is a novel approach that has not been proposed in WSC. The PADS module discovers app services and composed them like the WCS.

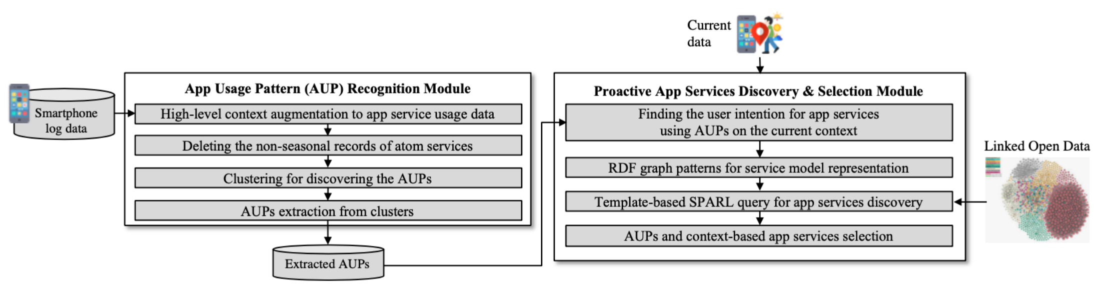

Figure 2: Overall execution procedure of ASDS Framework

4.1 App Usage Pattern (AUP) Recognition Module

In order to find the AUPs, we must extract log data only about the records of using atom services on the smartphone. Here, an atom service is defined as a single feature that makes up the specific app service. The atom services may be triggered by smartphone owners or pushed by the apps automatically depending on the context. In the previous scenario, the movie ticketing corresponds to the atom service. Since the log data source is his/her own smartphone, regardless of privacy concerns, we can collect execution records of the atom services on users' smartphone. The execution record is defined as follows.

Definition 1 j^{th} execution record of i^{th} atom service (r_{ij}) is simply represented by

$$r_{ij} = < t_{ij}, lo_{ij}, as_{ij} > \tag{1}$$

where $t_{ij} = \{st_{ij}, et_{ij}\}$, st_{ij} and et_{ij} is the timestamp on the start time and the end time of r_{ij}, respectively. $lo_{ij} = \{lt_{ij}, lg_{ij}\}$, lt_{ij} and lg_{ij} is the median of latitude and longitude changes during the execution of the i^{th} atom service. as_{ij} is a j^{th} label of i^{th} atom service.

However, a series of the execution records (r_{ij},$\forall i, j$) alone is not enough to accurately recognize the AUPs because it does not include high-level context required to infer a human perceptible situation like preference, taste, etc. So, in order to extend the r_{ij} into the high-level context, we augmented the temporal-spatial context. At this time, the temporal context is a time dependent and can inferred using web service like the Calendar or Weather at the time t_{ij}. And the spatial context is a location dependent and can inferred using GeoNames or PoIs database. We identified eight high-level context such as daily temperature, date, holiday, day of the week, time zone, geographical feature, point-of-interest (POIs), and address. As a result, r_{ij} is extended to context-augmented records $r^a_{ij}(|r^a_{ij}| > |r_{ij}|)$. Prior to recognizing the AUPs using r^a_{ij}, we trim non-seasonal records, which are event-driven to reduce the computational burden.

Deleting the non-seasonal records of atom services: Using the STL decomposition, which can decompose the time series data into trend, seasonality, and residual component [11], we estimate whether the records of the i_{th} service are seasonal. The degree of the seasonality of records of $as_i(dos_i)$ is calculated as follows.

$$dos_i = 1 - Var(R_i)/(Var(S_i + R_i)) \tag{2}$$

where R_i and S_i is the residual component and the seasonal component of the STL decompostion, respectively.

