Homogeneous Ensemble Instance Intervals Determination Method of Time Series Data Based on Granular Computing

Jaewoong Kang, Wooseong Yang, and Mye Sohn

***Abstract*—It is very important to determine the size of the instance since it has a large impact on the recognition performance of the devices. In this paper, we propose a novel method to recognize the intervals of the time-series data using granular computing. Unlike traditional methods which use fixed size or knowledge-based, our method is conducted data-driven. Based on the concept of the granular computing, we classified the operation data of devices into three levels and proposed a multi-SVM-based machine learning method that can automatically classify each granule. We have proven the excellence of our method by conducting and evaluating experiments with two perspectives.**

***Index Terms*—Feature selection, granular computing, instance interval, time-series data.**

I. Introduction

In general, the operation data of devices are recognized using time-series data for which the size of an instance is not known. However, since it has a large impact on the recognition performance of the devices, it is very important to determine the size of the instance. To do so, some researchers proposed some ways to amplify the volume and the type of data that can be collected by increasing the number of sensors instead of determining the exact size of the instances [1]. Recently, some methods are proposed that extract the exact motion data in interest using the deep learning method [2]. But in the former, it is becoming increasingly difficult to apply it due to the increased complexity of data patterns and the resulting computational cost even though recognizable activities have diversified. In the contrary, in the latter, the accuracy of the activity recognition is very high. However, it takes a lot of computational burden to find the exact interval that varies depending on the types of them.

To overcome the limitations, we attempted to use granular computing [3] to recognize the activities. Based on the concept of the granular computing, we classified the operation data of devices into three levels and proposed a multi-SVM-based machine learning method that can automatically classify each granule using time-series data. We also proposed a feature selection method that maximizes the classification performance according to the level of each granularity. By using granular computing on the recognition of the operation data, it becomes possible to subdivide the predefined activities, and to extract excellent features to find subdivided activities. As result, it is possible to find out the start and the end of the instances more clearly, and then extract the excellent features. This makes it possible to derive high performance with basic machine learning methods rather than deep learning.

The paper is organized as follows. Section II reviews the related research. Section III offers detailed descriptions about the overall architecture and the components of the framework. In Section IV, experimental results are suggested to demonstrate the effectiveness of the framework. Finally, Section V presents the conclusions and further research.

Manuscript received July 15, 2019; revised May 2, 2020. This research is supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2019R1A2C1004102).

Jaewoong Kang, Wooseong Yang, and Mye Sohn are with the Sungkyunkwan University, Suwon, Korea (e-mail: {kjw1727, yus0363, myesohn}@skku.edu).

II. Related Works

The multimodal sensor data is data that are collected from various types of sensors such as auditory, visual, and state for specific purposes. The characteristic of this data is that it has a lot of information due to its large volume compared to that collected from a single sensor [4]. As shown in Table I, the multimodal sensor data has been used in many fields that need a large volume of information such as medical, robotics, activity recognition Nonetheless, the heavy volume of the multimodal sensor data may cause a computational burden [5]. To reduce the burden, granular computing is emerging as a solution.

Granular computing is defined as an umbrella term to cover any theories, methodologies, techniques, and tools that make use of granules in complex problem solving [6]. A granule which is a basic element of the granular computing is a small particle especially, one of the numerous particles forming a larger unit [6]. The granule is made by any elements that are bundled by similarities, differences, and functions. Furthermore, data granularity is a measure that represents a scale of the granule. In the granular computing, the key features may varied depend on the data granularity. So, it requires different learning models that can reflect the characteristic of the granular computing. As result, the prediction performance will be improved [7]. In order to take advantage of granular computing, research is being conducted in various fields. It is summarized in Table I.

Granular computing is useful to make it possible to segment and to analyze time-series sensor data that cannot be defined in advance. In addition, it can contribute to reduce the preprocessing cost for each granule. Using the advantages of the granular computing, we propose the

doi: 10.18178/ijmlc.2020.10.6.998

framework of determining the intervals of the instance for multimodal sensor data.

