December 06, 2018

RATIONALIZED CLUSTERING IN MACHINE LEARNING

Diveplane Corporation

Christopher Hazard PhD
Diveplane Corporation

UNITED STATES PATENT APPLICATION

FOR

RATIONALIZED CLUSTERING IN MACHINE LEARNING

INVENTORS:

CHRISTOPHER HAZARD

FIELD OF THE INVENTION

[0001] The present invention relates to machine learning and more specifically to clustering techniques in machine learning.

BACKGROUND

[0002] In machine learning, there are currently debates about what an explanation or explainable model is and what is necessary for a given purpose. The techniques herein overcome these issues.

[0003] The approaches described in this section are approaches that could be pursued, but not necessarily approaches that have been previously conceived or pursued. Therefore, unless otherwise indicated, it should not be assumed that any of the approaches described in this section qualify as prior art merely by virtue of their inclusion in this section.

SUMMARY

[0004] The claims provide a summary of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] In the drawings:

[0006] FIG. 1 is a flow diagram depicting example processes for controlling systems.

[0007] FIG. 2 is a block diagram depicting example systems for rationalized clustering.

[0008] FIG. 3 is a block diagram of example hardware rationalized clustering.

DETAILED DESCRIPTION

[0009] In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to avoid unnecessarily obscuring the present invention.

[0010] In machine learning, there are currently debates about what an explanation or explainable model is and what is necessary for a given purpose. This post details the concepts of explanation and interpretation to help clarify the difference between the two; discusses how, although interpretation is preferable, explanation is the only option for many machine learning techniques; and then details a clustering technique that aids explanation for unsupervised machine learning.

[0011] One pair of definitions from academia splits the field into interpretability and explainability. An interpretation is a set of understandable reasons why a model made a decision, which means that the model itself must in some way be sufficiently understandable to interpret. Examples of interpretable models include decision trees and some applications of linear regression. An explanation is a set of understandable reasons that rationalize or justify why a model made a decision that may or may not be at all related to why a model made a particular decision. Examples of explanations for models include LIME, Google's What-If Tool, and IBM's AIF360.

[0012] Interpretable models are essential for understanding why a decision was actually made. The difference between an interpretable model and an explainable model can be illuminated via a simple example. Suppose a person buys a car. An explanation, or rationalization, for the decision might be that the person justifies buying the car because their old car was becoming unreliable, the new car was in their price range, and the new car had all the features and properties they desired. However, the real reasons that the person bought the car, if we were to be able to interpret the internals of the person's thought process, were because the person was feeling jealous of a friend's car and the sales person used effective emotional leverage.

[0013] Unfortunately, some deployed machine learning models use techniques that are very opaque, black boxes that are hard or impossible to interpret and thus rationalizations are the only available choice for understanding the data. Further, the existing tools for explaining opaque models work for supervised learning. How can we apply them to unsupervised learning?

[0014] Obtaining rationalizations for unsupervised learning techniques can be done easily. Once the unsupervised learning technique has been performed, the values returned from the unsupervised technique can be appended into a new feature or target to the original data. First, consider an anomaly detection method that assigns a score to each training case (which may be a feature vector of some sort) indicating how anomalous the point is. This anomaly score is treated as the label for each case as if it were a supervised learning system and run through the explanation tool. The same technique can be applied to hard or soft clustering. Hard clustering is easy, given that each case belongs to exactly one cluster. The cluster ID can be used as the new label and, as before, the data can be run through an explanation tool. In soft clustering, each case may potentially belong to any number of clusters, potentially in a fractional manner. With soft clustering, each cluster can be one-hot encoded with each cluster being a label as if it were the output of a multilabel classification supervised learning system, and then these labels can be used in conjunction with the other features in the explanation system.

