

Clustering Approach of Device Interactions for Automatic Generation of Device Social Relation

Dong-Oh Kang, Jang-Ho Choi, Joon-Young Jung, and Changseok Bae

Abstract—Recent increase of devices around us requires autonomous device interactions for convenient device collaboration. In this paper, we proposed a method to generate the device social relation without human intervention by clustering approach of device interactions. With the previously proposed device sociality framework, the proposed method can give autonomous device collaboration by using device sociality concept to minimize human intervention. To show the feasibility of the proposed method, we applied it to a group of smart devices in an office.

Index Terms—Clustering, device social relation, device collaboration, autonomous.

I. INTRODUCTION

Recently, many smart devices around us are used to interconnect the other devices through network due to the trend of enormous increase of a number of smart devices like smartphones, tablets, and smart TV's, etc. Especially, the incoming advent of Internet of Things will ignite the tremendous increase of device interactions. However, different from usage of single device, mutual interaction of multiple devices requires the configuration settings to interconnect with each other. If the number of devices around us becomes bigger than the number that human can deal with, the manual configuration of device interaction will be impossible or the very tedious job.

Some techniques have been introduced for mutual interaction of smart devices [1]–[5]. They studied about interoperability problem among devices for device collaboration. However, users have complained that the methods are not satisfactory enough in terms of convenience of mutual interaction because users should know the technical details of devices and network technology. Furthermore, if the number of devices increases, the previously suggested methods cannot provide solutions to the inconvenience at all. Therefore, the technology to make multiple devices interconnect and interact is needed with minimum human intervention in the process of mutual interaction.

In the area of ubiquitous computing, there were studies on device's self-awareness and smart response to the user's requests [1]. In robotics area, there were studies about how to assign autonomy level of machine [2]. And, recent researches of Machine-to-Machine communication, Internet of Things,

and Web of Things studied about how to manage and connect devices within M2M network, Internet or Web environment [3]–[5]. Some researches proposed the methods to use on-line human social media to connect devices [6], [7]. A rule-based autonomous method was proposed for the self-adaption abilities of machines to change their own actions autonomously against environment [8]. And, there have been some other studies to connect devices by using the concept of social relation among devices [9]–[13]. However, few researches dealt with the automatic generation of device social relation and how to make devices collaborate without human intervention.

In this paper, we propose how to classify device interaction relations automatically by clustering in terms of device interaction in a group of devices. The classification of device interaction relations is used to generate the device social relations. The derived device social relations of the group of devices are used to interconnect devices with the device sociality framework [12], and can minimize human intervention in the process of mutual device interaction. Therefore, the proposed method will provide autonomous collaboration of smart devices.

The outline of the paper is as follows: In Section II, we introduce some concepts of device sociality and device sociality framework. In Section III, we propose a method to classify device interaction relations by clustering approach and to derive the device social relations based on the clusters. In Section IV, we apply the proposed method to a group of devices in an office. Finally, we conclude research results of this paper and provide the discussion about the further studies in Section V.

II. DEVICE SOCIALITY FRAMEWORK

If there are many devices, the social relation among devices can be defined like human social relation [9]. The device social relation can be defined as the possible set of device interactions [12]. We proposed the device sociality framework for autonomous device interaction of smart devices by introducing the concept of device sociality [12]. The device sociality framework offers the software framework to deal with device social relation among smart devices and provide users autonomous device collaboration (see Fig. 1). Therefore, if the device social relation is well constructed, the autonomous device interaction is possible by referring to the device relation in the device sociality framework.

A. Device Sociality

We defined some notions related with device sociality in [14]. The device social relation is the relation to define

Manuscript received October 11, 2014; revised December 12, 2014. This research was supported by the ICT R&D program of MSIP/IITP, [K1400011001, Human Friendly Devices (Skin Patch, Multi-modal Surface) and Device Social Framework Technology].

The authors are with the Electronics and Telecommunications Research Institute, Daejeon, Korea (e-mail: dongoh@etri.re.kr, janghochoi@etri.re.kr, jjjung21@etri.re.kr, csbae @etri.re.kr).

permissible sets of device interactions among devices. The device social relation can be categorized as similar to human social relations like families, friends, and followers. The device sociality is the metadata to define the device social relation by using machine understandable description language like XML, JSON, etc. Therefore, the device sociality represents the descriptive modelling of the device social relation.

