(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
31 October 2019 (31.10.2019)

WIPO | PCT

(10) International Publication Number
WO 2019/209059 A1

(51) International Patent Classification:
G06N 20/00 (2019.01) *H04L 29/08* (2006.01)

(21) International Application Number:
PCT/KR2019/005034

(22) International Filing Date:
25 April 2019 (25.04.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
201841015708 25 April 2018 (25.04.2018) IN
201841015708 24 April 2019 (24.04.2019) IN

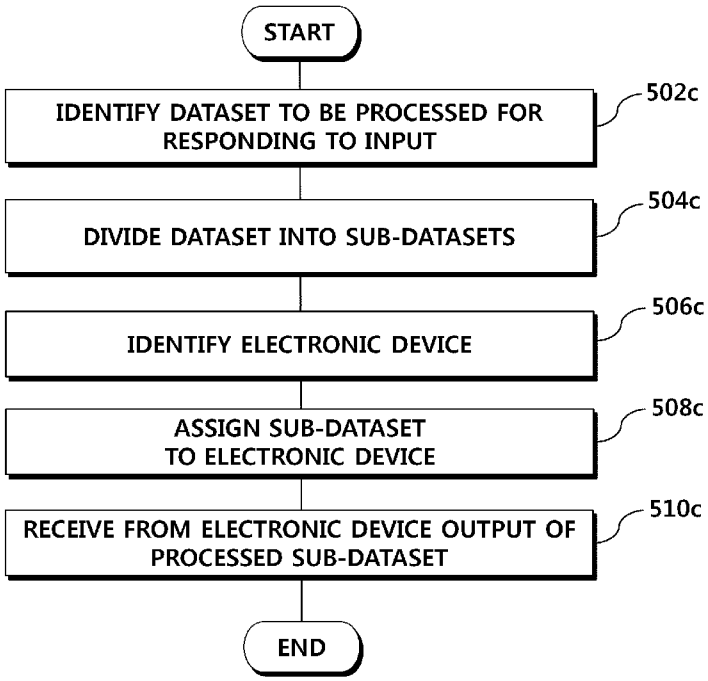
(71) Applicant: **SAMSUNG ELECTRONICS CO., LTD.** [KR/KR]; 129, Samsung-ro, Yeongtong-Gu, Suwon-Si, Gyeonggi-do 16677 (KR).

(72) Inventors: **SINGLA, Kushal**; #2870, Pheonix Building, Bagmane Constellation Business park, ORR, Doddanekundi Circle, Bangalore,Karnataka, 560037 (IN). **BOSE, Joy**; #2870, Pheonix Building, Bagmane Constellation Business park, ORR, Doddanekundi Circle, Bangalore,Karnataka, 560037 (IN). **KATARIYA, Sharvil Manish**; #2870, Pheonix Building, Bagmane Constellation Business park, ORR, Doddanekundi Circle, Bangalore,Karnataka, 560037 (IN).

(74) Agent: **Y.P.LEE, MOCK & PARTNERS**; 12F Daelim Acrotel, 13 Eonju-ro 30-gil, Gangnam-Gu, Seoul 06292 (KR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA,

(54) Title: MACHINE LEARNING ON A BLOCKCHAIN



(57) Abstract: According to an embodiment, there is provided an electronic device comprising: a memory storing instructions; and at least one processor configured to execute the instructions to: in response to an input, identify a dataset to be processed for responding to the input; divide the dataset into a plurality of sub-datasets; identify at least one electronic device which processes at least one sub-dataset; assign the at least one sub-dataset to the at least one electronic device to process the at least one sub-dataset; and receive from the at least one electronic device at least one output of the processed at least one sub-dataset to generate a response to the input.

WO 2019/209059 A1

[Continued on next page]

SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

Description

Title of Invention: MACHINE LEARNING ON A BLOCKCHAIN

Technical Field

[1] Embodiments herein relate to processing information, and more particularly to processing information for Internet of Things (IoT) devices by using a distributed ledger network.

Background Art

[2] In general, an IoT device generates streams of data that need to be processed to learn a user's behavioral patterns or routines by using a machine learning (ML) procedure. However, it is difficult to use the ML procedure on devices such as IoT sensors, because such sensors do not have enough memory and processing capability to run the ML procedure thereon. To work around this problem, data may be sent to a centralized server having an ML model as illustrated in FIGS. 1 and 2, but it is expensive to maintain the server, and may compromise the user's privacy. If the ML model is compressed to run on the IoT sensors, accuracy of prediction may decrease.

Disclosure of Invention

Solution to Problem

[3] According to an embodiment, there is provided an electronic device comprising: a memory storing instructions; and at least one processor configured to execute the instructions to: in response to an input, identify a dataset to be processed for responding to the input; divide the dataset into a plurality of sub-datasets; identify at least one electronic device which processes at least one sub-dataset; assign the at least one sub-dataset to the at least one electronic device to process the at least one sub-dataset; and receive from the at least one electronic device at least one output of the processed at least one sub-dataset to generate a response to the input.

Brief Description of Drawings

[4] The above and other advantages of embodiments will be better understood from the following description with reference to the accompanying drawings, in which:

[5] FIG. 1 is a diagram for explaining a machine learning (ML) model with a centralized server;

[6] FIG. 2 is a diagram illustrating communications between a server and IoT devices;

[7] FIG. 3 is a schematic view illustrating IoT devices according to an embodiment;

[8] FIG. 4a is a block diagram of an IoT device according to an embodiment;

[9] FIG. 4b is a block diagram of a processor included in the IoT device according to an embodiment;

[10] FIG. 5a is a flowchart illustrating a method of processing the ML data on the dis-

tributed ledger network according to an embodiment;

[11] FIG. 5b is a flowchart illustrating a method of processing the ML data by using a smart contract according to an embodiment;

[12] FIG. 5c is a flowchart illustrating a method according to an embodiment;

[13] FIG. 6a is a diagram for explaining washing machine personalization according to an embodiment;

[14] FIG. 6b is a diagram for explaining the use of a smart contract according to an embodiment;

[15] FIG. 7 is a diagram for explaining a navigation system according to an embodiment;

[16] FIG. 8 is a diagram for explaining a virtual assistance application using a server;

[17] FIG. 9 is a diagram for explaining an example scenario of the virtual assistance application according to an embodiment;

[18] FIG. 10 is a diagram for explaining a guest verification procedure;

[19] FIG. 11 is a diagram for explaining a guest verification procedure according to an embodiment;

[20] FIG. 12 is a diagram for explaining a guest verification procedure according to an embodiment;

[21] FIG. 13 is a diagram for explaining smart garbage collection according to an embodiment;

[22] FIG. 14 is a diagram for explaining a smart home according to an embodiment;

[23] FIG. 15 is a diagram for explaining smart fuel saving in a car according to an embodiment;

[24] FIG. 16 is a diagram for explaining division of a machine learning model via forking according to an embodiment;

[25] FIG. 17 is a diagram for explaining an ML prediction using a blockchain according to an embodiment;

[26] FIG. 18 is a diagram for explaining a random forest on a blockchain according to an embodiment;

[27] FIG. 19 is a diagram for explaining a decision tree trained model working as a smart contract on a node in the blockchain according to an embodiment;and

[28] FIG. 20 is a diagram for explaining a deep neural network (DNN) on blockchain according to an embodiment.

Best Mode for Carrying out the Invention

[29] Aspects of the disclosure are to address at least the above-mentioned problems and/or disadvantages and to provide at least the advantages described below.

[30] According to an embodiment, there is provided an electronic device comprising: a memory storing instructions; and at least one processor configured to execute the in-

structions to: in response to an input, identify a dataset to be processed for responding to the input; divide the dataset into a plurality of sub-datasets; identify at least one electronic device which processes at least one sub-dataset; assign the at least one sub-dataset to the at least one electronic device to process the at least one sub-dataset; and receive from the at least one electronic device at least one output of the processed at least one sub-dataset to generate a response to the input.

[31] According to an embodiment, there is provided an electronic device comprising: a memory storing instructions; and at least one processor configured to execute the instructions to: receve from another electronic device a sub-dataset divided from a dataset which is identifed, at the other electronic device, for responding to an input; process the sub-dataset to produce an output; and transmit to the other electronic device the output of the processed sub-dataset to generate a response to the input.

[32] According to an embodiment, there is provided a method comprising: in response to an input, identifying a dataset to be processed for responding to the input; dividing the data into a plurality of sub-datasets; identifying at least one electronic device which processes at least one sub-dataset; assigning the at least one sub-dataset to the at least one electronic device to process the at least one sub-dataset; and receiving from the at least one electronic device an output of the processed at least one sub-dataset to generate a response to the input.

[33] According to an embodiment, there is provided a method comprising: receving from another electronic device a sub-dataset divided from a dataset which is identifed, at the other electronic device, for responding to an input; processing the sub-dataset to produce an output; and transmitting to the other electronic device the output of the processed sub-dataset to generate a response to the input.

[34] According to an embodiment, there is provided a computer program comprising instructions which, when the computer program is executed by at least one processor, cause the at least one processor to carry out one of the above methods.

[35] Accordingly, an embodiment provides a method for IoT devices to process data assistively in a distributed ledger network. The method includes receiving, by a first IoT device in the distributed ledger network, a query from a user. Further, the method includes identifying, by the first IoT device, the ML data to process the query. Further, the method includes dividing, by the first IoT device, the ML data into a plurality of sub-models. Further, the method includes identifying, by the first IoT device, a plurality of second IoT devices having capacity to process at least one sub-models of the ML data. Further, the method includes executing, by the first IoT device, a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models. Further, the method includes assigning, by the first IoT device, at least

one sub-models from the plurality of sub-models to at least one second IoT device in the blockchain-based IoT network based on the smart contract. Further, the method includes obtaining, by the first IoT device, an output data of the remaining sub-models of the ML data from each of the second IoT devices. Further, the method includes generating, by the first IoT device, a response for the query by processing an output data of the at least one sub-models processed by the first IoT device and the output of the remaining sub-models processed by the at least one second IoT device.

[36] In an embodiment, assigning the remaining sub-models to the at least one second IoT device in the blockchain-based IoT network includes identifying the at least one second IoT device for processing the remaining sub-models based on at least one of capacity, a neighbor IoT device of the first IoT device, and a priority set by a user, executing at least one smart contract between the first IoT device and the at least one identified second IoT devices for processing the remaining sub-models, and assigning the remaining sub-models to the at least one second IoT device for processing the remaining sub-models.

[37] In an embodiment, generating the response for the query by processing the output data of the at least one sub-model processed by the first IoT device and the output of the remaining sub-models processed by the at least one second IoT device includes aggregating each of the output data of the at least one sub-models processed by the at least one second IoT device and the output of the at least one sub-model processed by the first IoT device to generate a trained machine learning model, and generating the response for the query by applying the trained machine learning model on the query.

[38] In an embodiment, the output data of the first IoT device and the at least one second IoT device added as individual blocks in the distributed ledger network.

[39] In an embodiment, the at least one second IoT device processes the remaining sub-models according to terms and specifications of at least one smart contract.

[40] In an embodiment, the first IoT device links at least one smart contract to a blockchain infrastructure, where a private ledger and a public ledger of the blockchain infrastructure are used to verify the processing of the sub-models on the at least one second IoT device according to the smart contract.

[41] In an embodiment, identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the ML data comprises detecting, by the first IoT device, a user presence in the distributed ledger network, obtaining, by the first IoT device, a profile of the user, and identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the ML data based on the profile.

[42] In an embodiment, identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the sub-models

comprises detecting, by the first IoT device, context associated with the ML data in the distributed ledger network, and identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the ML data based on the context.

[43] In an embodiment, identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the ML data comprises detecting, by the first IoT device, at least two user's presence in the distributed ledger network, determining, by the first IoT device, a conflict to identify the plurality of second IoT devices having capacity to process at least one sub-models of the ML data in response at least two users presence in the distributed ledger network, and identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the ML data based on a setting.

[44] In an embodiment, the setting comprises at least one of a First come first serve (FCFS) setting, a Last come first serve (LCFS), and a voting based setting.

[45] In an embodiment, the trained machine learning model is generated by receiving the query from the first IoT device, selecting the at least one second IoT device from the plurality of IoT devices to process the received query based on capability of each of the IoT devices, dividing the query into a label data and a non-label data, identifying at least one of second IoT devices to process the label data and the non-label data, and aggregating each divided data result to generate the trained machine learning model.

[46] Accordingly, the embodiment herein provides a method for IoT devices to process data assistively in a distributed ledger network. The method includes receiving, by a first IoT device in the distributed ledger network, a query from a user. Further, the method includes identifying, by the first IoT device, non-machine learning (ML) data for processing the query. Further, the method includes dividing, by the first IoT device, the non-ML data into a plurality of sub-models. Further, the method includes identifying, by the first IoT device, a plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data. Further, the method includes executing, by the first IoT device, a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models. Further, the method includes assigning, by the first IoT device, at least one sub-models from the plurality of sub-models to the at least one second IoT devices from the plurality of second IoT devices based on the smart contract. Further, the method includes obtaining, by the first IoT device, an output data of the processed sub-models of the non-ML data from each of the second IoT devices. Further, the method includes generating, by the first IoT device, a response for the query by processing the output data of the at least one sub-models processed by the first IoT device and the output of the remaining sub-models

processed by the at least one second IoT device.

[47] In an embodiment, identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data comprises detecting, by the first IoT device, a user presence in the distributed ledger network, obtaining, by the first IoT device, a profile of the user, and identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data based on the profile.

[48] In an embodiment, identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the sub-models comprises detecting, by the first IoT device, context associated with the non-ML data in the distributed ledger network, and identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data based on the context.

[49] In an embodiment, identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data comprises detecting, by the first IoT device, at least two user's presence in the distributed ledger network, determining, by the first IoT device, a conflict to identify the plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data in response at least two users presence in the distributed ledger network, and identifying, by the first IoT device, the plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data based on a setting.

[50] In an embodiment, the trained machine learning model is generated by receiving the query from the first IoT device, selecting the at least one second IoT device from the plurality of IoT devices to process the received query based on capability of each of the IoT devices, dividing the query into a label data and a non-label data, identifying at least one of second IoT devices to process the label data and the non-label data, and aggregating each divided data result to generate the trained machine learning model.

[51] Accordingly, the embodiment herein provides an IoT device for processing data assitively in a distributed ledger network. The IoT device includes a processor coupled to a memory. The processor may receive a query from a user and determine a ML data to process the query. Further, the processor may divide the ML data into a plurality of sub-models. Further, the processor may identify a plurality of second IoT devices having capacity to process at least one sub-models of the ML data. Further, the processor may execute a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models. Further, the processor may assign remaining sub-models from the plurality of sub-models to at least one another IoT

device in the blockchain-based IoT network and obtain an output data of the remaining sub-models processed by the at least one another IoT device. Further, the processor may generate a response for the query by processing an output data of the at least one sub-models processed by the processor included in the IoT device and the output of the sub-models processed by the at least one another IoT device.

[52] Accordingly, the embodiment herein provides a first IoT device for processing data assistively in a distributed ledger network. The first IoT device comprises a processor coupled with a memory. The processor may receive a query from a user and identify non-machine learning (ML) data for processing the query. The processor may divide the non-ML data into a plurality of sub-models. The processor may identify a plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data. The processor may execute a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models. The processor may assign at least one sub-models from the plurality of sub-models to the at least one second IoT devices from the plurality of second IoT devices based on the smart contract. The processor may obtain an output data of the processed sub-models of the non-ML data from each of the second IoT devices. The processor may generate a response for the query by processing the output data of the at least one sub-models processed by the first IoT device and the output of the remaining sub-models processed by the at least one second IoT device.

Mode for the Invention

[53] The following description with reference to the accompanying drawings is provided to assist in a comprehensive understanding of various embodiments of the disclosure as defined by the claims and their equivalents. The following description includes various specific details to assist in that understanding but these are to be regarded as merely exemplary. Accordingly, those of ordinary skill in the art will recognize that various changes and modifications of the various embodiments described herein may be made without departing from the scope and spirit of the disclosure as defined in the appended claims. In addition, descriptions of well-known functions and constructions may be omitted for clarity and conciseness.

[54] It may be noted that to the extent possible, like reference numerals have been used to represent like elements in the drawings. Further, those of ordinary skill in the art will appreciate that elements in the drawings are illustrated for simplicity and may not have been necessarily drawn to scale. For example, the dimensions of some of the elements in the drawings may be exaggerated relative to other elements to help to improve understanding of aspects of the embodiments. Furthermore, the one or more elements

may have been represented in the drawings by various symbols, and the drawings may show only those specific details that are pertinent to understanding the embodiments of the disclosure so as not to obscure the drawings with details that will be readily apparent to those of ordinary skill in the art having benefit of the description herein.

[55] The terms and words used in the following description and claims are not limited to the bibliographical meanings, but, are merely used by the inventors to enable a clear and consistent understanding of the disclosure. Accordingly, it should be apparent to those skilled in the art that the following description of various embodiments of the disclosure is provided for illustration purpose only and not for the purpose of limiting the disclosure as defined by the appended claims and their equivalents.

[56] It is to be understood that the singular forms "a," "an," and "the" include plural referents unless the context clearly dictates otherwise. Thus, for example, reference to "a component surface" includes reference to one or more of such surfaces. Additionally, expressions such as "at least one of," when preceding a list of elements, modify the entire list of elements and do not modify the individual elements of the list. For example, the expression, "at least one of a, b, and c," should be understood as including only a, only b, only c, both a and b, both a and c, both b and c, or all of a, b, and c.