SYSTEMS FOR CONVICTION-BASED IMPUTATION IN COMPUTER-BASED REASONING SYSTEMS

[0015] FIG. 2 is a block diagram depicting example systems for rationalized clustering. Numerous devices and systems are coupled to a network 290. Network 290 can include the internet, a wide area network, a local area network, a Wi-Fi network, any other network or communication device described herein, and the like. Further, numerous of the systems and devices connected to 290 may have encrypted communication there between, VPNs, and or any other appropriate communication or security measure. System 200 includes a training and analysis system 210 coupled to network 290. The training and analysis system 210 may be used for collecting data related to systems 250 - 258 and creating machine learning models and / or creating rationalized clusters. Further, training and analysis system 210 may perform aspects of processes described herein. Control system 220 is also coupled to network 290. A control system 220 may control various of the systems 250 - 258. For example, a vehicle control 221 may control any of the vehicles 250 - 253, or the like. In some embodiments, there may be one or more network attached storages 230, 240. These storages 230, 240 may store training data, computer based reasoning models, updated computer based reasoning models, audit trails of imputed data, and the like. In some embodiments, training and analysis system 210 and / or control system 220 may store any needed data including computer based reasoning models locally on the system.

[0016] FIG. 2 depicts numerous systems 250 - 258 that may be controlled by a control system 220 or 221. For example, automobile 250, helicopter 251, submarine 252, boat 253, factory equipment 254, construction equipment 255, security equipment 256, oil pump 257, or warehouse equipment 258 may be controlled by a control system 220 or 221.

EXAMPLE PROCESSES FOR CONTROLLING SYSTEMS

[0017] FIG. 1 depicts an example process 100 for controlling a system. In some embodiments and at a high level, the process 100 proceeds by receiving or receiving 110 a computer-based reasoning model for controlling the system. The computer-based reasoning model may be one created using various processes. In some embodiments, the process 100 proceeds by receiving 120 a current context for the system, determining 130 an action to take based on the current context and the computer-based reasoning model, and causing 140 performance of the determined action (e.g., labelling an image, causing a vehicle to perform the turn, lane change, waypoint navigation, etc.). If operation of the system continues 150, then the process returns to receive 120 the current context, and otherwise discontinues 160 control of the system.

[0018] As discussed herein the various processes 100, etc. may run in parallel, in conjunction, together, or one process may be a subprocess of another. Further, any of the processes may run on the systems or hardware discussed herein. The features and steps of processes 100 could be used in combination and / or in different orders.

HARDWARE OVERVIEW

[0019] According to some embodiments, the techniques described herein are implemented by one or more special-purpose computing devices. The special-purpose computing devices may be hard-wired to perform the techniques, or may include digital electronic devices such as one or more application-specific integrated circuits (ASICs) or field programmable gate arrays (FPGAs) that are persistently programmed to perform the techniques, or may include one or more general purpose hardware processors programmed to perform the techniques pursuant to program instructions in firmware, memory, other storage, or a combination. Such special-purpose computing devices may also combine custom hard-wired logic, ASICs, or FPGAs with custom programming to accomplish the techniques. The special-purpose computing devices may be desktop computer systems, portable computer systems, handheld devices, networking devices or any other device that incorporates hard-wired and/or program logic to implement the techniques.

[0020] For example, FIG. 3 is a block diagram that illustrates a computer system 300 upon which an embodiment of the invention may be implemented. Computer system 300 includes a bus 302 or other communication mechanism for communicating information, and a hardware processor 304 coupled with bus 302 for processing information. Hardware processor 304 may be, for example, a general purpose microprocessor.

[0021] Computer system 300 also includes a main memory 306, such as a random access memory (RAM) or other dynamic storage device, coupled to bus 302 for storing information and instructions to be executed by processor 304. Main memory 306 also may be used for storing temporary variables or other intermediate information during execution of instructions to be executed by processor 304. Such instructions, when stored in non-transitory storage media accessible to processor 304, render computer system 300 into a special-purpose machine that is customized to perform the operations specified in the instructions.

[0022] Computer system 300 further includes a read only memory (ROM) 308 or other static storage device coupled to bus 302 for storing static information and instructions for processor 304. A storage device 310, such as a magnetic disk, optical disk, or solid-state drive is provided and coupled to bus 302 for storing information and instructions.