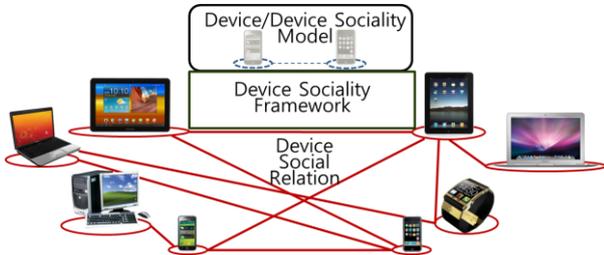


Fig. 1. Concept of device social relation and device sociality framework.

Because the device social relation is defined as the server/client relation of shared resources with its members, the network of the social relations can be a directed graph among devices (see Fig. 2). The knowledge from the graph represents the relational modelling of the device social relation.

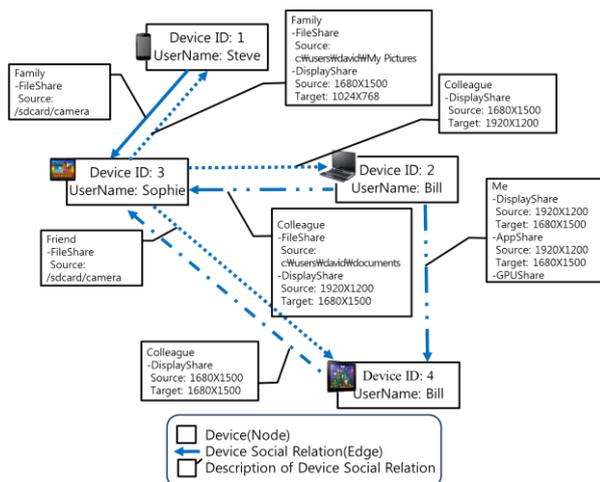


Fig. 2. An example of the graph of device social relations [12].

B. Device Interaction with Device Sociality

We classified the resources of device as three categories: computing resource layer, data layer, and service layer [15]. The members of peer devices of a device sociality can request the resources of the owner device of the device sociality. The device sociality framework manages the flow of the information of the device sociality in the process of device interaction. By using the device sociality framework, the client of the shared resources can get the permission to access the share resources which are permitted to the client by the owner device as described in the device sociality. The can be two types: Server-based or P2P(Peer-to-Peer). In [12], the system of D-SNS(Device Social Network Service) was introduced where there is the management server of D-SNS service, which has the information of devices and their social relations similar to the role of the server of human on-line social network service. The exchange of the information of device sociality can be made in a way of P2P between devices without the management server. In case of P2P style, there may be synchronization mechanism to synchronize the

contents of device sociality of devices with the contents of device sociality in the server. From the device sociality framework, the devices can interact with each other on the basis of device sociality.

C. Autonomous Device Interaction with Device Sociality

If the device sociality between devices can be constructed without human intervention, autonomous device interaction can be possible via the device sociality framework. Therefore, in this paper, we propose how to formulate the device social relation between devices without human intervention. For the purpose, the device interactions of devices within a community of people can be monitored. The device interactions are clustered based on the patterns of device interactions. The clusters are classified with features. After the process of clustering, we can assign the device social relation to the cluster of device interactions (see Fig. 3).

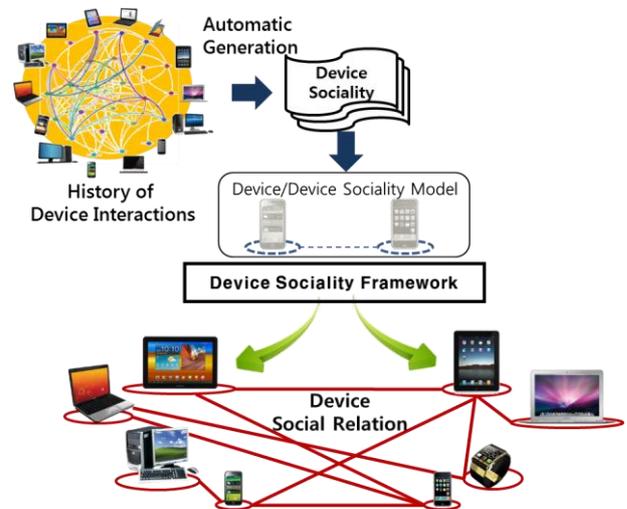


Fig. 3. Concept of autonomous device interaction with device sociality.