TABLE I: RESEARCH OF GRANULAR COMPUTING USING MULTIMODAL SENSOR DATA

Domain	Description	Ref.
Medical	- Granules: chunk of medical image pixels; take into detailed information and reflect the inherent spatial relation of the image	[10], [11]
Data Processing	- Make data process easier by dividing big data set with higher dimensions into relatively smaller data subsets.	[12], [13], [14]
Situation Awareness	- Adopt to solve open issues in the three levels of the Situation Awareness model	[15]
Learning	- Make information granules by collecting detailed data in same type. - Figure out data complexity problem by learning different type granule separately.	[16], [17]

III. Ensemble Instance Intervals Determination

A. Preliminaries

To divide time-series data collected using a multimodal sensor into instances, we must first convert it to structured tabular data, and then add the target label column, which is required for classification, to the tabular data. However, it is very hard to determine the intervals of the transformed tabular data because it is difficult to find the determined decision boundaries of them. To reduce the difficulty, we attempt to apply granular computing to the transformed tabular data. To do so, it is necessary to determine the data granularity representing the granule and granule. It is defined as follows.

Let $G=\{G_{ln} | l \in \{0,1,2,3\}, n \in \mathbb{N}\}$ be a collection of layers composing the hierarchical granularities. So, each layer is composed of data granules, which is defined as a class or a target label to be classified. If the concept of data granule and the data granularity are applied to the operating data of the water purifier, it can be schematized as depicted in Fig. 1. From now on, all discussions focus on the operating data of the water purifier.

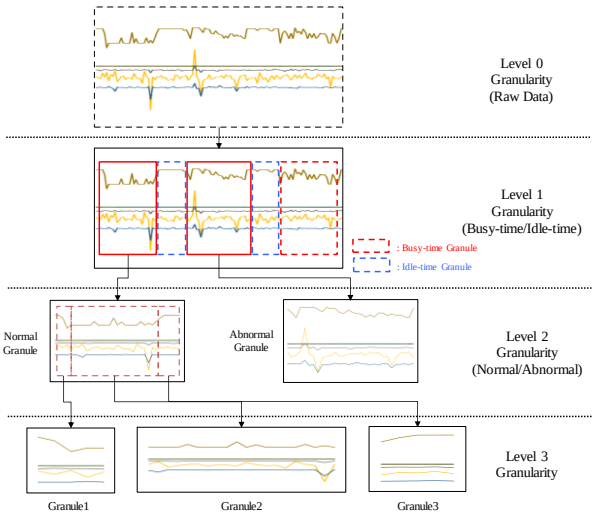

Fig. 1. Data granularity of the multimodal sensor data.

B. Overall Framework

As depicted in Fig. 2, the framework for determining the interval of an instance consists of three modules: Idle-time Granule Deletion Module, Abnormal Granules Deletion Module, and Level 3 Granularity Classifiers Learning Module.

1) Idle-time granule deletion module

This module is conducted to find a model that can distinguish between busy-time data granule and idle-time data granule among multimodal sensor data. To do this, we analyzed the characteristics of the two granules. As a result, we can see that the variance of the busy-time granule is larger than that of the idle-time granule. However, it is difficult to prove the applicability of the characteristics of the granules to all sensors and all data fields. So, we select the sensors and the data fields to which the characteristics can be applied. Overall procedure is as follows.

Step 1. Data normalization We perform normalization for each data field to compare the variance of sensor data measured with different measurements.

Step 2. Instance Segmentation We divided the normalized tabular data according to the target label that is appended. At this time, the instance i ($i_{t_i}^{(1,a)}$) of a^{th} granule in granularity G_{1n} is defined as the tabular data that is included in interval in which the same target label value is repeated.

$$i_{t_i}^{(1,a)} = (df_{t_i}^{(1)} \quad \dots \quad df_{t_i}^{(k)}) \text{ where } a \in \mathbb{N} \tag{1}$$

where $df_{t_i}^{(k)}$ is a column vector of k^{th} data field that is collected during time interval $t_i = \lfloor t_{is}, t_{if} \rfloor$ t_{is} is starting time of instance i and t_{if} is ending time of instance i.