If dos_i is close to 1.0, the records r^a_{ij} ($\ni as_i$) have very strong seasonality. On the other hand, if it is close to 0 or negative, there is no seasonality. In order to determine whether the records of the i^{th} atom service are seasonal, we calculate the average of dos_i ($avg(dos_i)$) for all i and identify as_i less than $avg(dos_i)$ as the non-seasonal atom services (as^-_i). Then, it filters out all r^a_{ij}, which has as^-_i as an atom service label. Finally, we can obtain the tr^a_{ij} consisted of only seasonal records of the i^{th} atom service.

Clustering for discovering the AUPs: Using the tr^a_{ij} as an input, we perform the clustering to discover the AUPs. In order to perform the clustering, the distance matrix M ($N \times N, N = len(i) \times len(j)$) for tr^a_{ij} must developed. However, since tr^a_{ij} are mixed dataset with numerical and categorical data, it is necessary two types of distance measures to calculate a distance among records. So, we use Manhattan distance for the numerical data and propose new semantic distance for the categorical data. The newly proposed semantic distance is to calculate a distance between two words using the WordNet as the lexical knowledge base. The semantic distance of arbitrary two words ($SD(a,b), a, b \in i, a \neq b$) is calculated as follows.

$$SD(a,b) = log(CL_{a+}CL_b/2+1) + log(|CL_a - CL_b| + 1) \tag{3}$$

where CL_a is the number of classes that exist between arbitrary word a and a closest shared hypernym of two words on WordNet. CL_b is the same.

Using two distance measures, we generate distances matrix M and perform the clustering with it.

AUPs extraction from clusters: After clustering, we can obtain a series of clusters. Let C_k be k^{th} cluster that has tr_{ij}^{a} as an instance ($tr_{ij}^{a} \in C_k$) and the C_k corresponds to AUP_k. However, since the sequence pattern of the atom services is not known by C_k, the sequence patterns must be found among the instances of C_k and extracted as AUP_k. The process of the AUPs extraction from the clusters is summarized in Figure 5.

Input: a series of the execution records ($r_{ij}, \forall i, j$)
Output: App services Usage Patterns (AUP_k, , $\forall k$)

for all r_{ij} **do**
  for all API_s **do**
    $r_{ij}^{a} = r_{ij}$.append.(request(API_s, r_{ij}))
for all as_i **do**
  S_i, T_i, R_i = STL_decomposition($r_{ij} \ni as_i$)
  $dos_i = 1 - Var(R_i)/(Var(S_i + R_i))$
for all as_i **do**
  if $as_i \leq avg(dos_i)$ **then** $as_i^{-} = as_i$
for all r_{ij}^{a} **do**
  if as_{ij}^{a} ***not in*** $as_i^{-} (\forall i)$ **then** $tr_{ij}^{a} = r_{ij}^{a}$
perform clustering(method='OPTICS') **return** C_k
for all C_k **do**
  for all $as_{ij}^{a} \in C_k$ **do**
    avg_$st_i = 1/len(j) \sum_j st_{ij}^{a}$
  AUP_k= List_Index(sorting(avg_st_i, order= 'ascending'),i)
Return AUP_k

Figure 3: AUPs extraction algorithm using SLT-based filtering and the clustering

4.2 Proactive App Services Discovery & Selection (PADS) Module

Using the AUP_k, the PADS module discovers and selects the adequate app services to be composed depending on the current context of the smart owners. To do so, it should resolve two issues. The first is to notice their intention before smart owners execute app services. There are two approaches to notice the smart owners' intention. One is reactive approach and the other is proactive approach. The former instantly discovers and selects the adequate app services according to the changing smartphone owners' context or situation. So, it requires significant processing time before the service is delivered to smartphone owners. To overcome the limitation, we adopt the proactive approach. The second issue is the heterogeneity among the app services specification. Although the smartphone owners want to utilize the multiple app services seamlessly as a single service, the heterogeneity among the app services may hinder it. As a result, even if the adequate app services may exist for smartphone owners, it is very difficult to discover and select automatically by the system. To solve the problem, we designed the RDF graph patterns to flexibly represent the service model based on OWL-S.