[57] The terms "include" and "comprise," as well as derivatives thereof, mean inclusion without limitation. The term "or," is inclusive, meaning and/or. The phrases "associated with" and "associated therewith," as well as derivatives thereof, may mean to include, be included within, interconnect with, contain, be contained within, connect to or with, couple to or with, be communicable with, cooperate with, interleave, juxtapose, be proximate to, be bound to or with, have, have a property of, or the like.

[58] Moreover, various functions described below may be implemented or supported by one or more computer programs, each of which may be formed from computer readable program code and embodied in a computer readable medium. The terms "application" and "program" refer to one or more computer programs, software components, sets of instructions, procedures, functions, objects, classes, instances, related data, or a portion thereof adapted for implementation in a suitable computer readable program code. The phrase "computer readable program code" includes any type of computer code, including source code, object code, and executable code. The phrase "computer readable medium" includes any type of medium capable of being accessed by a computer, such as read only memory (ROM), random access memory (RAM), a hard disk drive, a compact disc (CD), a digital video disc (DVD), or any other type of memory. A "non-transitory" computer readable medium excludes wired, wireless, optical, or other communication links that transport transitory electrical or other signals. A non-transitory computer readable medium includes media where data

may be permanently stored and media where data may be stored and later overwritten, such as a rewritable optical disc or an erasable memory device.

[59] The term "unit", "manager", "engine", or "device" may refer to a unit which processes at least one function or operation and may be implemented by hardware, software, or a combination of hardware and software.

[60] Accordingly, embodiments disclose a method for processing data on. The data may include ML data and non-ML data. The data may be processed in a distributed ledger network. The method may include receiving, by a first device in the distributed ledger network, a query from a user. The first device may be a first IoT device. Further, the method may include determining, by the first IoT device, data (e.g., ML data) to process the query. Further, the method may include dividing, by the first IoT device, the ML data into a plurality of sub-models. The sub-models may be ML sub-models. Further, the method may include identifying, by the first IoT device, a plurality of second devices having capacity to process at least one sub-models of the ML data. The second devices may be second IoT devices. Further, the method may include executing, by the first IoT device, a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models. Further, the method may include assigning, by the first IoT device, sub-models from the plurality of sub-models to at least one second IoT device in the blockchain-based IoT network. The sub-models assigned to the at least one second IoT device may be ones remained after assigning at least one sub-model to the first IoT device. Further, the method may include obtaining, by the first IoT device, an output data of the sub-models processed by the at least one second IoT device. Further, the method may include generating, by the first IoT device, a response for the query by processing an output data of the at least one sub-model processed by the first IoT device and the output of the sub-models processed by the at least one second IoT device.

[61] According to an embodiment, each IoT device may automatically divide the ML model into smaller parts to send the divided parts to surrounding IoT devices for computation as per their capabilities, and merge results of the computations. The division of the ML model may be performed by forking of smart contracts. Distributed and secure Blockchain storage may be used to store the ML model and the results of the computation. This results in processing the ML data on the distributed ledger network in a cost effective, secure and accurate manner.

[62] According to an embodiment, a secure system for personalization of the IoT devices in a smart home may be provided to a user based on using the blockchain. The distributed ledger network may store the device profiles of individual users in the shared household. According to an embodiment, automatic personalization of each device

may be accomplished in response to detecting a particular user entering the room where the device is located, in a secure and cost effective manner. In an example, when first user has a preferred brightness or luminescence value for a smart bulb, and the first user enters a room, then the smart bulb may detect the first user presence in the home, obtain the first user`s luminescence setting preference from the blockchain, and automatically set its brightness value based on the first user's luminescence setting. When a different user (i.e., second user) is detected in the same room prior to the first user, then an appropriate conflict resolution strategy such as computing the average or First Come First Serve (FCFS) may be applied.

[63] FIG. 3 is a schematic view illustrating IoT devices according to an embodiment. IoT devices may process data based on their capability.

[64] The distributed ledger network 1000 includes a first IoT device 100a and a set of IoT devices 100b-100n. The first IoT device 100a may communicate with the set of IoT devices 100b-100n. The IoT devices 100a-100n may be, for example but not limited to a smart TV, a smart bulb, a smart phone, a smart Air conditioner (AC), a smart fridge, a door sensor or the like. The IoT devices 100a-100n may communicate with each other. The IoT devices 100a-100n may be in the same home network, but is not limited thereto. The distributed ledger network may be, for example, but not limited to a blockchain-based Internet of Things (IoT) network, a graph-based distributed ledger system such as the IOTA (based on Tangle), Hashgraph (based on gossip protocol) or the like.

[65] In an embodiment, the first IoT device 100a may receive a query of a user and determine a ML data or non-ML to process the query. The query of the user may be receved from the user's personal device. The ML data may include components of a core machine learning model, for example weights learnt by the model/procedure. The ML Data may not be provided to the user externally but generated in the procedure of running its core logic. Data other than the ML Data may be referred to as non-ML data. The non-ML Data may be, for example, input data like device logs, images, text, etc. The non-ML data may be given as inputs for the machine learning model, e.g., an image having a cat/dog , an image not having a cat/dog, the labels 'cat'/'dog' along with the images may be categorized into the non-ML data.

[66] In an example, the weights and hyperparamaters may be learnt by processing the Decision Trees, Random Forests, Neural Network, etc The non-ML Datamay be fed as inputs for the ML Model (even labels for ML task may fall under the Non-ML data). The non-ML data may include images, text, news articles, app category, app metadata, and app usage data, but is not limited thereto.Further, the first IoT device 100a may divide the ML data/non-ML data into a plurality of sub-models. The ML data may include a ML model or components of the ML model. Further, the first IoT device

100a may identify a plurality of IoT devices 100b-100n. The first IoT device 100a may identify the plurality of IoT devices 100b-100n based on capacity of the plurality of IoT devices 100b-100n to process at least one sub-models of the ML data/non-ML data. Further, the first IoT device 100a may generate a smart contract between the first IoT device 100a and the second IoT device 100b from the plurality of IoT devices 100b-100n for processing the at least one sub-models of the plurality of the sub-models. The smart contract may be executed.

[67] Further, the first IoT device 100a may assign sub-models from the plurality of sub-models to at least one of the IoT devices 100b-100n. The sub-models assigned to the at least one IoT device may be ones remained after assigning at least one sub-model to the first IoT device. The assignment may be performed based on the smart contract betweeh the first IoT device 100a and the IoT devices 100b-100n. The first IoT device 100a may obtain an output data of the sub-models processed by the at least one of the IoT devices 100b-100n. Further, the first IoT device 100a may generate a response to the query by processing an output data of the at least one sub-models processed by the IoT device 100a and the output of the sub-models processed by the at least one of the IoT devices 100b-100n.

[68] In an embodiment, the first IoT device 100a may assign the sub-models to the at least one of the IoT devices 100b-100n in the distributed ledger network by identifying the at least one of the IoT devices 100b-100n based on capacity of the IoT devices 100b-100n. In an embodiment, the at least one of the IoT devices 100b-100n may be identified based on a neighbor IoT device 100b-100n of the IoT device 100a. In an embodiment, the at least one of the IoT devices 100b-100n may be identified based on a priority set by a user. In an embodiment, at least one smart contract between the IoT device 100a and the identified at least one of the IoT devices 100b-100n may be executed for processing the sub-models.

[69] In an embodiment, the first IoT device 100a may aggregate each output of the sub-models processed by the at least one of the IoT devices 100b-100n and the output of the at least one sub-model processed by the IoT device 100a to generate a trained machine learning model. The trained ML model may be used to generate the response to the query of the user.

[70] In an embodiment, the trained machine learning model may be generated by receiving the query from the first IoT device 100a, identifying the at least one IoT device from the plurality of IoT devices 100b-100n to process the received query based on capability of each of the IoT devices, dividing the query into a label data and a non-label data, causing the at least one IoT device to process the label data and the non-label data, and aggregating each output from each divided data.

[71] The labelled data may include a set of input data that is mapped to one or more

labels. For example, when predicting whether an image has dog or cat or both, each image may be labelled with 'dog', 'cat', 'none' or 'both'.

[72] The unlabeled data may include a set of input data which is not mapped to one or more labels. In order to find out similar devices in a smart home environment without explicitly labeling each device, an ML models may be built to find similar devices based on users' usage of each device, which may be helpful to replace a device with another one. In this case, "labels" for a device, for example, 'refrigerator', is not provided, but a corresponding device may be found by the ML models based on the users' usage of devices.

[73] Further, the first IoT device 100a may generate the response for the query by processing the output data of the at least one sub-model processed by the IoT device 100a and the output of the sub-models processed by the at least one of IoT devices 100b-100n.

[74] In an embodiment, the output data of the IoT device 100a and the at least one of the IoT devices 100b-100n may be added as individual blocks in the distributed ledger network. In an embodiment, the at least one of the IoT devices 100b-100n may process the sub-models according to terms and specifications of at least one smart contract between the first IoT device 100a and the at least one of the IoT devices 100b-100n.

[75] In an embodiment, the IoT device 100a may link the at least one smart contract to a blockchain infrastructure, and a private ledger or a public ledger of the blockchain infrastructure may be used to verify the processing of the sub-models on the at least one of the IoT devices 100b-100n according to the smart contract.

[76] 1. In an embodiment, the IoT device 100a may identify the plurality of IoT devices 100b-100n to process at least one sub-models of the ML data or non-ML data based on detecting a user. The user may be detected in the distributed ledger network 1000. A profile of the user may be used to identify the plurality of IoT devices 100b-100n to process at least one sub-models of the ML data or non-ML data.

[77] In an embodiment, the IoT device 100a may identify the plurality of IoT devices 100b-100n to process at least one sub-models of the ML data or non-ML data based on detecting context associated with the ML data or non-ML in the distributed ledger network 100.

[78] In an embodiment the first IoT device 100a may identify the plurality of IoT devices 100b-100n to process at least one sub-models of the ML data or non-ML data based on a setting when at least two users are detected. The at least two users may be detected in the distributed ledger network 1000. The first IoT device 100a,

[79] In an embodiment, the setting may include at least one of a first come first serve (FCFS) setting, a last come first serve (LCFS), and a voting based setting.

[80] In an embodiment, the first IoT device 100a may store a user data on a blockchain

with an Interplanetary file system (IPFS), and the ML models may be stored on the the blockchain with IPFS. The IPFS may be implemented on multiple IoT devices 100a-100n consisting of the blockchain. The IPFS may be treated as one distributed file system used in the blockchain. The hash of the address of the IPFS file may be stored on the blockchain's of the IoT devices 100a-100n. The machine learning procedure of prediction/training (implemented by blockchain smart contracts) may be divided into sub-procedures, and distributed to nearby devices, results of the ML procedures may be aggregated. The sub-procedures may be referred to as sub-contracts.

[81] In an example, the blockchain may be a public blockchain system (e.g., Ethereum or the like). The blockchain may store the data in the IPFS. In another example, the blockchain may be a private blockchain system and the blockchain may store the data in an IoT hub or IoT devices on a network.

[82] Using Ethereum: In an example, the first step may be computing the IoT device profile on the smart hub for each user and each device. The device profile may be updated periodically, for example monthly. The user preferences may be assumed to change not too often. The IoT devices 100a-100n may send the usage data to the IoT hub which analyses the data and computes the device profile. An example of the IoT hub may be SmartThings, but is not limited thereto. The person ordinary skill in the art, they may use other tool (e.g., Truffle, Ganache, or the like) to perform the same operation.

[83] The computed profile may be stored on the blockchain using the IPFS, which in turns generates a hash. This generated hash may be stored on the Ethereum network. The transaction hash may be stored securely on the IoT hub.

[84] 2. Using the private blockchain system and store data in the hub: In another example, a block mining calculation (with a reduced difficulty level) may be performed in the IoT hub (e.g., SmartThings) consisting of an IoT environment, rather than in a remote server. In this architecture, the IoT hub may compute the device profiles and store them in a private blockchain system. According to an embodiment, protection of privacy may be strengthened, because the preferences of the users for different device settings are not stored on any external server, and the preferences of the users are stored on the IoT hub.

[85] Using IOTA: In another example, the IOTA may be used. Using IOTA, each IoT device 100a-100n or the IoT hub may function as parts of the blockchain itself, and may have the capability of mining data. Each device computes and stores its own user device profile for each of the users, rather than store the profiles for all the IoT devices 100a-100n in the smart house.

[86] The IOTA has a distributed ledger based on Tangle, which consists of nodes

connected in a web form instead of blockchain. The Tangle uses a Directed Acyclic Graph (DAG) for storing transactions. Any transaction is verified by two other closer nodes in the web, unlike blockchain where only miners verify the transaction. In Tangle, the transaction may be confirmed when the transaction acquires enough cumulative weight. Thus, technological resources may be shared locally in real time between IoT devices in a distributed network. The verification of transactions may be completed by the nearest nodes. The IOTA utilizes a hash function called Curl, which is quantum- immune. IOTA may be used to trade exact amounts of resources on demand as well as store data from sensor and data loggers securely after verification.

[87] FIG. 4a is a block diagram of an IoT device according to an embodiment, and FIG. 4b is a block diagram of a processor included in the IoT device according to an embodiment. In an embodiment, the IoT device 100 includes a processor 110, a communicator 120, and a memory 130. The processor 110 may receive the query from the user and determine the ML data to process the query. Further, the processor 110 may divide the ML data into the plurality of sub-models and process at least one sub-model from the plurality of sub-models based on the capability of the first IoT device 100a.

[88] Further, the processor 110 may assign remaining sub-models from the plurality of sub-models to at least one another IoT device 100b-100n in the blockchain-based IoT network and obtain the output data of the remaining sub-models processed by the at least one another IoT device 100b-100n. Further, the processor 110 may generate the response for the query by processing the output data of the at least one sub-models processed by the processor 110 included in the IoT device 100a and the output of the remaining sub-models processed by the at least one another IoT device 100b-100n.

[89] In an embodiment, the processor 110 may assign the remaining sub-models to the at least one another IoT device 100b-100n in the blockchain-based IoT network by identifying the at least one another IoT device 100b-100n for processing the remaining sub-models based on at least one of capacity, the neighbor IoT device 100b-100n of the IoT device 100a, and the priority set by the user, and executing at least one smart contract between the IoT device 100a and the at least one identified another IoT devices 100b-100n for processing the remaining sub-models.

[90] In an embodiment, the processor 110 may aggregate each of the output data of the at least one sub-models processed by the at least one another IoT device 100b-100n and the output of the at least one sub-model processed by the IoT device 100a to generate the trained machine learning model. Further, the processor 110 may generate the response for the query by processing the output data of the at least one sub-model processed by the processor 110 included in the IoT device 100a and the output of the remaining sub-models processed by the at least one another IoT device 100b-100n.

[91] In an embodiment, the processor 110 includes a ML data dividing unit, a ML-

sub-model processing unit, a ML-sub-model process assigning unit, and a response generation unit used for processing the ML model on the block -chain based IoT network.

[92] The communicator 120 is configured for communicating with the processor 110 to process the ML data on the blockchain-based IoT) network.

[93] The memory 130 also stores instructions to be executed by the processor 110. The memory 130 may include non-volatile storage elements. Examples of such non-volatile storage elements may include magnetic hard discs, optical discs, floppy discs, flash memories, or forms of electrically programmable memories (EPROM) or electrically erasable and programmable (EEPROM) memories. In addition, the memory 130 may, in some examples, be considered a non-transitory storage medium. The term "non-transitory" may indicate that the storage medium is not embodied in a carrier wave or a propagated signal. However, the term "non-transitory" should not be interpreted that the memory 130 is non-movable. In some examples, the memory 130 may be configured to store larger amounts of information than the memory. In certain examples, a non-transitory storage medium may store data that can, over time, change (e.g., in Random Access Memory (RAM) or cache).

[94] Although the FIG. 4a shows various hardware components of the IoT device 100a-100n but it is to be understood that other embodiments are not limited thereon. In another embodiment, the IoT device 100a-100n may include less or more number of components. Further, the labels or names of the components are used only for illustrative purpose and does not limit the scope of the invention. One or more components may be combined together to perform same or substantially similar function for IoT devices to process data assitively based on capability of the IoT devices in the distributed ledger network 1000.

[95] FIG. 4b shows various hardware components of the processor included in the IoT devices 100a-100n, according to an embodiment as disclosed herein. In an embodiment, the processor 110 includes a query analyser 110a, a data divider 110b, a capability identifier 110c, a data aggregator 110d, a smart contract executor 110e and a block-chain based response modeller 110e.

[96] The query analyser 110a may receive the query from the user and determine the ML data or non-ML to process the query. Further, the data divider 110b may divide the ML data or non-ML into the plurality of sub-models. Further, the capability identifier 110c may identify the plurality of IoT devices 100b-100n having capacity to process at least one sub-models of the ML data or non-ML data. A smart contract executor 100e may execute the smart contract between the first IoT device 100a and the at least one IoT device from the plurality of IoT devices 100b-100n for processing the at least one sub-models of the plurality of the sub-models.

[97] Further, the block-chain based response modeller 110f may assign remaining sub-models from the plurality of sub-models to at least one another IoT device 100b-100n and obtain the output data of the remaining sub-models processed by the at least one another IoT device 100b-100n. Further, the block-chain based response modeller 110f may generate the response for the query by processing the output data of the at least one sub-models processed by the processor 110 included in the IoT device 100a and the output of the remaining sub-models processed by the at least one another IoT device 100b-100n.

[98] In an embodiment, the block-chain based response modeller 110f may assign the remaining sub-models to the at least one another IoT device 100b-100n in the blockchain-based IoT network by identifying the at least one another IoT device 100b-100n for processing the remaining sub-models based on at least one of capacity, the neighbor IoT device 100b-100n of the IoT device 100a, and the priority set by the user, and executing at least one smart contract between the IoT device 100a and the at least one identified another IoT devices 100b-100n for processing the remaining sub-models.