III. CLASSIFICATION OF DEVICE INTERACTION

A. Overview of Automatic Generation of Device Social Relation

In this paper, automatic generation of device social relation is performed as the following steps (see Fig. 4).

- 1) Monitoring of the device interactions of a group of devices
- 2) Clustering of the device interactions
- 3) Similarity matching of a new device with other devices
- 4) Derivation of clusters of device interactions based on the similarity
- 5) Assignment of device social relations based on the clusters

In this paper, a closed group of devices is considered to derive their social relation. By monitoring the device interactions among the member devices of the closed group, the history of device interactions are accumulated. From the data accumulated, clusters of device interaction relations can be derived without human intervention by the proposed clustering approach. After the automatic generation of the clusters of device interaction relations, clusters of device interactions are assigned to the new device as device interaction relations when a new device joins the group. By a

similarity matching process, device interaction relations of the device similar to the new device is assigned as the new device. Finally, the assigned clusters of device interactions to the new device are converted to device social relations by adopting the device interactions of the device interaction relations over certain threshold values.

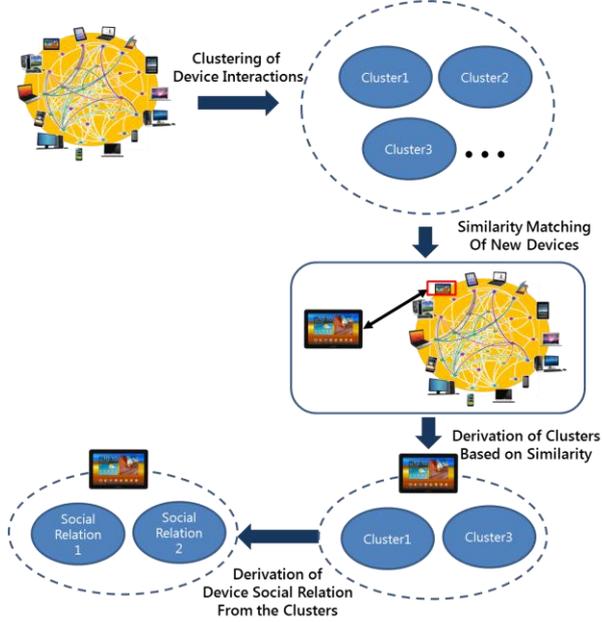


Fig. 4. Overall procedure of automatic generation of device social relation.

TABLE I: THE FEATURE ELEMENTS OF A DEVICE INTERACTION

Type	Property	Number	Example
Device	Category of device	1	Smartphone, laptop, etc.
User	Role of the user	1	Project manager, etc.
Resource	Type of resource	Many	E-mail, phone call, etc.
Strength	Frequency of interactions	Many	Any integer value like 42, etc.

B. Clustering of Device Interactions

In order to cluster the device interactions, the feature elements of device interaction should be defined. In this paper, the feature elements of device interaction are defined as follows (see Table I).

- 1) Device to participate the device interaction
- 2) User of the device
- 3) Resources to share in the device interaction
- 4) Strength of device interaction of sharing of resources

In 1), because it is hard to use the device identifiers as a feature of device interaction due to curse of dimensionality, the type of the device is used instead like a smartphone, a laptop, and a tablet, etc. That means that two devices may be classified as a cluster if two devices have similar types. Similarly, the user identifier cannot be used directly due to the same reason with the device identifier in 2). Rather than the user identifier, some types of the user can be used like teachers and students in a group of the school. That means that two devices may be classified as a cluster if two devices have similar users to have similar roles in the group. In 3), the classes of resources are used like phone calls, e-mail, etc. That means that two devices may be classified as a cluster if two devices have similar resources. In 4), the strength of device interaction is calculated as the frequency of the device

interaction of a resource. That means that two devices may be classified as a cluster if two devices have similar patterns of device interactions.