Finding the intention of the smartphone owners: First of all, it constantly collects records and augments their high-level context information in time interval $d(r_{ic}^{a})$ from current time. The matching among AUPs and collected records ($r_{ic}^{a}, \forall i, c, c > j$) is calculated by sum of context similarity and sequence similarity. In the matching, the numerical data of records such as time and location data cannot be considered due to the lack of abstraction. The context similarity (CS) is calculated by the categorical distance between AUPs and r_{ic}^{a} using $SD(a, b)$ in previous module. The sequence similarity (SS) is estimated by matching ratio between the sequence of app services for AUPs and as^a_{ic}. The matching function between AUP_k and r^a_{ic} ($Ma(AUP_k, r^a_{ic})$) is formulated as follows.

$$SS = 1/(len(l)) \times \underset{u}{Max}(|\{as^a_{ic} \in sub_{ku} : sub_{ku} \subset AUP_k\}|) \tag{4}$$

where $Ma(AUP_k, r^a_{ic}) = CS + SS$, $CS = AD(r^a_{ic}, AUP_k)/\underset{i,j,c}{Max}(SD(r^a_{ic}, tr^a_{ij}))$, $AD(r^a_{ic}, AUP_k)$ is average distance between r^a_{ic} and the records included in AUP_k ($tr^a_{ij} \in AUP_k$), sub_{ku} is a subsequence included in AUP_k, u is a length of sub_{ku} ($u = |sub_{ku}|, 1 \leq u \leq l$)

We find a most similar AUP^*_k by the matching function and obtain a list of the adequate app services from AUP^*_k in current context.

RDF graph patterns for app services model representation: After finding the intention from AUP^*_k, we discover the real-world app services and its service providers that are needed in the current context. In order to guaranty the interoperability among them, the representation of the semantic service model is required. In WSC, the Ontology Web Language for Services (OWL-S) was proposed to semantically represent the web services. However, OWL-S requires a strict definition of the concept and a minimalistic approach to representing the services, which makes it inflexible to cover various IoT services and their components. So, we proposed a RDF graph pattern for the representing of the app services model in IoT based on OWL-S. To do so, we identified the necessary components to represent the app services that cannot be included in OWL-S and add them to the RDF graph pattern. The RDF graph pattern for the app services model is depicted in Figure 5. We additionally identified the smartphone model and added it to represent the log data of smartphone. Also, user entity is added to represent user context in consideration of temporal and spatial features.

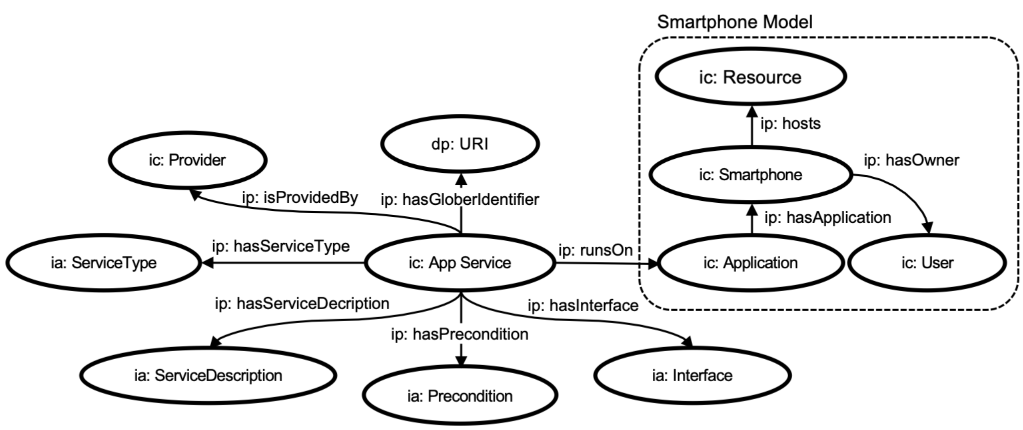

Figure 4: RDF graph pattern for app services model representation in IoT environment

Using the proposed RDF graph pattern and a list of the adequate app services, we perform the template based SPARQL query writing and execute it. Then, we select the most adequate real-world app services in query results by the context and requirements matching.