[99] 4. In an embodiment, the block-chain based response modeller 110f may aggregate each of the output data of the at least one sub-models processed by the at least one another IoT device 100b-100n and the output of the at least one sub-model processed by the IoT device 100a to generate the trained machine learning model. Further, the block-chain based response modeller 110f may generate the response for the query by processing the output data of the at least one sub-model processed by the processor 110 included in the IoT device 100a and the output of the remaining sub-models processed by the at least one another IoT device 100b-100n.

[100] Although the FIG. 4b shows various hardware components of the processor 110 but it is to be understood that other embodiments are not limited thereon. In another embodiment, the processor 110 may include less or more number of components. Further, the labels or names of the components are used only for illustrative purpose and does not limit the scope of the invention. One or more components may be combined together to perform same or substantially similar function for IoT devices to process data in the distributed ledger network 1000.

[101] 5. FIG. 5a is a flowchart illustrating a method of processing the ML data on the distributed ledger network according to an embodiment, and FIG. 5b is a flowchart illustrating a method of processing the ML data by using a smart contract according to an embodiment.Steps 502a-514a may be processed by the processor 110.

[102] At 502a, the method includes receiving, by the first IoT device 100a in the distributed ledger network, the query from the user. At 504a, the method includes determining the ML data to process the query. At 506a, the method includes dividing the

ML data into the plurality of sub-models. At 508a, the method includes identifying the plurality of IoT devices having capacity to process at least one sub-models of the ML data. At 510a, the method includes executing the smart contract between the first IoT device 100a and the at least one IoT device from the plurality of IoT devices 100b-100n for processing the at least one sub-models of the plurality of the sub-models.

[103] At 512a, the method includes assigning the sub-models from the plurality of sub-models to at least one IoT device of the IoT devices 100b-100n in the blockchain-based IoT network. At 514a, the method includes obtaining the output data of the remaining sub-models processed by the at least one IoT device. At 516a, the method includes generating the response for the query by processing the output data of the at least one sub-models processed by the first IoT device 100a and the output of the remaining sub-models processed by the at least one IoT device.

[104] Steps 502b-516b may be processed by the processor 110. At 502b, the method includes receiving the query from the user. At 504b, the method identifying non- ML data for processing the query. At 506b, the method dividing the non-ML data into the plurality of sub-models.

[105] At 508b, the method identifying the plurality of IoT devices 100b-100n having capacity to process at least one sub-models of the non-ML data. At 510b, the method executing the smart contract between the first IoT device 100a and the at least one IoT device from the plurality of IoT devices 100b-100n for processing the at least one sub-models of the plurality of the sub-models.

[106] At 512b, the method assigning at least one sub-models from the plurality of sub-models to the at least one IoT device from the plurality of IoT devices 100b-100n based on the smart contract.At 514b, the method obtaining the output data of the processed sub-models of the non-ML data from each of the IoT devices. At 516b, the method generating the response for the query by processing the output data of the at least one sub-models processed by the first IoT device 100a and the output of the sub-models processed by the at least one IoT device.

[107] The various actions, acts, blocks, steps, or the like in the flowcharts 500a and 500bmay be performed in the order presented, in a different order or simultaneously. Further, in some embodiments, some of the actions, acts, blocks, steps, or the like may be omitted, added, modified, skipped, or the like without departing from the scope of the invention.

[108] FIG. 5c is a flowchart illustrating a method performed at an electronic device according to an embodiment.

[109] At 502c, a dataset may be identified at an electronic device. The electronic device may be the first IoT device 100a. The dataset may be identified in response to an input.

The input may be a query. The input may be input by a user or received from the user.

[110] In an embodiment, the dataset may include ML data. The dataset may include non-ML data. The dataset may include both of the ML data and the non-ML data, but is not limited thereto. The ML data may be an ML model.

[111] At 504c, the dataset may be divided into sub-datasets. When the dataset is or include ML data, the divided sub-datasets may be or include ML sub-models.

[112] At 506c, at least one electronic device may be identified. The identified at least one electronic device may be at least one of the IoT devices 100b-100n. The identified at least one electronic device may process at least one sub-datasets.

[113] At 508c, the at least one sub-dataset may be assigned to the at least one electronic device. The at least one electronic device may process the assigned at least one sub-dataset. Each of the at least one electronic device may process each of the at least one sub-dataset.

[114] At 510c, at least one output of the processed at least one sub-dataset may be received from the at least one electronic device. The received at least one output may be used to generate a response to the input. In an embodiment, the electronic device may also process a sub-dataset. The sub-datasets processed at the at least one electronic device and the electronic device are different and divided from the dataset.

[115] In an embodiment, the at least one output may be added as a block in a distributed ledger network, and consisting of a blockchain. The at least one output may be aggregated to generate a ML model. The generated ML model may be used to generate the response to the input. When the input includes a label input and a non-label input, each output for each input may be aggregated.

[116] In an embodiment, a smart contract between the electronic device and the at least one electronic device may be generated to process the sub-dataset. The sub-dataset may be assigned based on the generated smart contract. The output of the processed sub-dataset may be verified based on the smart contract on a distributed ledger network.

[117] In an embodiment, the electronic device may identify the at least one electronic device based on processing capability of the at least one electronic device. The at least one electronic device that has processing capability to process the sub-dataset may be identified among a plurality of electronic devices. The identified at least one electronic device may process the sub-dataset.

[118] In an embodiment, the electronic device may identify the at least one electronic device based on information of a user. The information of the user may be the user's profile information, the user's preference information, the user's behavior pattern information, the users' routine information, the user's calendar information, but is not limited thereto. The identified at least one electronic device may process the sub-dataset.

[119] In an embodiment, the electronic device may identify the at least one electronic device based on a context associated with the dataset.

[120] In an embodiment, the electronic device may identify the at least one electronic device based on a predetermined crieteria when at least two users are detected. FIG. 6a is a diagram for explaining washing machine personalization according to an embodiment, and FIG. 6b is a diagram for explaining the use of a smart contract according to an embodiment.

[121] The washing machine personalization may be performed according to an embodiment. For example, the table 1 indicates various devices and corresponding device profile data. Further, the device logs for the users (i.e., User 1, washing machine, Wednesday 7 am (morning), Colored, Regular, Warm; User 2, washing machine, Wednesday 10 am (morning), Whites, Regular, Warm; User 1, washing machine, Sunday 5 pm (evening), Delicates, Cold; and User 2, washing machine, Wednesday 8 am (morning), Colored, Regular, Warm). According to an embodiment, the IoT device 100a may create a distributed association rule mining information and association rules. The ML data may include the distributed association rule mining information and the association rules. In an example, based on the association rules, the washing machine personalization for the user 2 may be determined as frequent time of day: morning, frequent days of week: Wednesday, Fabric Type: Whites/Colored, Cycle: Regular, temperature: Warm, and the washing machine personalization for the user 1 may be determined as frequent time of day: morning, frequent days of week: Wednesday, Sunday, Fabric Type: Colored, Cycle: Regular, temperature: Warm.

[122] [Table 1]

Device Type	**Device Profile Data**
Washing Machine	Day of the week, Time, Type of fabric, Cycle, Water temperature
Fridge	Average Temperature, Time of Use, Duration, No. of times in a day door opened
Microwave	Use Frequency, Temperature Set, Average time of use
Lightbulb	Brightness / Luminescence
AC	Temperature setting
TV	Number of times used each day, what time of use, how long used

[123] The washing machine, the smart TV, and the smart bulb link at least one smart

contract to the blockchain infrastructure. In order to process the ML data, according to an embodiment, the processor 110 included in the washing machine may divide the ML data into the plurality of sub-models and process at least one sub-model from the plurality of sub-models based on the capability of the smart washing machine. Further, the processor 110 may assign remaining sub-models from the plurality of sub-models to the smart bulb and smart TV in the blockchain-based IoT network.

[124] The processor 110 included in the washing machine may obtain the output data of the remaining sub-models processed by the smart bulb and smart TV. Further, the processor 110 included in the washing machine may generate the response for the query by processing the output data of the at least one sub-models processed by the processor 110 included in the washing machine and the output of the remaining sub-models processed by the smart bulb and smart TV. The output data of the at least one sub-models processed by the washing machine and the output of the remaining sub-models processed by the smart bulb and smart TV may be aggregated to generate the response to the input or the query.

[125] 1. In an example, in the blockchain-based IoT network, the association rules of the transaction may be defined in terms of contracts. For example, the contract is defined as follows:

[126] Call_profile_for_user (username, device_type)

[127] Hence, if the device type is DEVICE_LIGHTBULB and username is USER_A, the following contract may be executed:

[128] Call_profile_for_user (USER_A, DEVICE_LIGHTBULB)

[129] In another example, the IoT device 100a may analyze personalization information for each user with respec to each device type, and generate device user profile for each user with resepect to each device type. The generated device user profile may be stored in the blockchain. The personalization procedure may use device logs for each device as inputs and produce outputs of the user device profile for each user for the given device. The generation of the IoT user profile for each device type may be performed on an IoT hub and other IoT devices 100b-100n on the IoT network. The IoT device 100a may be the IoT hub.

[130] In another example, in case of multiple users in the same home, the minimum and maximum setting values used by the users may be used to determine a mutually acceptable value for the device setting of the IoT devices 100a-100n connected to the hub. According to an embodiment, predefined rules may be used to create the device user profile on the hub. In an embodiment, the predefined rules may be defined by each user.

[131] For example, the following may be extracted from the usage log for the smart washing machine:

[132] a) 20 minutes on Wednesday 7 am (morning) by the first user,

[133] b) 30 minutes on Wednesday 10 am (morning) by the second user,

[134] c) 20 minutes on Sunday 5 pm (evening) by the first user, and

[135] d) 10 minutes on Wednesday 8 am (morning) by user 1.

[136] An example rule is as follows:

[137] Within the last month, if the smart washing machine is used for more than 5 minutes continuously and the same pattern is observed for at least 2 times, then the smart washing machine may update the user device profile for the user and device with the most frequent time of day and 2 most frequent days of week.

[138] The rules may be programmed and applied on the smart hub and devices connected to the hub. The profile information for all users and for all connected devices may be analyzed once a month and stored on the Interplanetary File System (IPFS). The IPFS is a protocol to store and share distributed content, and is frequently used on the blockchain.

[139] By applying this rule, the updated user device profile is as follows:

[140] a) (Second User, Washing machine, frequent time of day: morning, frequent days of week: Wednesday)

[141] b) (First User, Washing machine, frequent time of day: morning, frequent days of week: Wednesday, Sunday)

[142] This created device user profile for the washing machine may be stored in the IPFS. Below is sample of an JSON code for the user device profiles, which is prepared by the hub and stored in the IPFS.

[143] {

[144] "user_id":"user_1"

[145] "Washing machine":

[146] {

[147] "frequent time of day": "morning",

[148] "frequent days of week": "Wednesday"

[149] },

[150] "TV":

[151] {

[152] "frequent time of day": ["evening", "night"],

[153] "frequent days of week": "Wednesday"

[154] }

[155] }

[156] In another example, whenever the user enters the smart home, the following actions may take place according to an embodiment.

[157] The smart sensor may identify the user by associating his mobile phone or by using

sensors. Presence of the user may be determined in either of the following ways:

[158] a) Implicitly, by inferring a person from their activity profile, and

[159] b) Explicitly, by asking a person's smartphone to identify the person when he enters a room, use a new device, or log in to the hub using his smartphone.

[160] After identifying the user, the stored user device profile may be fetched from the blockchain in the following way: the smart hub may use the stored hash to obtain the transaction message from the blockchain (e.g., Ethereum). This first hash may be used to read the user profile information from the IPFS, which is then used to determine the settings to be applied on the smart device. Finally, the fetched profile parameters/ settings may be applied to the smart device.

[161] In another example, according to an embodiment, if there are multiple users present in a room, the IoT device 100 may resolve conflicts based on its settings which is any of the following:

[162] a. First come first serve (FCFS): The settings of the user who first enters the room are applied, until the user leaves the room,

[163] b. Last come first serve (LCFS): The settings of the user who enters the room in the last are applied, and

[164] c. Mean or Median: In case the setting value is numerical (such as AC temperature), the mean or median of the settings is taken. In case it is categorical, a voting system may be used and the setting with the most votes may be applied.

[165] FIG. 7 is a diagram for explaining a navigation system according to an embodiment.

[166] According to an embodiment, the navigation system in the smart car may provide a recommendation to the user based on the ML data. The ML data is trained by using the real time traffic data, the user data from the electronic device, a car data using on board display (OBD) and user personal calendar data. An output of the ML data may be obtained by processing an output data of the at least one sub-models processed by the processor 110 included in the smart car and the output of the remaining sub-models processed by another device, so that the navigation system provides a personalized recommendation to the user based on the ML data. The other device may be included in the smart car, but is not limited thereto.

[167] FIG. 8 is a diagram for explaining a virtual assistance application using a server. . In this method, the user may speak out a command of “Turn backlights ON” to a virtual assistance application. The virtual assistance application may send the command to a cloud server, and the machine learning model is used to find a response corresponding to the command. The cloud server may send the response to the virtual assistance application. The virtual assistance application may forward the response to the corresponding device (e.g., smart bulb, smart TV, smart washing machine or the like) to cause the corresponding device to perform a function corresponding to the command.

[168] FIG. 9 is a diagram for explaining an example scenario of the virtual assistance application according to an embodiment. According to an embodiment, the user may speak out a command of "Turn backlights ON" to the virtual assistance application. The virtual assistance application processes the command in the IoT environment without connecting to the cloud server. Further, the machine learning model may be used to find a response corresponding to the command of the user. A corresponding function may be executed on the corresponding IoT device (e.g., smart bulb).

[169] FIG. 10 is a diagram for explaining a guest verification procedure. According to this method, the security person needs to call the home owner to ask if the home owner is expecting a visitor. Based on the response, the guest verification procedure is completed manually.

[170] FIG. 11 is a diagram for explaining a guest verification procedure according to an embodiment, and FIG. 12 is a diagram for explaining a guest verification procedure according to an embodiment.

[171] As shown in the FIG. 11, according to an embodiment, the guest verification may be performed by a security guard's capturing photo of the visitor and processing the photo in the IoT environment. Based on the processing, when the visitor is identified correctly in any of the IoT devices then the gate will be opened. According to an embodiment, the photo may not be sent to any external device out of the IoT environment, which may enhance the security and privacy in an effective manner.

[172] As shown in the FIG. 12, the guest verification may be performed by the security guard's capturing the photo of the visitor and processing the photo in the IoT environment using the ML model. Based on the processing, when the visitor is identified correctly in any of the IoT devices then the gate will be opened. According to an embodiment, the photo is not shared with any external device out of the IoT enviornment, which may enhance the security and privacy in an effective manner.

[173] FIG. 13 is a diagram for explaining smart garbage collection according to an embodiment. . For example of smart garbage collection, when a container has an IoT sensor, the distributed ML may be used to calculate an optimized garbage collection route by using outputs from the IoT fill sensor. Further, the container may inform a truck where to go. This results in reducing the work load and allowing the garbage to be collected effectively.

[174] FIG. 14 is a diagram for explaining a smart home according to an embodiment.

[175] When different people living in shared house/office have different IoT device preferences (i.e., a first user wants air conditioner (AC) to be set as 20 degrees and a second user wants 30 degrees). According to an embodiment, when a person enters a smart home, a door sensor may be triggered based on the person walking past toward the smart home, and the user identification may be identified from user's phone.

Further, the door sensor may send event data to a smart TV to apply predicted user preferences of the identified person by using the ML model. The smart TV may obtain stored predictions for preferences for the person and apply the preferences for the person to its setting.

[176] FIG. 15 is a diagram for explaining smart fuel saving in a car according to an embodiment.

[177] In an example, an IoT mirror may predict fuel saving by using a mirror data ML model. An IoT light may predict fuel saving by using light data ML model, IoT door predicts fuel saving, Tablet/phone predicts fuel saving using engine speed data ML model, and IoT wheel may predict fuel saving by using wheel pressure data ML model, the fuel saving prediction may be performed based on the mirror data ML model, the light data ML model, the engine speed data ML model, and the wheel pressure data ML model.

[178] In another example, the proposed method may be used to provide equitable sharing of resources in a shared household, which may be used to split the electricity bill based on actual usage of devices, or as per some other arrangement (such as where the owner and renters live in same house and the owner does not need to pay but renters do). Here, once each user is identified for each device, the length of time may be billed in virtual currency to each user. At the end of each month, the virtual currency usage for each person is logged and the total bill for the shared household may be divided based on the actual usage of the devices by each user.

[179] In another example, the proposed method may be used to provide a water distribution management in a smart building. Here the water usage for each flat in the building may be determined with help of sensors fitted to water pipelines or taps, and accordingly the water distribution and water bill may be inferred for each flat.

[180] FIG. 16 is a diagram for explaining division of a machine learning model via forking according to an embodiment.

[181] The IoT device (e.g., smart bulb) may receive the input data and passes it to the smart hub. The smart hub may create the smart contracts and fork it to other nodes (i.e., other IoT devices). The smart contract may be defined to compute error gradients using a stochastic gradient descent (SGD) procedure. The other nodes may send the aggregated updated weight matrix to the smart hub. The updated weight matrix may be stored on the IPFS. The IPFS may send the transaction message to a blockchain platform (e.g., Ethereum). The hash of weight matrix may be stored on a blockchain in the smart hub. Ethereum may send another transaction message to the smart hub.

[182] In general, the hash of the device user profile information stored on the IPFS may be stored in Ethereum blockchain as a transaction message. The Ethereum network computes a second hash based on the transaction message which is stored in the smart

hub.

[183] FIG. 17 is a diagram for explaining an ML prediction using a blockchain according to an embodiment.