[0023] Computer system 300 may be coupled via bus 302 to a display 312, such as an OLED, LED or cathode ray tube (CRT), for displaying information to a computer user. An input device 314, including alphanumeric and other keys, is coupled to bus 302 for communicating information and command selections to processor 304. Another type of user input device is cursor control 316, such as a mouse, a trackball, or cursor direction keys for communicating direction information and command selections to processor 304 and for controlling cursor movement on display 312. This input device typically has two degrees of freedom in two axes, a first axis (e.g., x) and a second axis (e.g., y), that allows the device to specify positions in a plane. The input device 314 may also have multiple input modalities, such as multiple 2-axes controllers, and / or input buttons or keyboard. This allows a user to input along more than two dimensions simultaneously and / or control the input of more than one type of action.

[0024] Computer system 300 may implement the techniques described herein using customized hard-wired logic, one or more ASICs or FPGAs, firmware and/or program logic which in combination with the computer system causes or programs computer system 300 to be a special-purpose machine. According to some embodiments, the techniques herein are performed by computer system 300 in response to processor 304 executing one or more sequences of one or more instructions contained in main memory 306. Such instructions may be read into main memory 306 from another storage medium, such as storage device 310. Execution of the sequences of instructions contained in main memory 306 causes processor 304 to perform the process steps described herein. In alternative embodiments, hard-wired circuitry may be used in place of or in combination with software instructions.

[0025] The term "storage media" as used herein refers to any non-transitory media that store data and/or instructions that cause a machine to operate in a specific fashion. Such storage media may comprise non-volatile media and/or volatile media. Non-volatile media includes, for example, optical disks, magnetic disks, or solid-state drives, such as storage device 310. Volatile media includes dynamic memory, such as main memory 306. Common forms of storage media include, for example, a floppy disk, a flexible disk, hard disk, solid-state drive, magnetic tape, or any other magnetic data storage medium, a CD-ROM, any other optical data storage medium, any physical medium with patterns of holes, a RAM, a PROM, and EPROM, a FLASH-EPROM, NVRAM, any other memory chip or cartridge.

[0026] Storage media is distinct from but may be used in conjunction with transmission media. Transmission media participates in transferring information between storage media. For example, transmission media includes coaxial cables, copper wire and fiber optics, including the wires that comprise bus 302. Transmission media can also take the form of acoustic or light waves, such as those generated during radio-wave and infra-red data communications.

[0027] Various forms of media may be involved in carrying one or more sequences of one or more instructions to processor 304 for execution. For example, the instructions may initially be carried on a magnetic disk or solid-state drive of a remote computer. The remote computer can load the instructions into its dynamic memory and send the instructions over a telephone line using a modem. A modem local to computer system 300 can receive the data on the telephone line and use an infra-red transmitter to convert the data to an infra-red signal. An infra-red detector can receive the data carried in the infra-red signal and appropriate circuitry can place the data on bus 302. Bus 302 carries the data to main memory 306, from which processor 304 retrieves and executes the instructions. The instructions received by main memory 306 may optionally be stored on storage device 310 either before or after execution by processor 304.

[0028] Computer system 300 also includes a communication interface 318 coupled to bus 302. Communication interface 318 provides a two-way data communication coupling to a network link 320 that is connected to a local network 322. For example, communication interface 318 may be an integrated services digital network (ISDN) card, cable modem, satellite modem, or a modem to provide a data communication connection to a corresponding type of telephone line. As another example, communication interface 318 may be a local area network (LAN) card to provide a data communication connection to a compatible LAN. Wireless links may also be implemented. In any such implementation, communication interface 318 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information. Such a wireless link could be a Bluetooth, Bluetooth Low Energy (BLE), 802.11 WiFi connection, or the like.

[0029] Network link 320 typically provides data communication through one or more networks to other data devices. For example, network link 320 may provide a connection through local network 322 to a host computer 324 or to data equipment operated by an Internet Service Provider (ISP) 326. ISP 326 in turn provides data communication services through the world wide packet data communication network now commonly referred to as the "Internet" 328. Local network 322 and Internet 328 both use electrical, electromagnetic or optical signals that carry digital data streams. The signals through the various networks and the signals on network link 320 and through communication interface 318, which carry the digital data to and from computer system 300, are example forms of transmission media.