5 Performance Evaluation

In order to show the superiority of the ASDS framework, we created three app services usage scenarios to create the virtual dataset: request for food delivery, order and pay for coffee and drinks in a café, and use for payment and membership in convenience stores. Using three scenarios, we generated the total 610 app usage log data and about 40 sequences for each scenario. Each scenario corresponds to the patterns that we want to discover as APUs. Using the virtual dataset, we performed the two experiments: the effect of the high-level context augmentation and the performance of the semantic distance measure. At this time, we use Adjusted Rand index (AR), Completeness Score (CS), Normalized Mutual Information (NMI), and V-measure (Vm) as the evaluation metrics. We performed 50 times clustering and calculated their average value of the metrics. The results are shown in Figure 5. The left pane of Figure 5 shows the superiority of high-level context augmented dataset. The original log dataset without context has no information to discovery AUPs. It is necessary to infer the high-level context information to discovery the AUPs as well as easy to obtain them from external systems. The right pane of Figure 5 is the results about the performance of semantic similarity using wordnet. In general, the one-hot-encoding method is used to transform the categorical data into pseudo numerical code that can be inputted to clustering algorithm. However, our semantic distance outperforms the one-hot-encoding approach in all metrics. Even with over 93 % accuracy in all metrics, we proved that our method can discovery AUPs.

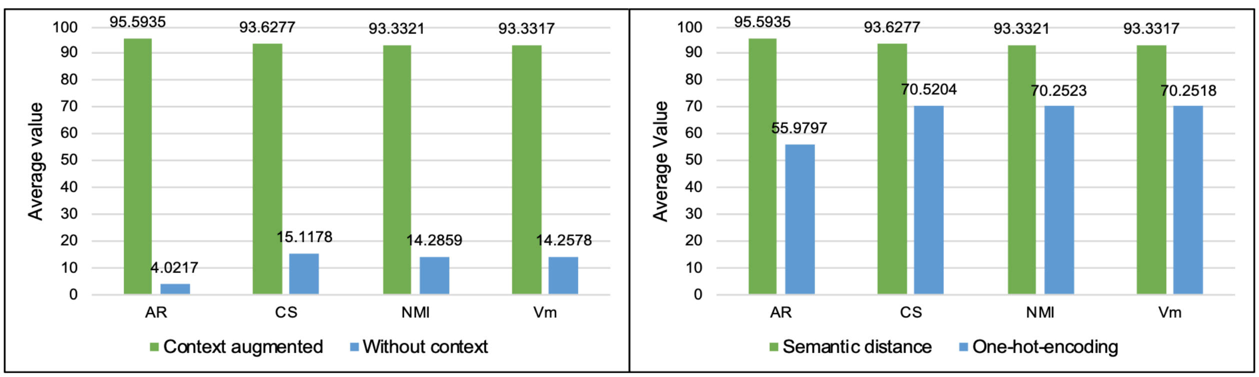

Figure 5: Experiment results with context augmentation (left-hand) and semantic distance (right-hand)

6 Conclusion Further Research

We proposed the framework to recognize the users' app usage patterns using STL decomposition and clustering method from the smartphone log data and to discover and select the adequate services with them. Also, we demonstrated the feasibility of our system through log data generated by identified scenario and experiments. In some part, however, our research has limitations. First, the clustering method becomes inefficient both in terms of accuracy and processing time as the data size increases. Therefore, the more fast and efficient method is needed to recognize AUPs than clustering to deal with huge smartphone log dataset. Second, we proposed the service model based on RDF graph pattern but could not provide a concrete way to query it. Since LOD queries are complex and require a long response time, we need a way to overcome this problem. In the future, we will design and develop advanced methods around these identified issues and deal with long-term smartphone user log data to demonstrate superiority of our method.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2016 R1D1A1B03932110).