Step 3. Calculation of Variance Average To identify the occurrence patterns of the instances, we calculated the variance of the data fields of the instances and then converted them into vectors of constant size. The variance vector of the data fields ($var(i_{t_i}^{(1,a)})$) and Average of variance of a^{th} granule (AoV_a) are as follows.

$$var(i_{t_i}^{(1,a)}) = (v_{i1}, \dots, v_{ik}) \tag{2}$$

$$\text{AoV}_a = \left(\frac{\sum_{i=1}^{N(a)} v_{i1}}{N(a)}, \dots, \frac{\sum_{i=1}^{N(a)} v_{ik}}{N(a)} \right) \tag{3}$$

where $v_{ik} = var\left(df_{t_i}^{(k)}\right)$ and $N(a)$ is the number of instances in a^{th} granule.

Step 4. Top-K method for data field selection We conducted Hadamard division to identify the top-k data fields using AoV of the busy-time granule and AoV of the idle-time granule [8]. Based on the results of Hadamard division, we select the top-k data fields.

Step 5. Features matrix for input vector generation $\alpha \times k$ features matrix is generated using top-k data fields (α is the number of features to be generated). As a next, the

features matrix is converted into $\alpha k \times 1$ input vector. As a result, the input vector is generated as many as the total number of instances of G_{1n} .

Step 6. Support Vector Machine learning The appropriateness of the instance interval to be determined by applying the concept of granularity will be verified in the last module. So, we concluded that it is necessary to learn high-order data faster than classification performance at Level 1 granularity. It is the reason why we select the SVM as Level 1 learner.

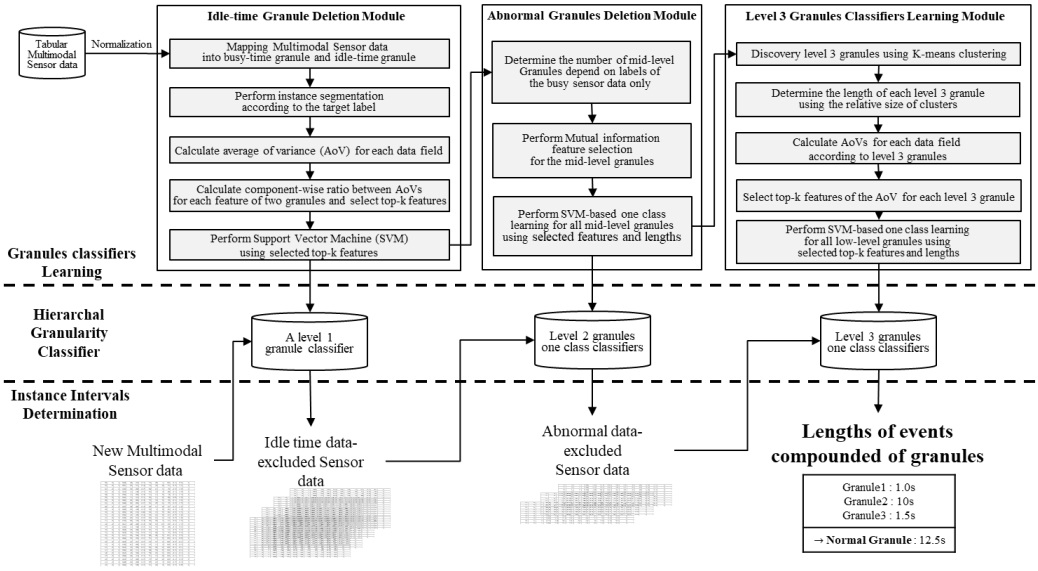

Fig. 2. Overall framework of ensemble instance intervals determination method.

Finally, we save the learned SVM model and the average of the lengths of the instances of a^{th} granule ($d_a = \frac{\sum_{i \in a} |t_i|}{\mathrm{N}\left(a^{th}\text{granule}\right)}$).

2) Abnormal granules deletion module

The purpose of this module is to classify normal granules and abnormal granules using only busy-time granule data and finally to remove abnormal granules. The process of removing the abnormal granules is similar to the previous module except that the data field is determined using Mutual Information (MI) instead of AoV. The reason for using MI instead of AoV is that both the normal granules and the abnormal granules data have a similar distribution. We generate the feature matrix and input vectors using the two granules with the highest information gain among all the granules of Level 2 and the data fields contained in these granules. As a next, we have found a model that removes the abnormal granules by learning one-class SVM models using these features and sizes of each granule. As in the previous step, one-class SVM model of b^{th} granule $\left(b \in \mathbb{N}\right)$ of granularity G_{2n} and the average of instance lengths of b^{th} granule (d_b) are stored.