[0030] Computer system 300 can send messages and receive data, including program code, through the network(s), network link 320 and communication interface 318. In the Internet example, a server 330 might transmit a requested code for an application program through Internet 328, ISP 326, local network 322 and communication interface 318.

[0031] The received code may be executed by processor 304 as it is received, and/or stored in storage device 310, or other non-volatile storage for later execution.

[0032] In the foregoing specification, embodiments of the invention have been described with reference to numerous specific details that may vary from implementation to implementation. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense. The sole and exclusive indicator of the scope of the invention, and what is intended by the applicants to be the scope of the invention, is the literal and equivalent scope of the set of claims that issue from this application, in the specific form in which such claims issue, including any subsequent correction.

CLAIMS

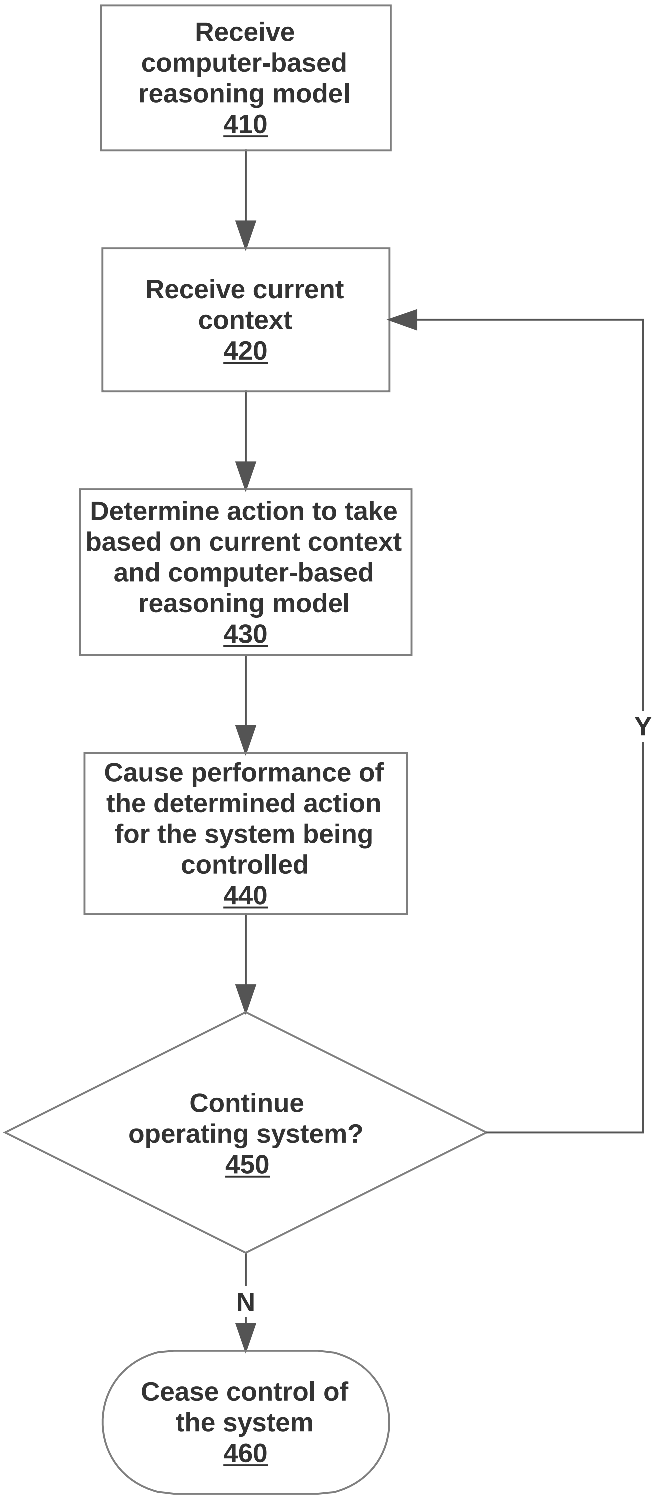
What is claimed is:

1. A method as substantially described herein.
2. One or more computer readable media containing instructions, which, when executed, perform the method of Claim 1.
3. A system comprising as substantially described herein.