[184] The IoT device (e.g., smart bulb) may receive the input data and pass it to the smart hub. The smart hub may create and fork the smart contracts to other nodes (i.e., other IoT devices). The smart hub may store the hash value of weight matrix and send the hash value of weight matrix to a blockchain (e.g., Ethereum). Ethereum may send another transaction message in the form of the hash value to the distributed storage. The distributed storage may send the weight matrix to the smart hub.

[185] FIG. 18 is a diagram for explaining a random forest on a blockchain according to an embodiment, and FIG. 19 is a diagram for explaining a decision tree trained model working as a smart contract on a node in the blockchain according to an embodiment.

[186] As shown in the FIG. 18 and FIG. 19, the IoT device may implements each decision tree trained model computation as the smart contract on the node which is stored on the IPFS. The hash of each block may be stored on the blockchain. The decision tree smart contract in the nodes may be used to predict class given an input and to fork the computation to next node. The final block smart contract (running on the smart hub) may be for majority voting. The smart contract in the majority voting block is to perform the voting and get final prediction.

[187] In the random forest model, there is an ensemble of decision tree models and the majority of votes is taken as the predicted class.

[188] FIG. 20 is a diagram for explaining a deep neural network (DNN) on blockchain according to an embodiment.

[189] In the DNN model, there may be multiple nodes in the input, hidden, output layers, and each node may perform a nonlinear computation (such as Sigmoid or RELU). According to an embodiment, the IoT device may use the weight matrix of each layer (input layer, hidden layer, and output layer) as a smart contract computation in the node in the blockchain. The smart contract in each node is to (a) Forward phase: compute product of weight matrix with the input vector in the forward pass (b) Backward phase: modify weights after back propagating the errors in the forward phase. Each block performs its computation or forks it to a neighboring node

[190] The various actions, acts, blocks, steps, or the like in the flowchart may be performed in the order presented, in a different order or simultaneously. Further, in some embodiments, some of the actions, acts, blocks, steps, or the like may be omitted, added, modified, skipped, or the like without departing from the scope of the invention.

[191] The embodiments disclosed herein may be implemented using at least one software program running on at least one hardware device and performing network management functions to control the elements.

[192] The foregoing description of the specific embodiments will so fully reveal the general nature of the embodiments herein that others can, by applying current knowledge, readily modify and/or adapt for various applications such specific embodiments without departing from the generic concept, and, therefore, such adaptations and modifications should and are intended to be comprehended within the meaning and range of equivalents of the disclosed embodiments. It is to be understood that the phraseology or terminology employed herein is for the purpose of description and not of limitation. Therefore, while the embodiments herein have been described in terms of preferred embodiments, those skilled in the art will recognize that the embodiments herein may be practiced with modification within the spirit and scope of the embodiments as described herein.

Claims

[Claim 1] An electronic device comprising:
a memory storing instructions; and
at least one processor configured to execute the instructions to:
in response to an input, identify a dataset to be processed for responding to the input;
divide the dataset into a plurality of sub-datasets;
identify at least one electronic device which processes at least one sub-dataset;
assign the at least one sub-dataset to the at least one electronic device to process the at least one sub-dataset; and
receive from the at least one electronic device at least one output of the processed at least one sub-dataset to generate a response to the input.

[Claim 2] The electronic device of claim 1, wherein the at least one output of the processed at least one sub-dataset is added as a block in a distributed ledger network.

[Claim 3] The electronic device of claim 1, wherein the at least one processor is further configured to execute the instructions to:
generate a smart contract with the at least one electronic device to process the at least one sub-dataset; and
assign the at least one sub-dataset to the at least one electronic device based on the smart contract,
wherein the at least one output of the processed at least one sub-dataset is verified on a distributed ledger network based on the smart contract.

[Claim 4] The electronic device of claim 1, wherein the dataset comprises a machine learning (ML) model, and the plurality of sub-datasets comprises a plurality of sub-models.

[Claim 5] The electronic device of claim 1, wherein the at least one processor is further configured to execute the instructions to:
process a sub-dataset to produce an output; and
aggregate the output of the processed sub-dataset, and the at least one output of the processed at least one sub-dataset to generate the response to the input.

[Claim 6] The electronic device of claim 5, wherein the at least one processor is further configured to execute the instructions to:
generate a ML model based on the aggregation of outputs to generate the response to the input by using the generated ML model.

[Claim 7] The electronic device of claim 5, wherein the input is divided into a label input and a non-label input, and each output for each divided input are aggregated to generate the response to the input.

[Claim 8] The electronic device of claim 1, wherein the at least one electronic device is identified based on processing capability of the at least one electronic device.

[Claim 9] The electronic device of claim 1, wherein the at least one electronic device is identified based on information of a user.

[Claim 10] The electronic device of claim 1, wherein the at least one processor is further configured to execute the instructions to detect a context associated with the dataset, wherein the at least one electronic device is identified based on the context associated with the dataset.

[Claim 11] The electronic device of claim 1, wherein the at least one electronic device is identified based on a predetermined criteria when at least two users are detected by the electronic device.

[Claim 12] An electronic device comprising:
a memory storing instructions; and
at least one processor configured to execute the instructions to:
receve from another electronic device a sub-dataset divided from a dataset which is identifed, at the other electronic device, for responding to an input;
process the sub-dataset to produce an output; and
transmit to the other electronic device the output of the processed sub-dataset to generate a response to the input.

[Claim 13] A method comprising:
in response to an input, identifying a dataset to be processed for responding to the input;
dividing the data into a plurality of sub-datasets;
identifying at least one electronic device which processes at least one sub-dataset;
assigning the at least one sub-dataset to the at least one electronic device to process the at least one sub-dataset; and
receiving from the at least one electronic device an output of the processed at least one sub-dataset to generate a response to the input.

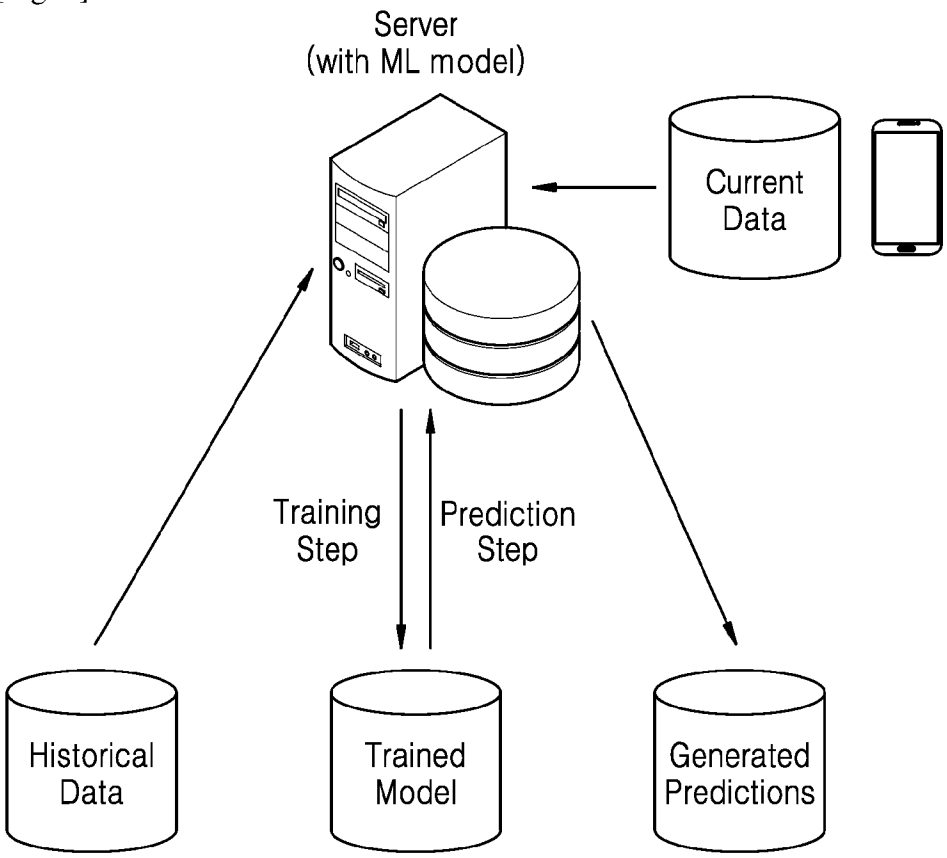
[Claim 14] A method comprising:
receving from another electronic device a sub-dataset divided from a dataset which is identifed, at the other electronic device, for responding to an input;

processing the sub-dataset to produce an output; and

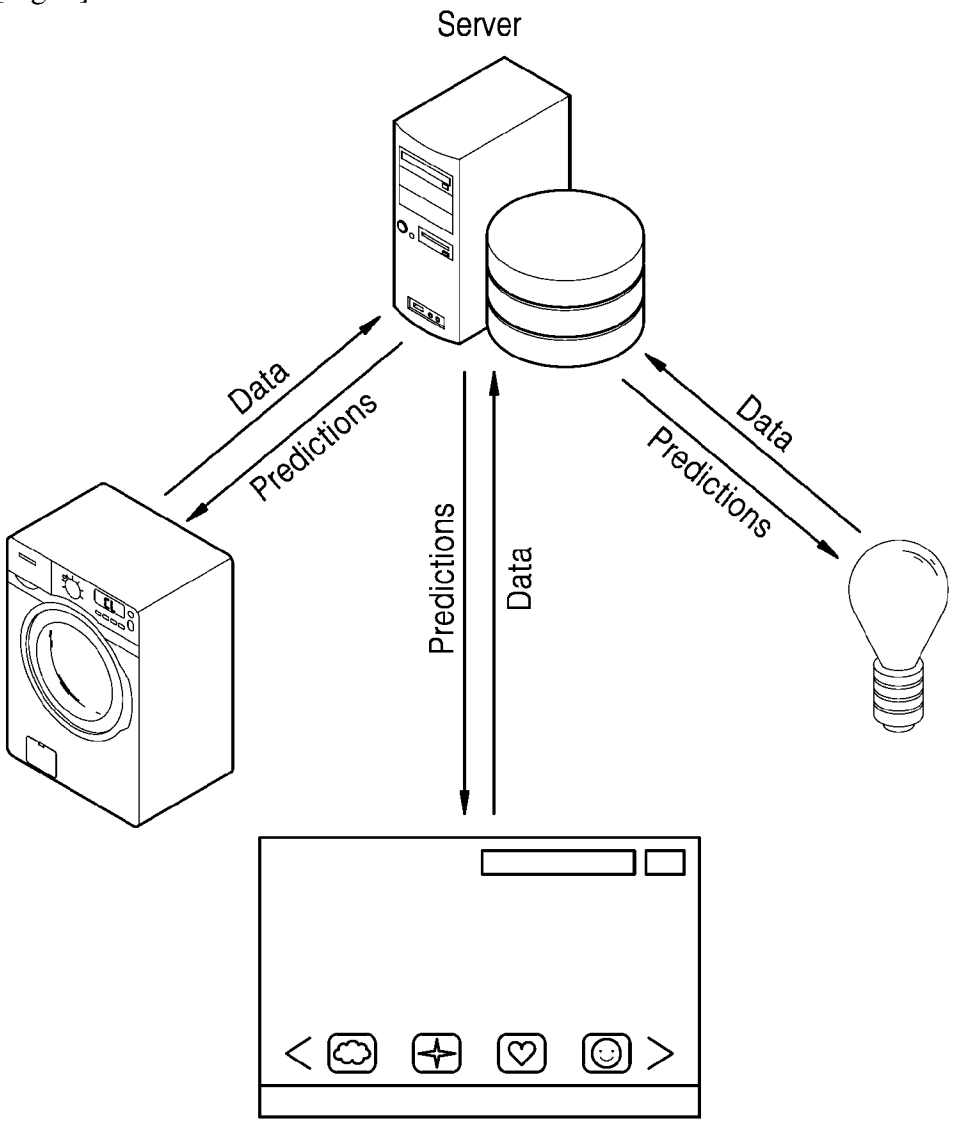
transmitting to the other electronic device the output of the processed sub-dataset to generate a response to the input.

[Claim 15] A computer program comprising instructions which, when the computer program is executed by at least one processor, cause the at least one processor to carry out the method of one of claims 13 or 14.

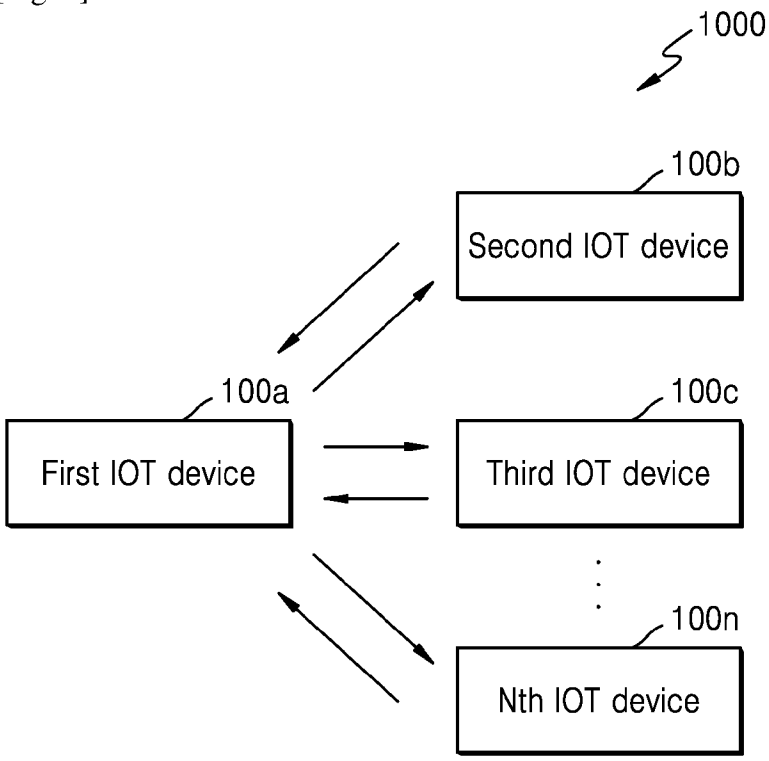
[Fig. 1]


Server
(with ML model)
Current
Data
Training
Step
Prediction
Step
Historical
Data
Trained
Model
Generated
Predictions

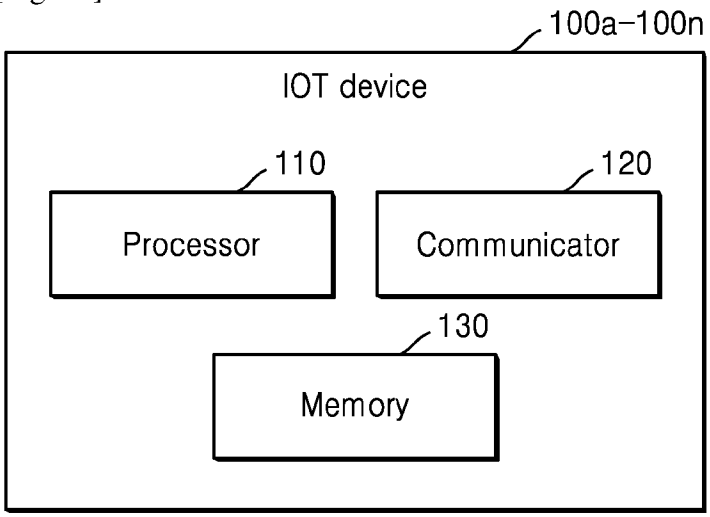
[Fig. 2]


Server
Data
Predictions
Predictions
Data
Data
Predictions

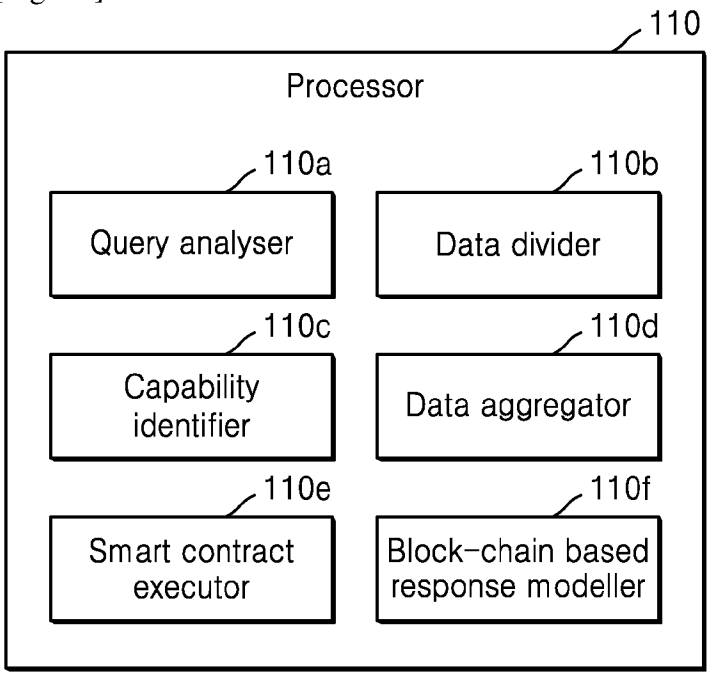
[Fig. 3]


1000
100b
Second IOT device
100a
100c
First IOT device
Third IOT device
100n
Nth IOT device