3) Level 3 granules discovery module

This module is conducted to find a model that can distinguish among Level 3 granules in a normal granule. To do this, we performed clustering on a normal granule to find start-granule, several mid-granules, and end-granule, which are the minimum chunks of the data. The processing steps are as follows.

Step 1. K-means clustering for the normal granules Unlike level 1 and 2 granules, the target labels of level 3 granules are not defined in advance. To determine the number of the target labels, we adopted k-means clustering method on the instance $\left(i_{t_i}^{(2,b)}\right)$ a normal granule. This process was repeated for all kinds of the normal granules. After the clustering, $i_{t_i}^{(2,b)}$ as partitioned to several $i_{t_i}^{(3,c)}$ which is represented as follows.

$$i_{t_i}^{(2,b)} = \bigcup_{c \in b} i_{t_i}^{(3,c)}, \text{ where } i_{t_i}^{(3,c)} \text{ are disjoint for all c} \quad (4)$$

By time axis, partitioned $i_{t_i}^{(3,c)}$ were mapped to start-granule, mid1-granule, mid2-granule, …, and end-granule.

Step 2. Determination of Granule Size To determine the length of the actual instance using the union of level 3 granules, the length level 3 granules had to be determined. The length of c^{th} granule in b^{th} granule $(d_c^{(b)}, \forall c \in \mathbb{N})$ were calculated using relative size of c^{th} granule in granularity G_{3n} and the average length of instances in b^{th} granule.

$$d_c^{(b)} = d_b \times \frac{|\text{cluster} \forall c|}{\sum_{c \in b} |\text{cluster} \forall c|} \quad (5)$$

Step 3. Top-K method for data field selection We calculated AoV for all granules in level 3. However, Hadamard division is not performed, because we assume that level 3 data was decomposed sufficiently small.

Step 4. Feature matrix for input vector generation Feature matrices were created in a similar manner as before.

As a result, the input vectors with a size of αk_c were generated for each granule of level 3.

Step 5. Support Vector Machine Learning Finally, one-class SVMs learning is performed using the input vectors and their target labels corresponding to each level 3 granule.

Finally, we store the length $\left(d_c^{(b)}\right)$ and feature sets of the c^{th} granule along with the learned SVM models.

C. Instance Intervals Determination

The learned SVM models in each module is sequentially applied to the data when new data is created. The intervals are determined by finding the starting-points and end-points of the instances. Through this process, it is judged whether level 3 granules are sequentially detected. As a result, the intervals of the instances can be verified by the detected sequences. The process is shown in Fig. 3.

IV. Performance Evaluation

In order to demonstrate the superiority of the proposed method, we performed two experiments using the following sensors and features (Table II). The multimodal sensor integrates the sensors shown in Table II, which was installed in a water dispenser to collect water data.

MSD_{new} : New Multimodal Sensor data
CL_g: Classifiers of g^{th} granule, where $g \in \{a, b, c\}$
BG : Busy-time granule set
NG : Normal granule set
LG_c: c^{th} granule set of Level 3 Granularity
f_g, d_g: Feature set and Length of g^{th} granule
$t_{is}^{(3,c)}$: Starting point of c^{th} granule of Level 3 Granularity

Begin Instance Intervals Determination Process
  $TD \leftarrow Standandarization(MSD_{new})$
  For a^{th} granule **in** Level 1 Granularity
    $BG \leftarrow$ Idle − time Granule Deletion using $CL_a(TD, f_a, d_a)$
  End For
  For b^{th} granule **in** Level 2 Granularity
    $NG \leftarrow$ Abnormal Granule Deletion $CL_b(BG, f_b, d_b)$
  End For
  For c^{th} granule **in** Level 3 Granularity
    $LG_c \leftarrow$ Level 3 Granularity Discovery using $CL_c(NG, f_c, d_c)$
  End For
  Interval = []
  For all $L3G_c$
    If For all c
      $t_{is}^{(3,c)} < t_{is}^{(3,c+1)}$
      Interval. append(LG_c)
    Else
      Continue
    End If
  End For
  Return Interval
End Process

Fig. 3. Algorithm of instance interval determination.