References

[1] APP-Annie. 2017 restrospective: A monumental year for the app economy. In *A Monumental Year for the App Economy.*, pages https://www.appannie.com/en/insights/market–data/app–annie–2017–retrospective, 2019.

[2] T. Baker, M. Asim, H. Tawfik, B. Aldawsari, and R. Buyya. An energy-aware service composition algorithm for multiple cloud-based iot applications. *Journal of Network and Computer Applications*, 89:96–108, 2017.

[3] B. Cao, J. Liu, Y. Wen, H. Li, Q. Xiao, and J. Chen. Qos-aware service recommendation based on relational topic model and factorization machines for iot mashup applications. *Journal of Parallel and Distributed Computing*, 132:177–189, 2019.

[4] B. Cheng, D. Zhu, S. Zhao, and J. Chen. Situation-aware iot service coordination using the event-driven soa paradigm. *IEEE Transactions on Network and Service Management*, 13(2):349–361, 2016.

[5] S. Deng, L. Huang, J. Taheri, J. Yin, M. Zhou, and A. Y. Zomaya. Mobility-aware service composition in mobile communities. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(3):555–568, 2016.

[6] R. Karunamurthy, F. Khendek, and R. H. Glitho. A novel architecture for web service composition. *Journal of network and computer applications*, 35(2):787–802, 2012.

[7] M. E. Khanouche, F. Attal, Y. Amirat, A. Chibani, and M. Kerkar. Clustering-based and qos-aware services composition algorithm for ambient intelligence. *Information Sciences*, 482:419–439, 2019.

[8] T.-D. Nguyen, E.-N. Huh, and M. Jo. Decentralized and revised content-centric networking-based service deployment and discovery platform in mobile edge computing for iot devices. *IEEE Internet of Things Journal*, 2018.

[9] T. Soikkeli, J. Karikoski, and H. Hammainen. Diversity and end user context in smartphone usage sessions. In *2011 Fifth International Conference on Next Generation Mobile Applications, Services and Technologies*, pages 7–12. IEEE, 2011.

[10] A. Urbieta, A. González-Beltrán, S. B. Mokhtar, M. A. Hossain, and L. Capra. Adaptive and context-aware service composition for iot-based smart cities. *Future Generation Computer Systems*, 76:262–274, 2017.

[11] X. Wang, K. Smith, and R. Hyndman. Characteristic-based clustering for time series data. *Data mining and knowledge Discovery*, 13(3):335–364, 2006.

[12] T. Wu, W. Dou, C. Hu, and J. Chen. Service mining for trusted service composition in cross-cloud environment. *IEEE Systems Journal*, 11(1):283–294, 2014.

[13] S. Zhao, J. Ramos, J. Tao, Z. Jiang, S. Li, Z. Wu, G. Pan, and A. K. Dey. Discovering different kinds of smartphone users through their application usage behaviors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 498–509. ACM, 2016.

Author Biography

Jongmo Kim is in Ph. D. course in the department of Industrial Engineering at Sungkyunkwan University. He received his bachelor's degree from a Sungkyunkwan University. His main interests are ontology, semantic web, Linked Open Data, web service composition, and Web-of-Things.

Minhwan Kim is in Ph. D. course in the department of Industrial Engineering at Sungkyunkwan University. He received his bachelor's degree from a Sungkyunkwan University. His main interests are ontology alignment, semantic web, and graph mining in Linked Open Data.

Mye Sohn is a professor in the Department of Systems Management Engineering at Sungkyunkwan University. She received her MS and Ph. D. from the Korea Advanced Institute of Science and Technology (KAIST). Her main interest is the machine learning, ontology, Web-of-Things, Semantic Web, and so on.