C. Similarity Matching Process

In this paper, the value of cosine similarity of feature vectors of device interactions except the strength is used as similarity measure. Because the new device has no strength of device interactions, the elements of strength of device interactions are omitted in the similarity matching process. In order to calculate the similarity between two devices, the elements of feature vectors are transformed into numbers. In transformation to numbers, similar elements may be assigned subsequent values. For example, in case of device categories, the computing power of devices is used to order the values of device feature. Therefore, in terms of device feature, the value of a smartphone is smallest, and the value of a desktop is highest when devices considered are smartphones, tablets, and desktops.

D. Derivation of Device Social Relations

In the derived clusters from similarity matching process, every device interaction has the value of strength. A threshold values is assigned to each device interaction and device interactions of the strength values over the threshold values are adopted into the device social relation corresponding to the cluster. If we want to have closer device relationship, we can set smaller threshold values.

IV. APPLICATION

In this section, we will examine the feasibility of the proposed method by applying to devices of members in an office. The office has ten members. There are four project groups where seven persons participate in over two project groups simultaneously. The four types of devices are used: smartphones, tablets, laptops, and desktops. The total number of devices is twenty-five. The roles of users are as follows: project managers, project members, team manager. Overall monitoring duration is a week. Totally 397 device interactions happened during the monitoring duration. The number of device interactions can be accumulated to corresponding pairs of devices (see Fig. 5). In Fig. 5, the device corresponding to the number of identifiers of devices between zero to nine means the desktop. There are many device interactions between desktops because the members of the office work at the office mainly. However, some members used the smartphones also as the device corresponding to the number of identifiers of devices between fourteen to nineteen means the smartphone.

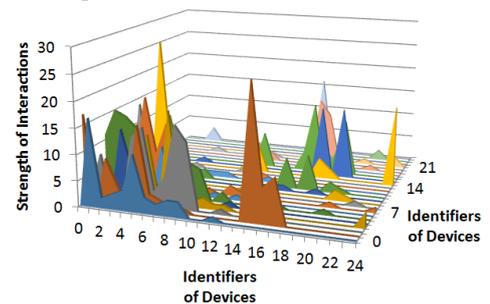


Fig. 5. Strength of device-to-device interactions in an office.

For clustering, the values of the number are assigned to the elements of the features of the device interactions (see Table II). Based on the elements, there are 77 types of device interactions in the experiment.

TABLE II: THE ASSIGNED NUMBERS TO FEATURE ELEMENTS OF A DEVICE INTERACTION

Type	Element	Value
Device	Smartphone	1
	Tablet	2
	Laptop	3
	Desktop	4
User	No Relation	0
	Project Member	1
	Project Leader	2
Resource	Team Leader	3
	E-mail	1 or 0
	Short Message	1 or 0
	Phone call	1 or 0

We applied the k-means algorithm for clustering of device interactions. By using three and four clusters in classification process of the k-means algorithm, we had the average feature vectors of clusters (see Table III). D1 and D2 are device types of device interaction. U1 and U2 mean user types of devices. E means the e-mail device interaction. S means the short message, and P means the phone call.

TABLE III: CLUSTERING RESULTS

# of Clusters	Cluster	D1	D2	U1	U2	E	S	P
3	1	4	4	2.2	1.5	13.2	0	0
	2	2.6	1.3	0.6	0.7	0.8	12.7	0
	3	3	3	1.4	1.1	1.9	0.4	0.1
4	1	4	4	2.3	1.5	13.2	0	0
	2	2.6	2.4	1.8	1.4	0.8	0.5	0.1
	3	2.6	1.3	0.6	0.7	0.8	12.7	0
	4	4	4	0.8	0.6	3.8	0	0

We assume a new smart device has the feature vector as follows:

$$[D1, D2, U1, U2, E, S, P] = [3, x, 3, x, 1, 1, 1] \quad (1)$$

where x means any value possible.