TABLE II: Summary of Sensors and Features

Sensor	Type of Features
Stopwatch	Time
Hygrometer	Ambient Temperature, Ambient Humidity
Accelerometer-Gyrosensor	(x axis, y axis, z axis) of Accelerometer, (x axis, y axis, z axis) of Gyrosensor, Device temperature
Ultrasonic	Distance
Infrared	Temperature of surface, Temperature of target
Audio sensor	Volume

The metadata of the data collected using the multimodal sensor is as follows. The data collection period of the multimodal sensor was 0.5hz, the total operation time of the sensor was 6,930 seconds, 304 watering operations, and 307 waiting operations. The target labels of the data consist of 280 normal watering, 24 over watering, and 307 idle states. We mapped the normal watering, over watering to busy-time granule and the idle state to idle-time granule. The programming language used in the experiment was python 3.7, and the data analysis packages were numpy, scikit-learn, and pandas. The contents of the experiment are as follows.

Experiment 1: Comparison of data fields (features of sensors) between AoV method and filter method

To verify the validity of the selected top-k data field using the proposed AoV, we compared each top-k data field from AoV and filter method. As a result, there was no significant difference between the two methods. It meant that AoV can guarantee the performance of the filter method.

Experiment 2: Comparison of complexity between AoV method and filter method

We compared the complexity of the filter method with the complexity of the AoV method in the second experiment. In general, the complexity of the filter method is $n!\times O(n)$ more than $O(n^3)$ while the complexity of AoV is only $O(n^2)$ with the number of data field n. The result of comparisons of computation time as k increases in Top-k is shown in Fig. 4.

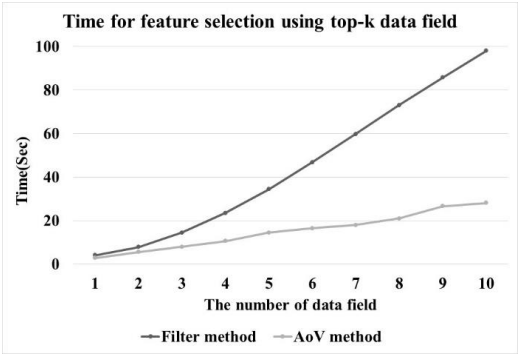

Fig. 4. Comparison of computational time between AoV and filter.

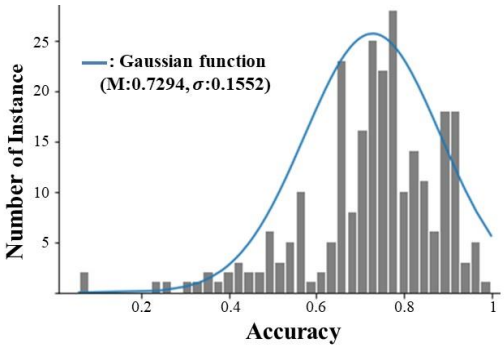

Fig. 5. Accuracy distribution using Jaccard's distances.

Experiment 3: Accuracy of granular computing-based predicted instance intervals

We performed experience to check the accuracy of the predicted instance interval obtained by the granular computing-based method. Our method found 258 instances out of 280 normal watering instances without appending the

target labels. We calculated 258 number of Jaccard's distances [9] between the predicted intervals and the actual interval. The result is shown in Fig. 5.

We checked that the distribution of the accuracy was a normal distribution using normality test with p-value 0.006. It followed as normal distribution of which mean was 0.7294 and standard deviation was 0.1552. Since 94.98% of the data was distributed in the range of 95% confidence interval [0.4252, 1.03], the predicted interval accuracy was 73%.

V. Conclusion

We proposed a method to determine the interval of instances that can be classified from normal granule of time series data by using granular computing. We also reduced the computational burden using different feature selection methods according to the granularity to handle multimodal sensor data of various kinds and volume.

The contribution of our paper is as follows. First, we applied the granular computing to the time series data and determined the instance intervals which were difficult to find by the conventional method. This is a major development of the existing research. Second, we could find out the deterministic instance length by finding the starting point and finishing point of the instance.