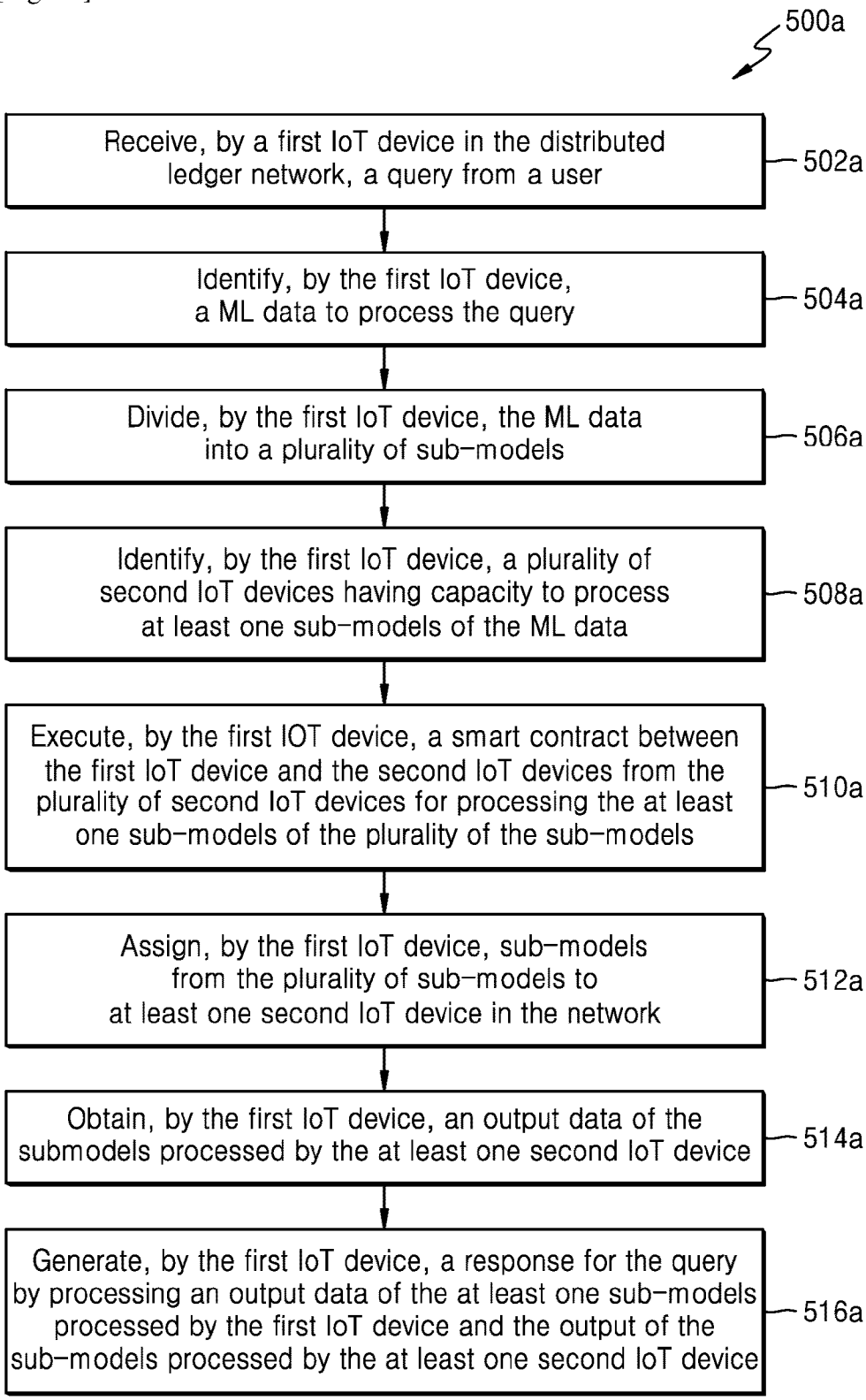
[Fig. 4a]


100a-100n
IOT device
110
Processor
120
Communicator
130
Memory

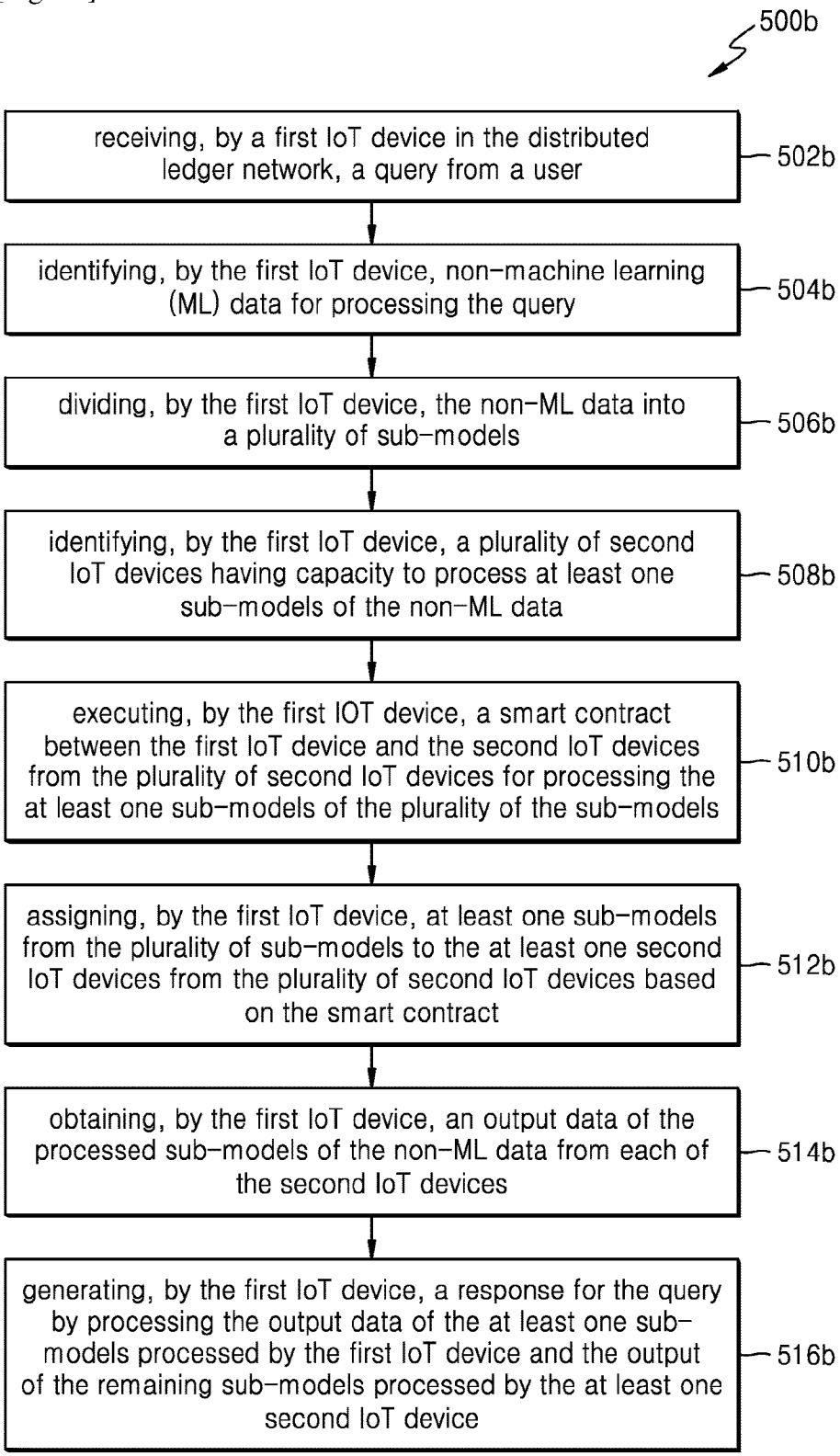
[Fig. 4b]


110
Processor
110a
Query analyser
110b
Data divider
110c
Capability identifier
110d
Data aggregator
110e
Smart contract executor
110f
Block-chain based response modeller

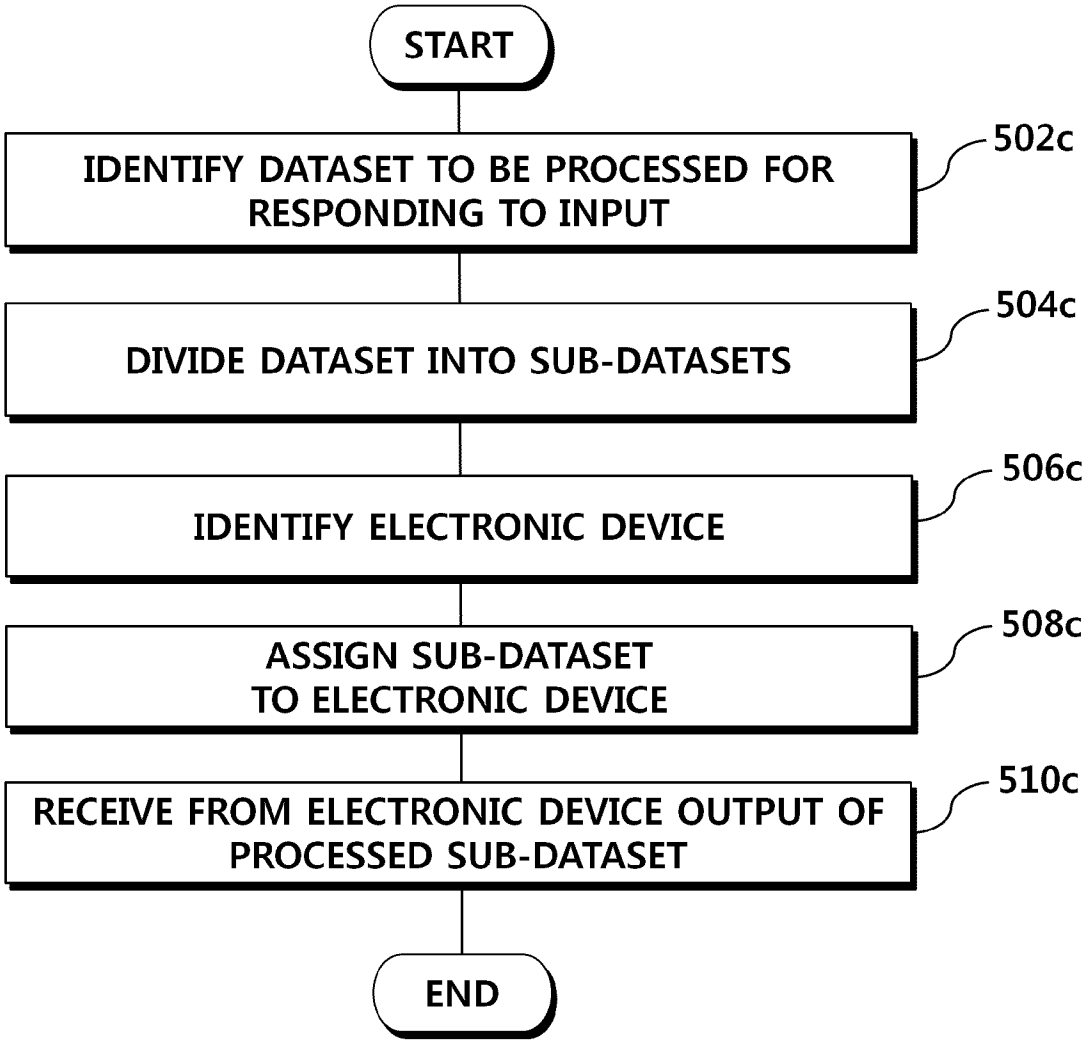
[Fig. 5a]


500a
Receive, by a first IoT device in the distributed ledger network, a query from a user
502a
Identify, by the first IoT device, a ML data to process the query
504a
Divide, by the first IoT device, the ML data into a plurality of sub-models
506a
Identify, by the first IoT device, a plurality of second IoT devices having capacity to process at least one sub-models of the ML data
508a
Execute, by the first IOT device, a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models
510a
Assign, by the first IoT device, sub-models from the plurality of sub-models to at least one second IoT device in the network
512a
Obtain, by the first IoT device, an output data of the submodels processed by the at least one second IoT device
514a
Generate, by the first IoT device, a response for the query by processing an output data of the at least one sub-models processed by the first IoT device and the output of the sub-models processed by the at least one second IoT device
516a

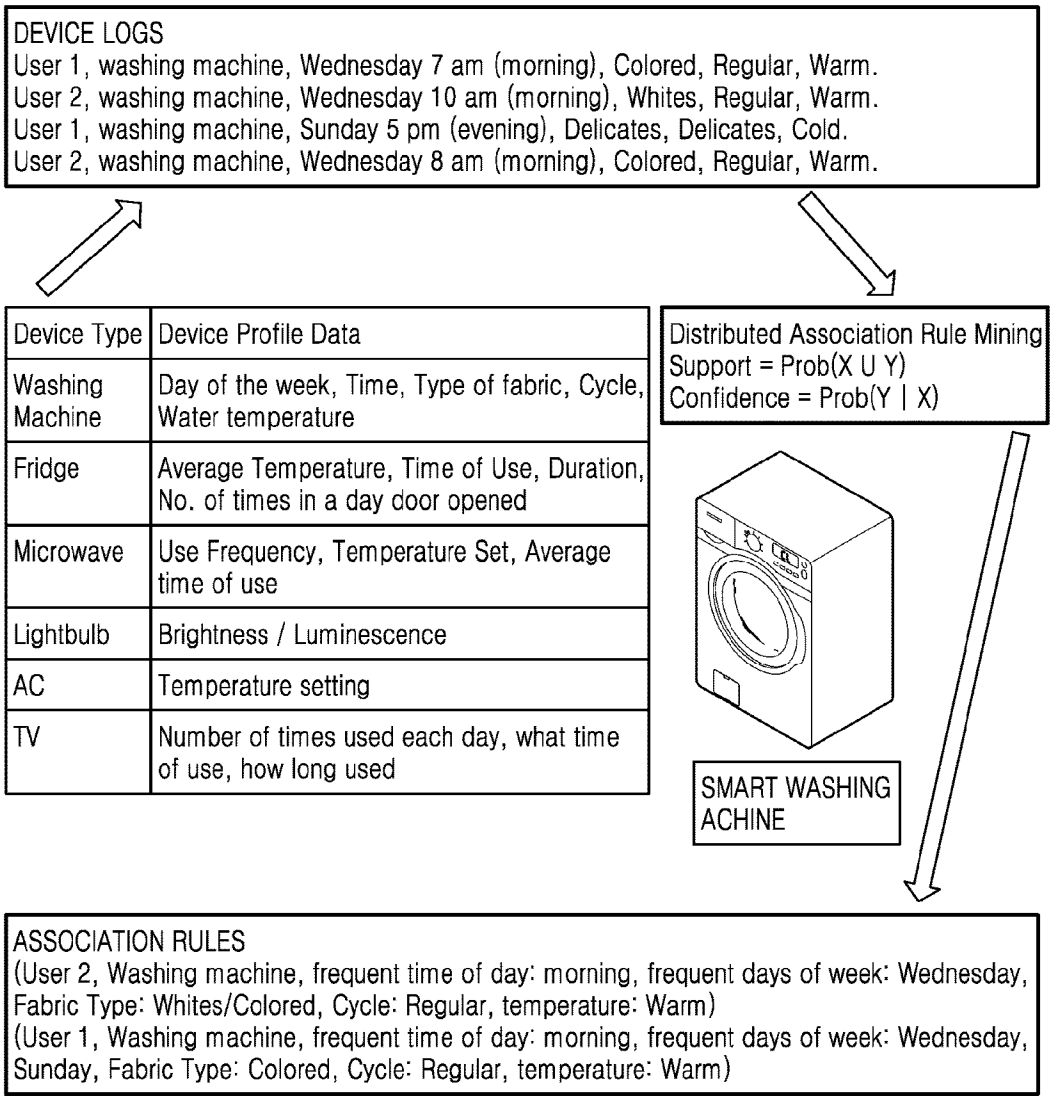
[Fig. 5b]


500b
receiving, by a first IoT device in the distributed ledger network, a query from a user
502b
identifying, by the first IoT device, non-machine learning (ML) data for processing the query
504b
dividing, by the first IoT device, the non-ML data into a plurality of sub-models
506b
identifying, by the first IoT device, a plurality of second IoT devices having capacity to process at least one sub-models of the non-ML data
508b
executing, by the first IOT device, a smart contract between the first IoT device and the second IoT devices from the plurality of second IoT devices for processing the at least one sub-models of the plurality of the sub-models
510b
assigning, by the first IoT device, at least one sub-models from the plurality of sub-models to the at least one second IoT devices from the plurality of second IoT devices based on the smart contract
512b
obtaining, by the first IoT device, an output data of the processed sub-models of the non-ML data from each of the second IoT devices
514b
generating, by the first IoT device, a response for the query by processing the output data of the at least one sub-models processed by the first IoT device and the output of the remaining sub-models processed by the at least one second IoT device
516b