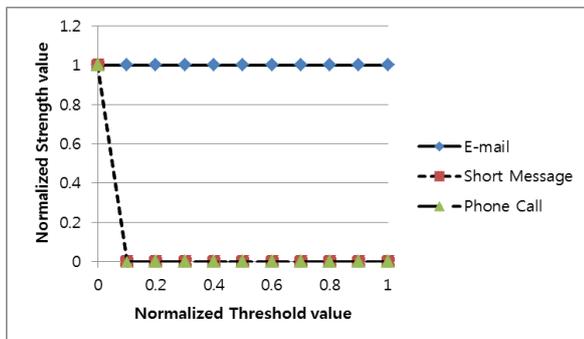


Fig. 6. Derivation of device social relation by changing the normalized threshold in case of three cluster classification.

In (1), x is assigned to zero because the value should not affect the result. By the similarity matching with average

values of the clusters, the cluster 1 is selected in case of three cluster classification. In case of four cluster classification, the cluster 2 is selected. After selecting the most similar cluster to the new device, device social relation is derived by restricting the device interactions which have the strength values below the threshold. If the threshold becomes higher, more device interactions are restricted (see Fig. 6 and Fig. 7). There are the changes of device interactions of the selected device social relation when the normalized threshold value for device social relation changes from 0 to 1.

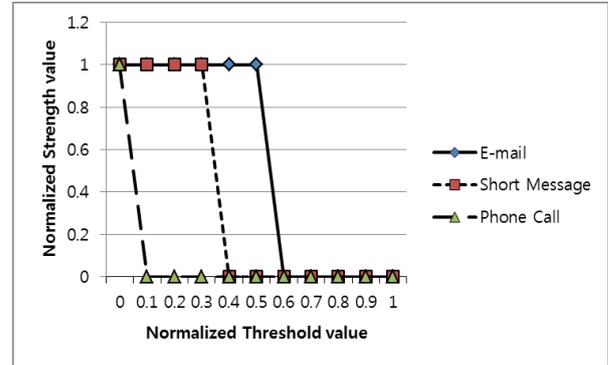


Fig. 7. Derivation of device social relation by changing the normalized threshold in case of three cluster classification.

If the more clusters are used, the device interactions can be selected the more specifically. As said in Section III, closer device social relation can be derived with smaller threshold values for the close relationship of device interactions because the small value of the threshold value permits wide range of device interactions.

V. CONCLUSION

In this paper, we proposed the method to generate device social relations automatically from clustering approach of device interactions. The automatically derived device classification from the clustering can be used when a new device is connected with other devices. The method can give easy collaboration of smart devices by minimizing human intervention with the device sociality framework to manage the device social relations and interconnect sharable resources of devices according to the device social relations. For the future research, we will study how to derive the feature vectors of device interactions relevant to the clusters in order to assign the device social relations whereas the average value of device interactions in clusters is used in this paper.

REFERENCES

- [1] B. Brumitt, B. Meyers, J. Krumm, A. Kern, and S. Shafer, "EasyLiving: Technologies for Intelligent Environments," *Handheld and Ubiquitous Computing*, 2000, pp. 97-119.
- [2] H. Huang, K. Pavsek, B. Novak, J. Albus, and E. Messina, "A framework for autonomy levels for unmanned systems (ALFUS)," in *Proc. The AUVSIUS North America 2005*, 2005, pp. 849-863.
- [3] K. Kouji, N. Yoshitaro, and S. Tadashi, "M2M service platform to support carrier cloud," *NEC Technical Journal*, vol. 5, no. 2, pp. 116-121, 2010.
- [4] A. G. Garca et al., "Introduction to devices orchestration in Internet of Things using SBPMN," *Int. Journal of Interactive Multimedia and Artificial Intelligence*, vol. 1, no. 4, pp. 16-22, 2011.