The limitations are as follows. Because our method used the supervised learning, it has disadvantage that it cannot be used if the target value of normal, abnormal, and idle is not attached. It is also impossible to classify in real time because we do not know appropriate collection period to determine the type and length of an instance. Therefore, in future studies, we will research with semi-supervised learning method which is used when target value does not attach entirely. Also, we will find optimal collection period for real time classification.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Jaewoong Kang, Wooseong Yang, and Mye Sohn wrote the paper; Jaewoong Kang, and Mye Sohn supposed and made the overall framework; Jaewoong Kang, and Mye Sohn draw the figures and mada the tables; Jaewoong Kang, and Wooseong Yang set the experimental environment and collected the water purifier data; Jaewoong Kang, and Mye Sohn analyzed the experimental results; Jaewoong Kang did a computer coding for experiments; Wooseong Yang wrote the algorithm of instance interval determination; Wooseong Yang surveyed the related works and previous works; Mye Sohn checked the overall paper.

References

[1] C. A. Ronao and S. B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Systems with Applications*, pp. 235-244, vol. 59, 2016.

[2] J. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, "Deep convolutional neural networks on multichannel time series for human activity recognition," presented at the Twenty-Fourth International Joint Conference on Artificial Intelligence, June 2015.

[3] Y. Y. Yao, "Granular computing: Basic issues and possible solutions," in *Proc. the 5th Joint Conference on Information Science*s, 2000, pp. 186-189, vol. 1.

[4] N. Srivastava and R. R. Salakhutdinov, "Multimodal learning with deep boltzmann machines," *Advances in Neural Information Processing Systems*, pp. 2222-2230, 2012.

[5] M. Chen, S. C. Chen, M. L. Shyu, and K. Wickramaratna, "Semantic event detection via multimodal data mining," *IEEE Signal Processing Magazine*, pp. 38-46, vol. 23, 2006.

[6] J. T. Yao, A. V. Vasilakos, and W. Pedrycz, "Granular computing: Perspectives and challenges," *IEEE Transactions on Cybernetics*, pp. 1977-1989, vol. 43, 2013.

[7] A. Bargiela and W. Pedrycz, "The roots of granular computing," in *Proc. IEEE International Conference on Granular Computing*, 2006, pp. 806-809.

[8] R. A. Horn, "The hadamard product," in *Proc. Symp. Appl. Math*, 1990, pp. 87-169.

[9] M. Levandowsky and D. Winter, "Distance between sets," *Nature*, p. 34, 1971.

[10] A. Hassanien, A. Abraham, J. Peters, G. Schaefer, and C. Henry, "Rough sets and near sets in medical imaging: A review," *IEEE Transactions on Information Technology in Biomedicine*, pp. 955-968, 2009.

[11] N. Senthilkumaran and R. Rajesh, "Brain image segmentation using granular rough sets," *International Journal of Arts and Sciences*, pp. 69-78, 2009

[12] M. Nair, A. Chadawar, and A. Bhosle, "An efficient divide and conquer approach for big data analytics in machine to machine communication," *International Journal of New Technology and Research*, pp. 439-453, 2016.

[13] J. Zhou, L. Hu, F. Wang, H. Lu, and K. Zhao, "An efficient multidimensional fusion algorithm for IoT data based on partitioning," *Tsinghua Science and Technology*, pp. 369-378, 2013.

[14] J. T. Yao, A. V. Vasilakos, and W. Pedrycz, "Granular computing: Perspectives and challenges," *IEEE Transactions on Cybernetics*, pp. 1977-1989, 2013.

[15] V. Loia, G. D'Aniello, A. Gaeta, and F. Orciuoli, "Enforcing situation awareness with granular computing: A systematic overview and new perspectives," *Granular Computing*, pp. 127-143, 2016.

[16] G. Peters and R. Weber, "DCC: A framework for dynamic granular clustering," *Granular Computing*, pp. 1-11, 2016.

[17] J. T. Yao and Y. Y. Yao, "A granular computing approach to machine learning," *FSKD*, pp. 45-47, 1993.

Jaewoong Kang is in a Ph. D. course in the Department of Industrial Engineering at Sungkyunkwan University. He received his bachelor's degree from Sungkyunkwan University. His main interests CNN-based deep learning, pattern recognition, and machine learning.

Wooseong Yang is in MS course in the Department of Industrial Engineering at Sungkyunkwan University. He received his bachelor's degree from Sungkyunkwan University. His main interests are machine learning, deep learning and pattern recognition.

Mye M. 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 machine learning, ontology, Web-of-Things, Semantic Web, and so on.