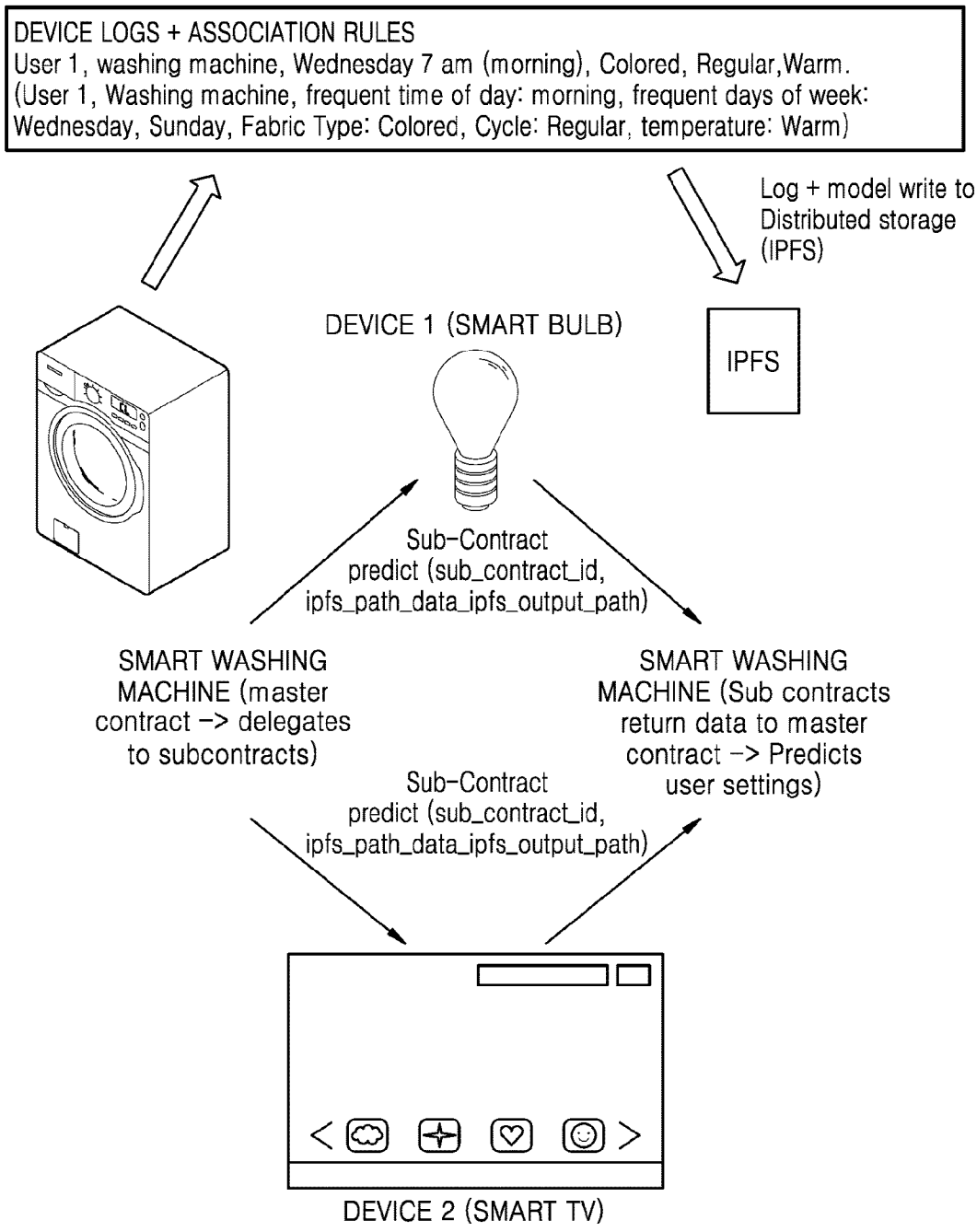
[Fig. 5c]


START
IDENTIFY DATASET TO BE PROCESSED FOR RESPONDING TO INPUT
502c
DIVIDE DATASET INTO SUB-DATASETS
504c
IDENTIFY ELECTRONIC DEVICE
506c
ASSIGN SUB-DATASET TO ELECTRONIC DEVICE
508c
RECEIVE FROM ELECTRONIC DEVICE OUTPUT OF PROCESSED SUB-DATASET
510c
END

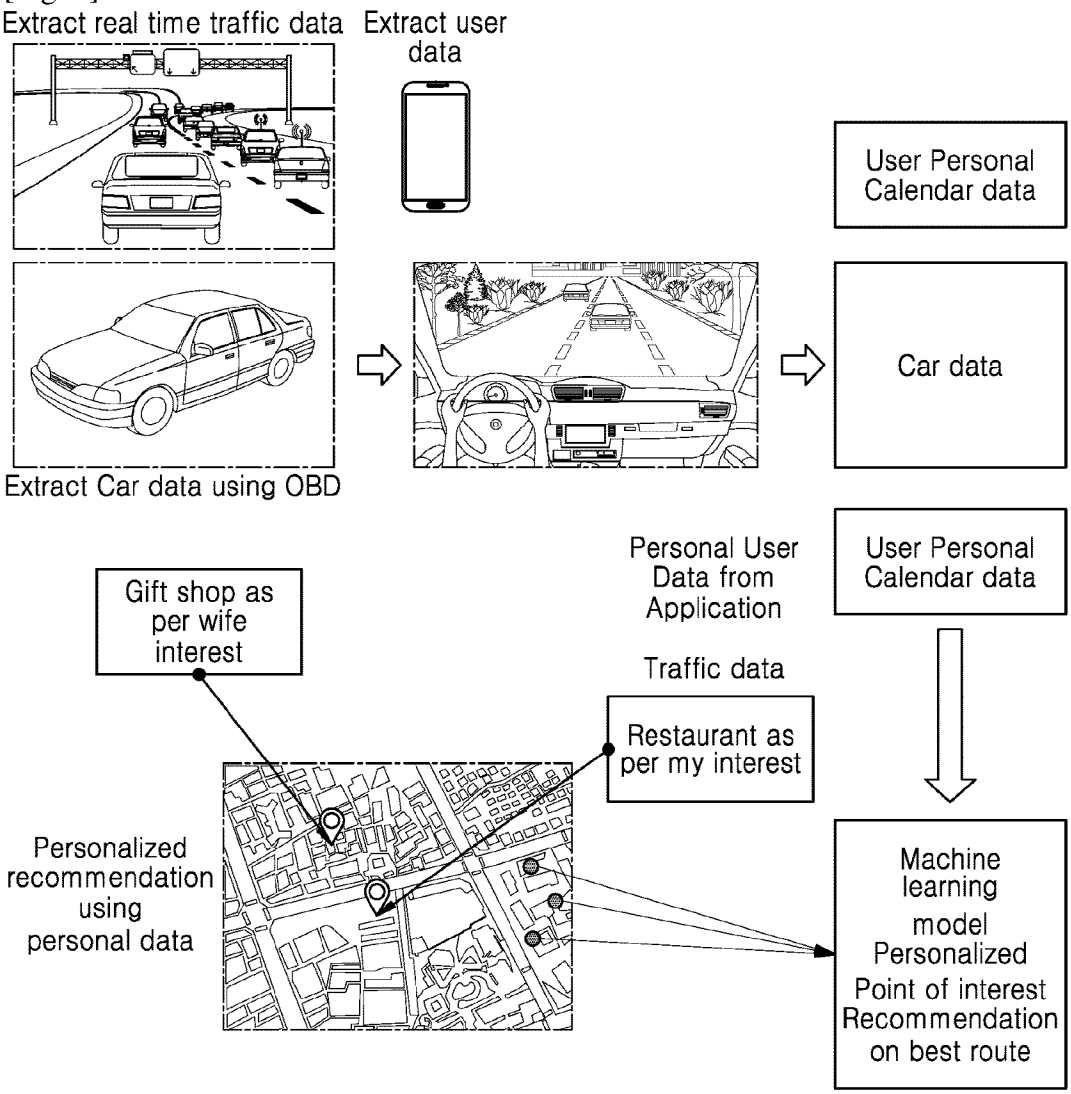
[Fig. 6a]


DEVICE LOGS
User 1, washing machine, Wednesday 7 am (morning), Colored, Regular, Warm.
User 2, washing machine, Wednesday 10 am (morning), Whites, Regular, Warm.
User 1, washing machine, Sunday 5 pm (evening), Delicates, Delicates, Cold.
User 2, washing machine, Wednesday 8 am (morning), Colored, Regular, Warm.
Device Type
Device Profile Data
Washing Machine
Day of the week, Time, Type of fabric, Cycle, Water temperature
Fridge
Average Temperature, Time of Use, Duration, No. of times in a day door opened
Microwave
Use Frequency, Temperature Set, Average time of use
Lightbulb
Brightness / Luminescence
AC
Temperature setting
TV
Number of times used each day, what time of use, how long used
Distributed Association Rule Mining
Support = Prob(X U Y)
Confidence = Prob(Y | X)
SMART WASHING ACHINE
ASSOCIATION RULES
(User 2, Washing machine, frequent time of day: morning, frequent days of week: Wednesday, Fabric Type: Whites/Colored, Cycle: Regular, temperature: Warm)
(User 1, Washing machine, frequent time of day: morning, frequent days of week: Wednesday, Sunday, Fabric Type: Colored, Cycle: Regular, temperature: Warm)

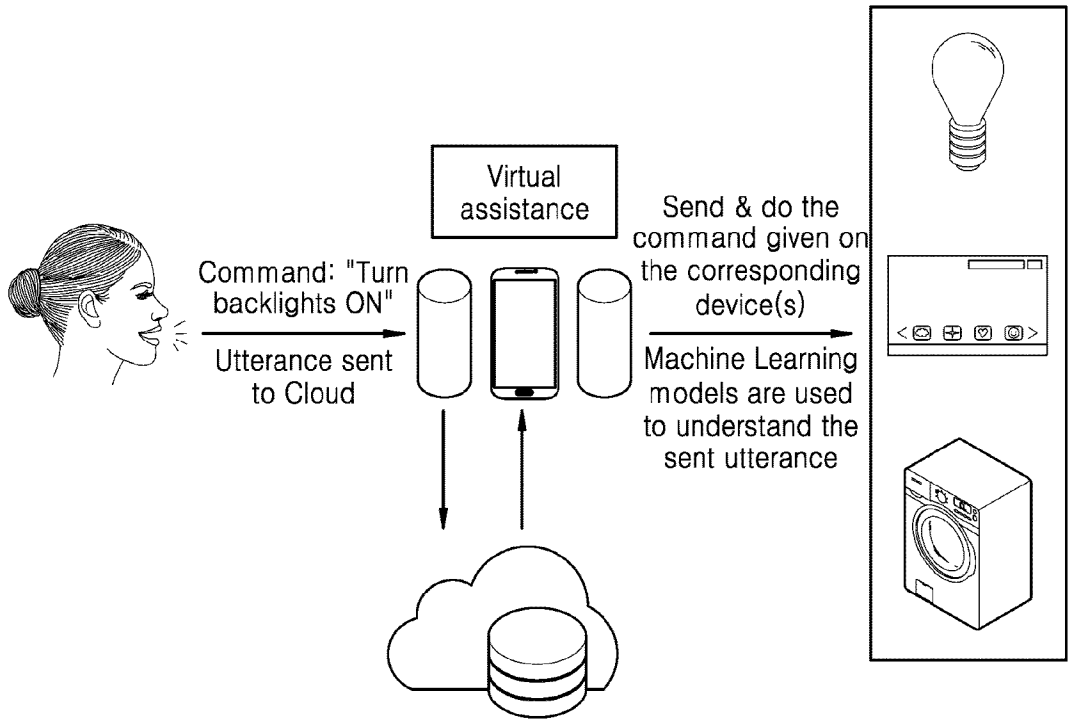
[Fig. 6b]


DEVICE LOGS + ASSOCIATION RULES
User 1, washing machine, Wednesday 7 am (morning), Colored, Regular,Warm.
(User 1, Washing machine, frequent time of day: morning, frequent days of week: Wednesday, Sunday, Fabric Type: Colored, Cycle: Regular, temperature: Warm)
Log + model write to Distributed storage (IPFS)
DEVICE 1 (SMART BULB)
IPFS
Sub-Contract predict (sub_contract_id, ipfs_path_data_ipfs_output_path)
SMART WASHING MACHINE (master contract -> delegates to subcontracts)
SMART WASHING MACHINE (Sub contracts return data to master contract -> Predicts user settings)
Sub-Contract predict (sub_contract_id, ipfs_path_data_ipfs_output_path)
DEVICE 2 (SMART TV)

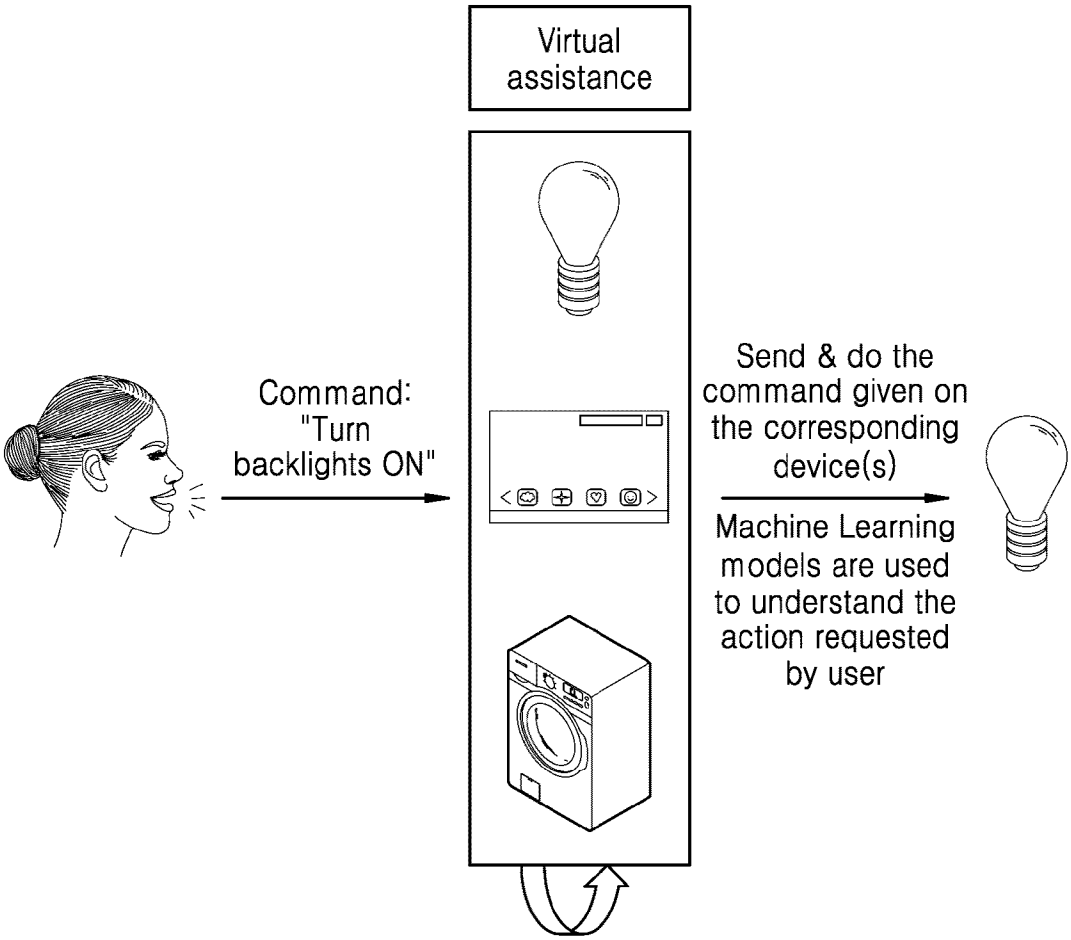
[Fig. 7]


Extract real time traffic data
Extract user data
User Personal Calendar data
Car data
Extract Car data using OBD
Personal User Data from Application
User Personal Calendar data
Gift shop as per wife interest
Traffic data
Restaurant as per my interest
Personalized recommendation using personal data
Machine learning model Personalized Point of interest Recommendation on best route

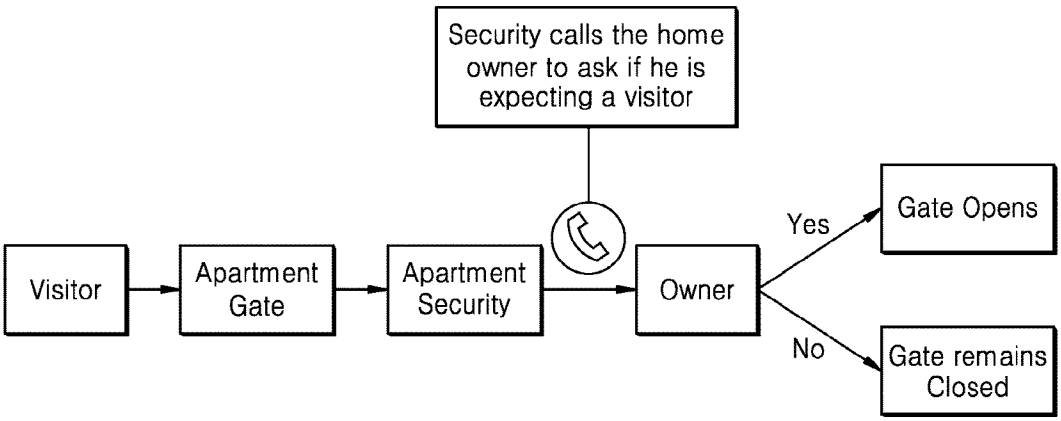
[Fig. 8]


Virtual assistance
Command: "Turn backlights ON"
Utterance sent to Cloud
Send & do the command given on the corresponding device(s)
Machine Learning models are used to understand the sent utterance