- [5] S. Mayer and D. Guinard, "An extensible discovery service for smart Things," in *Proc. the 2nd Int. Workshop on Web of Things*, 2011, pp. 7-12.
- [6] A. Kamilaris and A. Pitsillides, "Social Networking of the Smart Home," in *Proc. the 2010 IEEE 21st Int. Symposium on Personal Indoor and Mobile Radio Communications*, 2010, pp. 2632-2637.
- [7] A. Pintus, D. Carboni, and A. Piras, "Paraimpu: a platform for a social Web of Things," in *Proc. the 21st Int. Conference Companion on World Wide Web*, 2011, pp. 401-404.
- [8] H. Shuang and Y. Zhou, "A study of autonomous method of IoT component," in *Proc. Int. Conference on New Trends in Information Science and Service Science*, 2011, pp. 294-298.
- [9] L. Atzori, A. Iera, and G. Morabito, "SLoT: Giving a social structure to the Internet of Things," *IEEE Communications Letters*, vol. 15, no. 11, pp. 1193-1195, 2011.
- [10] T.-Y. Chung, I. Mashal, O. Alsaryrah, V. Huy, K. Wen-Hsing, and D.P. Agrawal, "Social Web of Things: a survey," in *Proc. IEEE Int. Conference on Parallel and Distributed Systems (ICPADS)*, 2013, pp. 570-575.
- [11] K. Kang, D. Kang, and C. Bae, "Novel Approach of Device Collaboration Based on Device Social Network," in *Proc. the 2013 Int. Conference on Consumer Electronics*, 2013, pp. 248-249.
- [12] D. Kang, K. Kang, and C. Bae, "A device sociality framework to enhance zero configuration degree of interaction of smart devices," in *Proc. Int. Conference on Internet(ICONI) 2013*, 2013, pp. 161-166.
- [13] L. Atzori, A. Iera, and G. Morabito, "From "smart objects" to "social objects": The next evolutionary step of the Internet of Things," *IEEE Communication Magazine*, vol. 52, no. 1, pp. 97-105, 2014.
- [14] *Reference Model for Services based on Device Sociality*, Korean Standard of TTA (in Korean), TTAK.KO-10.0608-2012.
- [15] D. Kang, K. Kang, J. Choi, and C. Bae, "A bridge structure of device sociality framework for smart home service," in *Proc. IEEE Int. Symposium on Consumer Electronics*, 2014, pp. 1-2.



Dong-Oh Kang was born in Busan, Korea, in 1969. He received his B.S. degree in electronic engineering from Yonsei University, Seoul, Korea, in 1994. And, he received his M.S. and Ph.D. degrees in electronic engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1996 and 2001 respectively. Since 2001, he has been working at

Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea. And, he is a principal researcher of the human computing research section of ETRI where he is developing an intelligent software framework for device collaboration. His research interests include home network middleware, distributed control, mobile computing, and cloud computing service.



Jang-Ho Choi received his bachelor's degree in computer science & business administration in 2010 from University of Southern California, USA. And, he received his M.S. degree in computer science from Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2013. Then, he has joined Human Computing Research Section at Electronics and Telecommunications Research Institute (ETRI) as a researcher from 2013. His research interests include Internet/Web of Things, End-user Software Engineering, and HCI.



Joon-Young Jung was born in Seoul, Korea, in 1971. He received his B.S. and M.S. degrees in computer network engineering from Soongsil University, Korea, in 1996 and 2000, respectively. Since 2000, he has been a researcher at the Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea. He is currently developing a distance estimation algorithm using wireless devices for space awareness. His research interests include mobile computing, wireless networking, and multimedia network service.



Changseok Bae received his B.S. and M.S. degrees in electronic engineering from Kyungpook National University, Korea, in 1987 and 1989 respectively. He also received his Ph.D. degree in electrical and electronic engineering from Yonsei University, Korea, in 2003. From 1989 to 1996, he was a senior researcher at Systems Engineering Research Institute (ETRI), where he worked on image processing and pattern recognition. From 1997 to 1999, he worked with Korea

Ministry of Information and Communication, where he participated in establishing national software research and development policy. From 2000 to 2004, he was a principal research staff of Embedded System Architecture Research Team at Electronics and Telecommunications Research Institute, where he worked on multimedia functions in the home server. From 2004 to 2005, he was a research fellow at School of Information Technologies in University of Sydney, Australia, where he studied on an error resilient multimedia codec algorithm. Since 2006, he has been a principal research staff and the team leader of Personal Service Research Team at Electronics and Telecommunications Research Institute, where he is developing a personalized service framework based on the life log and smart object called the intelligent gadget. His research interests include image processing, multimedia codec, personalized service framework, smart object, and life log.