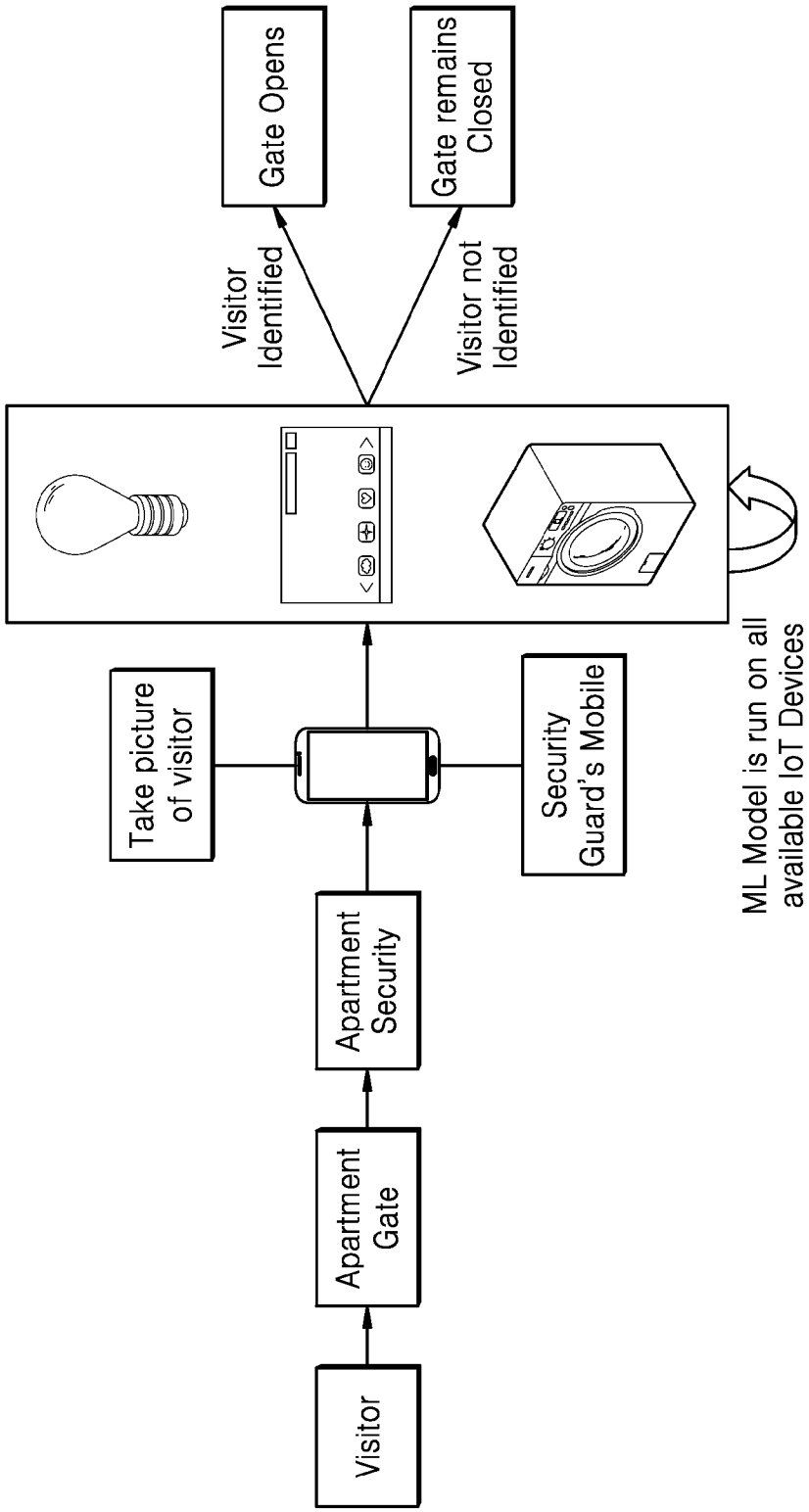
[Fig. 9]


Virtual assistance
Command: "Turn backlights ON"
Send & do the command given on the corresponding device(s)
Machine Learning models are used to understand the action requested by user

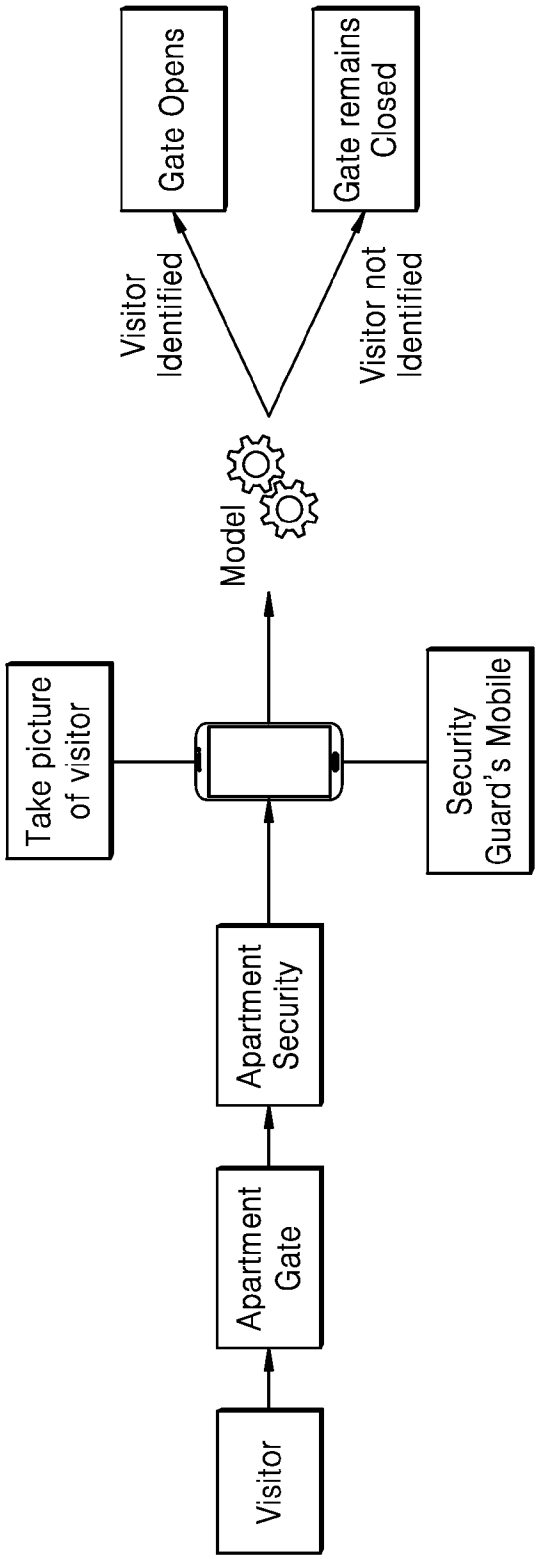
[Fig. 10]


Security calls the home owner to ask if he is expecting a visitor
Visitor
Apartment Gate
Apartment Security
Owner
Yes
Gate Opens
No
Gate remains Closed

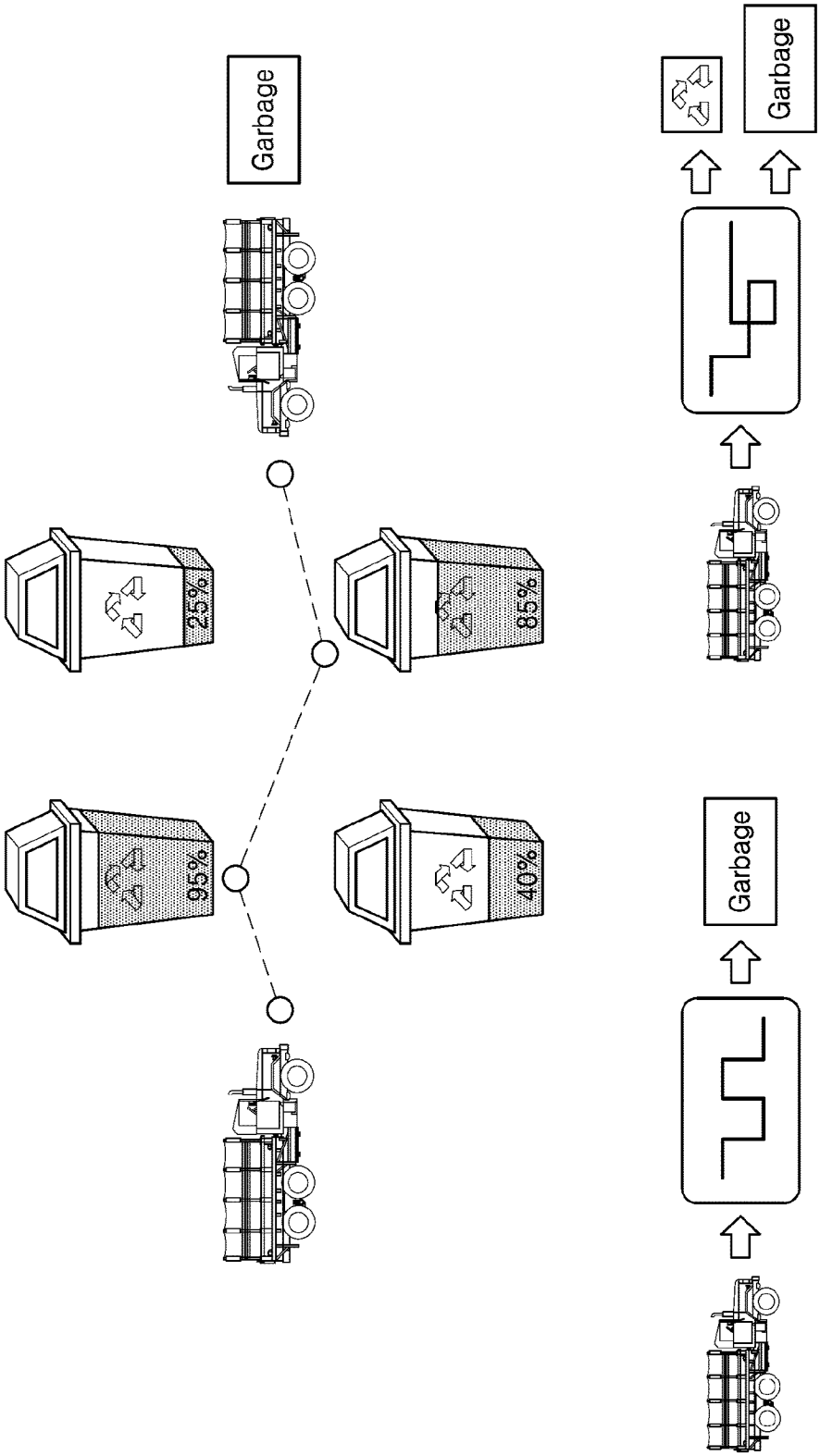
[Fig. 11]


Gate Opens
Gate remains Closed
Visitor Identified
Visitor not Identified
Take picture of visitor
Security Guard's Mobile
ML Model is run on all available IoT Devices
Apartment Security
Apartment Gate
Visitor

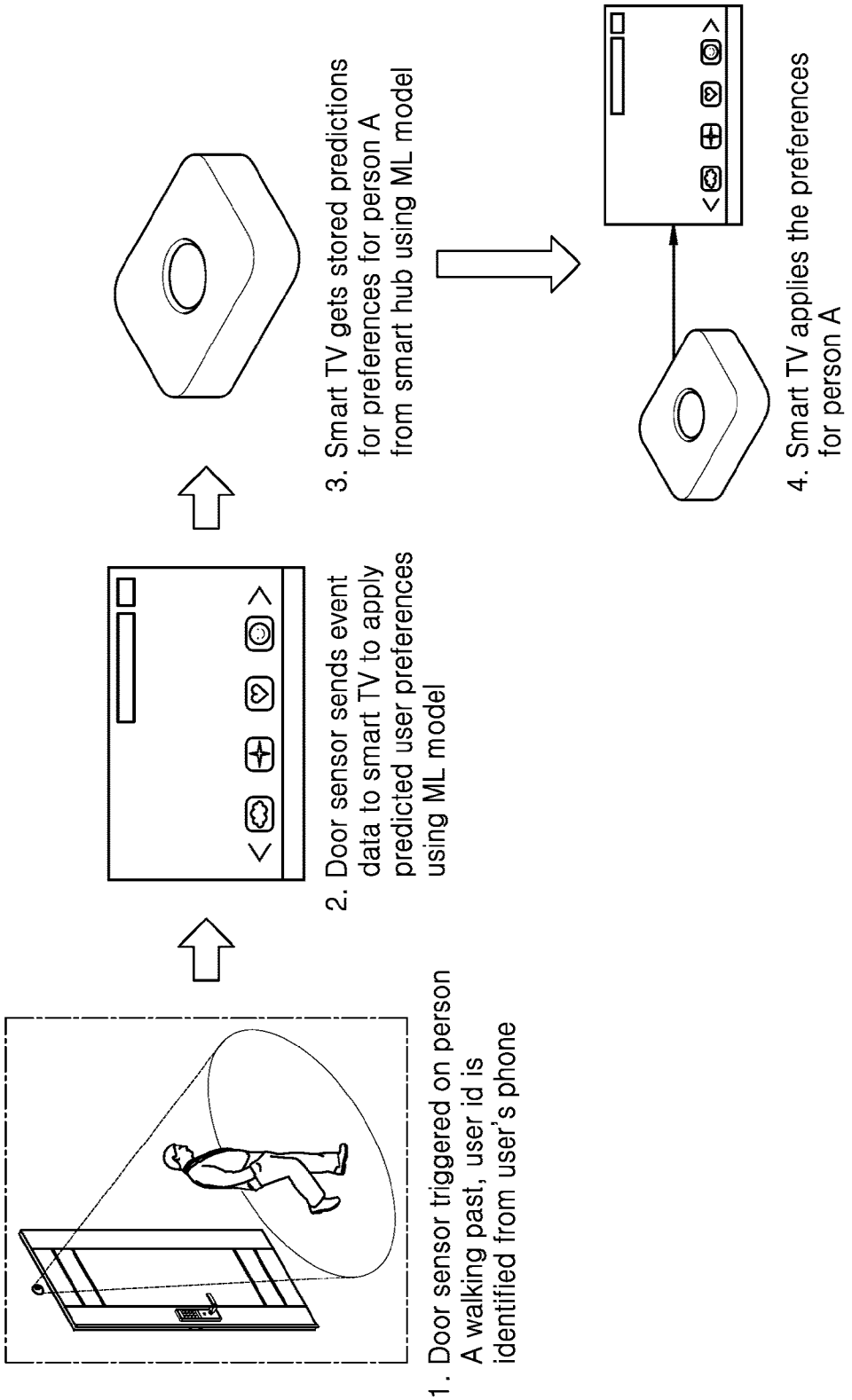
[Fig. 12]


Gate Opens
Gate remains Closed
Visitor Identified
Visitor not Identified
Model
Take picture of visitor
Security Guard's Mobile
Apartment Security
Apartment Gate
Visitor

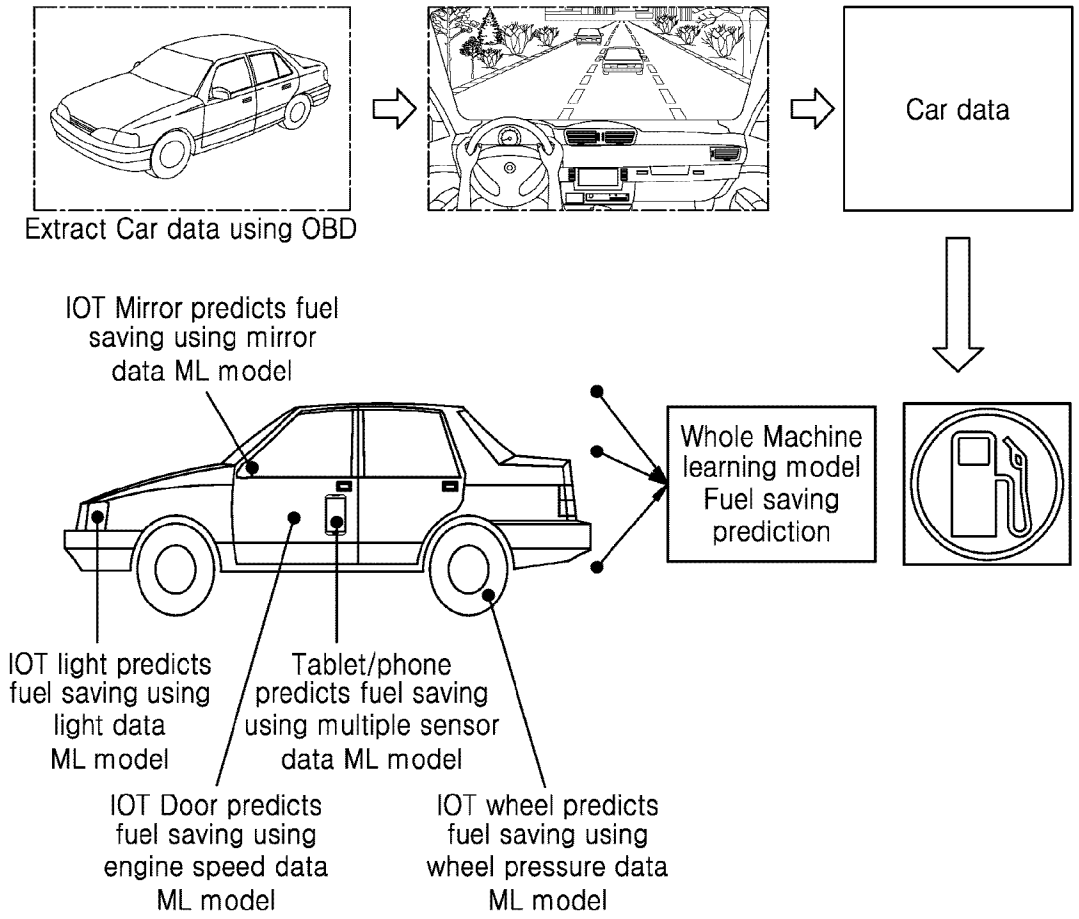
[Fig. 13]



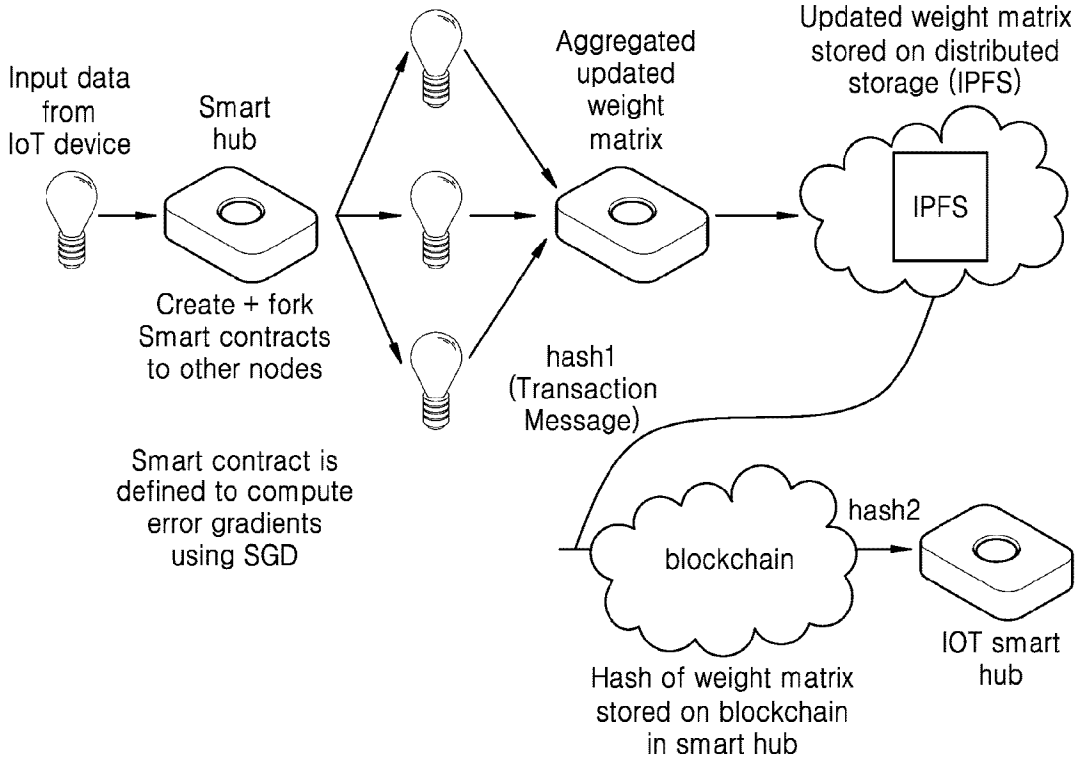
[Fig. 14]


1. Door sensor triggered on person A walking past, user id is identified from user's phone
2. Door sensor sends event data to smart TV to apply predicted user preferences using ML model
3. Smart TV gets stored predictions for preferences for person A from smart hub using ML model
4. Smart TV applies the preferences for person A

[Fig. 15]

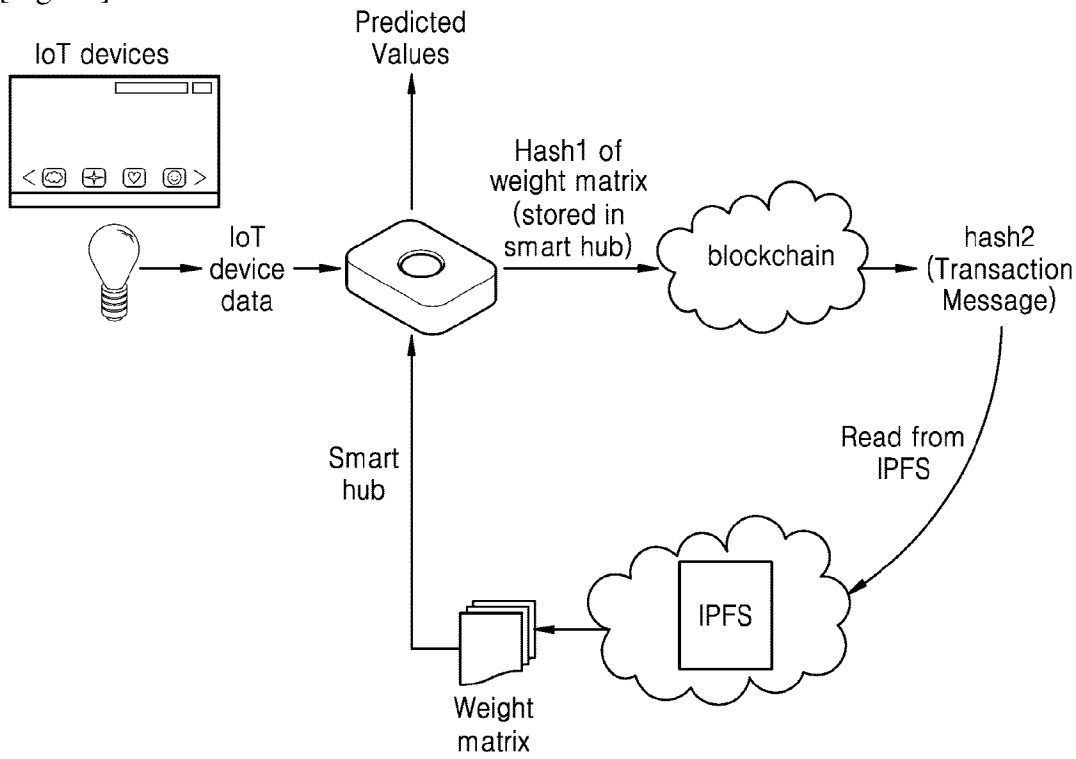


[Fig. 16]

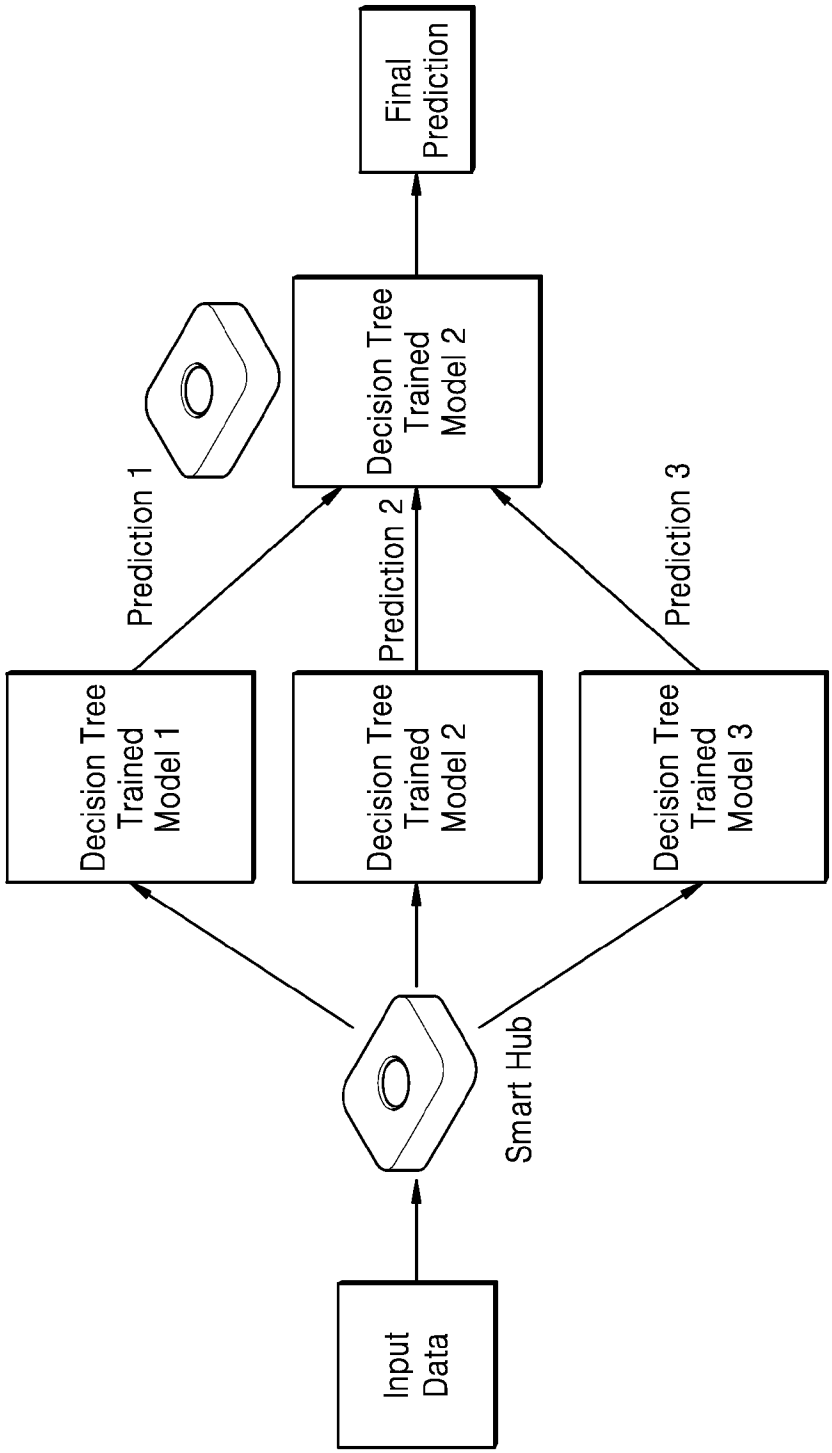


* SGD = Stochastic Gradient Descent
Weight update equation
$w(t+1) = w(t) + \lambda \Sigma^{k}_{i=1} \delta d_i / \delta w_i$
Where λ = learning rate

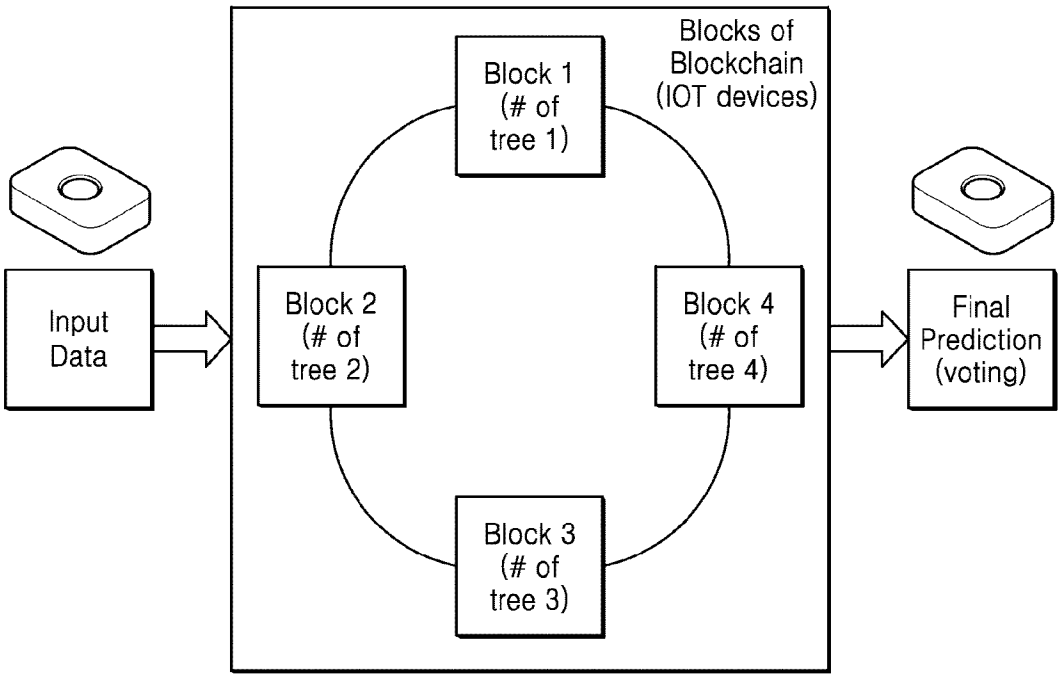
[Fig. 17]


IoT devices
Predicted Values
Hash1 of weight matrix (stored in smart hub)
blockchain
hash2 (Transaction Message)
IoT device data
Smart hub
Read from IPFS
IPFS
Weight matrix

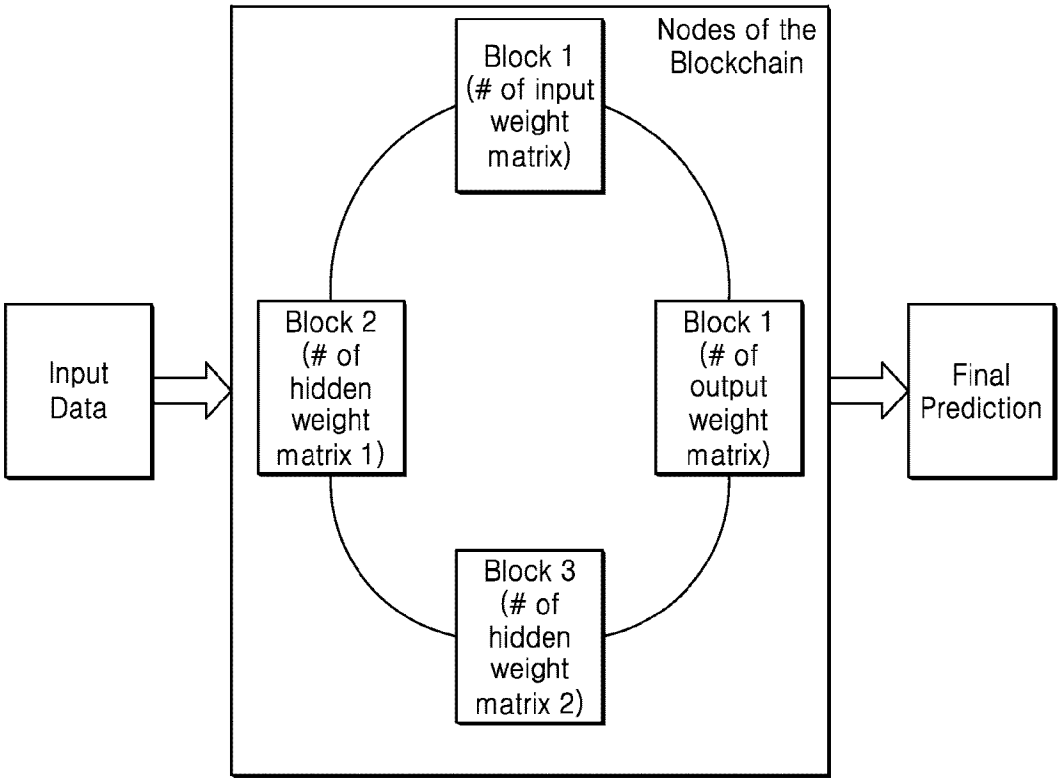
[Fig. 18]


Final Prediction
Decision Tree Trained Model 2
Prediction 1
Prediction 2
Prediction 3
Decision Tree Trained Model 1
Decision Tree Trained Model 2
Decision Tree Trained Model 3
Smart Hub
Input Data

[Fig. 19]


Blocks of Blockchain (IOT devices)
Block 1 (# of tree 1)
Block 2 (# of tree 2)
Block 3 (# of tree 3)
Block 4 (# of tree 4)
Input Data
Final Prediction (voting)

[Fig. 20]


Nodes of the Blockchain
Block 1 (# of input weight matrix)
Block 2 (# of hidden weight matrix 1)
Block 1 (# of output weight matrix)
Block 3 (# of hidden weight matrix 2)
Input Data
Final Prediction

INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR2019/005034

A. CLASSIFICATION OF SUBJECT MATTER

G06N 20/00(2019.01)i, H04L 29/08(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06N 20/00; G06F 17/30; G06F 19/00; G06F 21/62; G06N 3/08; G06N 99/00; G08B 21/04; G08B 25/01; H04L 29/08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & keywords: machine learning, blockchain, model, dataset, sub-dataset, electronic device, distributed ledger network

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2015-0193695 A1 (CISCO TECHNOLOGY, INC.) 09 July 2015 See paragraphs [0072]-[0074], [0129]-[0132]; and claims 11, 15-16.	1-15
A	TSUNG-TING KUO et al., `ModelChain: Decentralized Privacy-Preserving Healthcare Predictive Modeling Framework on Private Blockchain Networks`, arXiv:1802.01746v1, 06 Februrary 2018 [retrieved on: 2019.07.18]. Retrieved from <URL: https://arxiv.org/ftp/arxiv/papers/1802/1802.01746.pdf> See page 2; and figures 1-2.	1-15
A	US 2018-0018590 A1 (NANTOMICS, INC. et al.) 18 January 2018 See paragraphs [0083]-[0087]; claim 1; and figure 4.	1-15
A	US 2017-0103167 A1 (NETSPECTIVE COMMUNICATIONS LLC) 13 April 2017 See paragraphs [0080]-[0084]; and figures 1, 9.	1-15
A	US 2015-0061863 A1 (HTI IP, L.L.C.) 05 March 2015 See paragraphs [0068]-[0073]; and figures 1, 9.	1-15

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
06 August 2019 (06.08.2019)	**06 August 2019 (06.08.2019)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
KWON, Sung Ho
Telephone No. +82-42-481-3547

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/KR2019/005034

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2015-0193695 A1	09/07/2015	EP 3092742 A1	16/11/2016
		EP 3092776 A1	16/11/2016
		US 2015-0193693 A1	09/07/2015
		US 2015-0193694 A1	09/07/2015
		US 2015-0193696 A1	09/07/2015
		US 2015-0193697 A1	09/07/2015
		US 2015-0195145 A1	09/07/2015
		US 2015-0195146 A1	09/07/2015
		US 2015-0195296 A1	09/07/2015
		US 9160760 B2	13/10/2015
		US 9413779 B2	09/08/2016
		US 9450978 B2	20/09/2016
		US 9503466 B2	22/11/2016
		US 9521158 B2	13/12/2016
		US 9563854 B2	07/02/2017
		US 9870537 B2	16/01/2018
		WO 2015-103514 A1	09/07/2015
		WO 2015-103520 A1	09/07/2015
US 2018-0018590 A1	18/01/2018	AU 2017-300259 A1	14/02/2019
		CA 3031067 A1	25/01/2018
		CN 109716346 A	03/05/2019
		EP 3485436 A1	22/05/2019
		KR 10-2019-0032433 A	27/03/2019
		SG 11201900220 A	27/02/2019
		TW 201812646 A	01/04/2018
		WO 2018-017467 A1	25/01/2018
US 2017-0103167 A1	13/04/2017	US 2013-0290022 A1	31/10/2013
US 2015-0061863 A1	05/03/2015	US 9589442 B2	07/03/2017