ABSTRACT OF THE DISCLOSURE

Techniques are provided for rationalized clustering in machine learning.

FIG. 4


Receive computer-based reasoning model 410
Receive current context 420
Determine action to take based on current context and computer-based reasoning model 430
Cause performance of the determined action for the system being controlled 440
Continue operating system? 450
Y
N
Cease control of the system 460

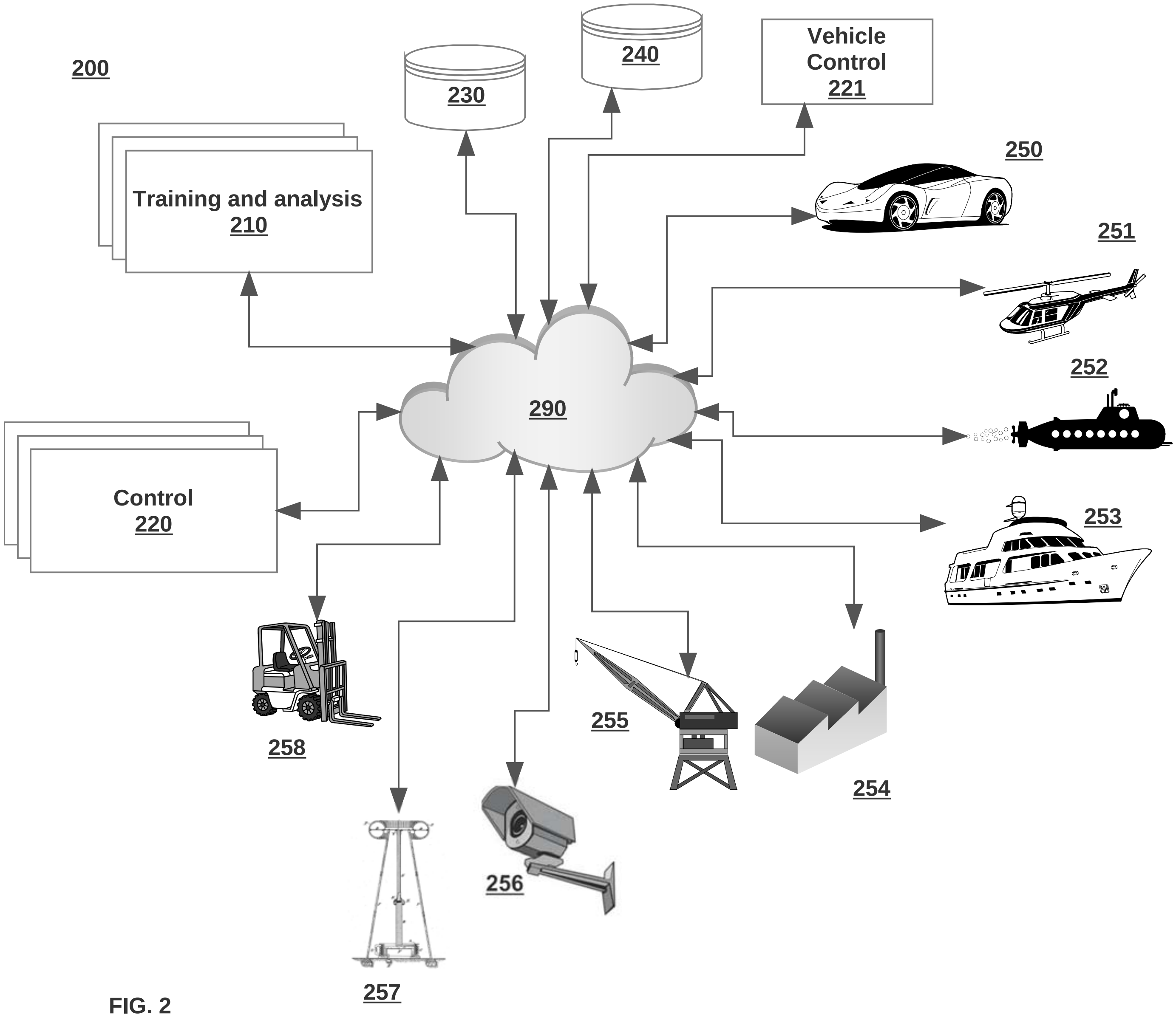
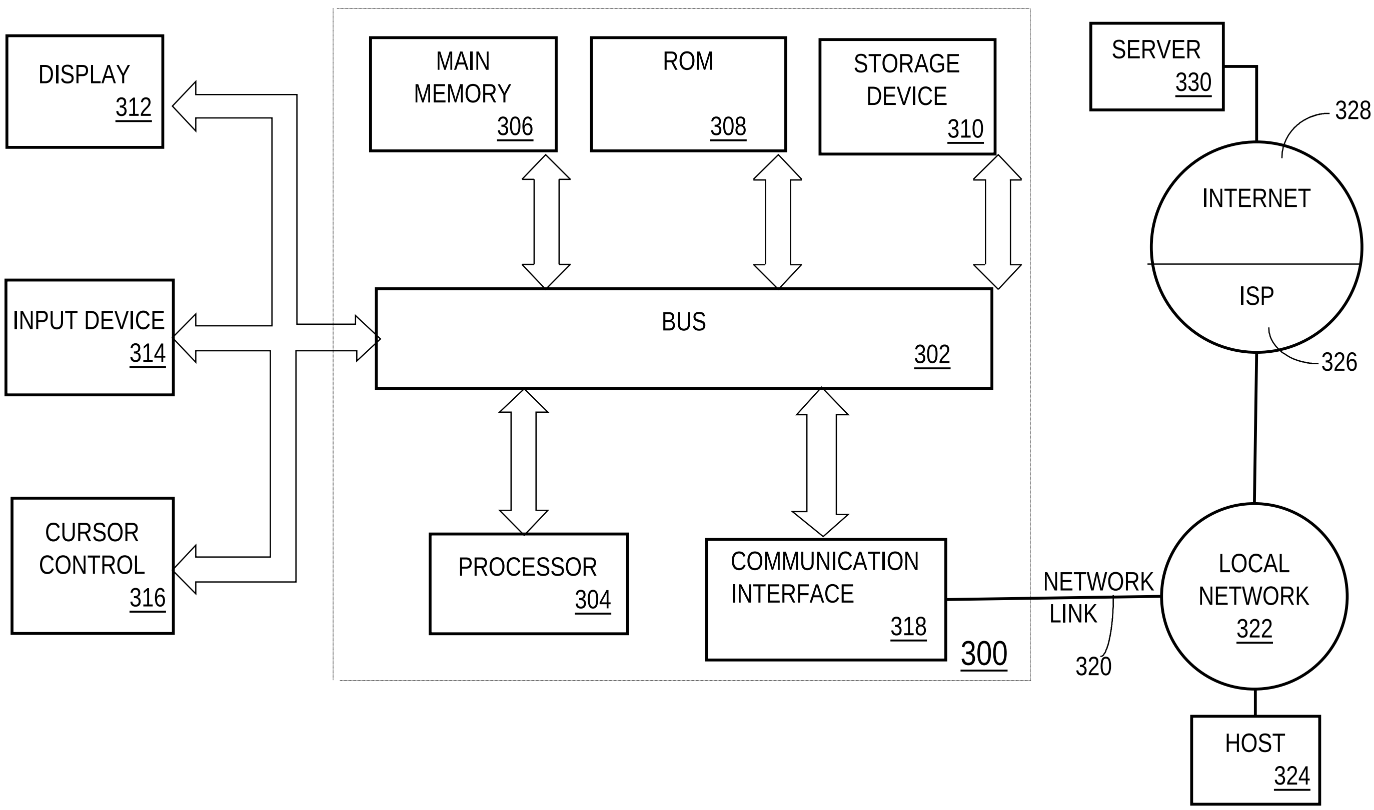

200
230
240
Vehicle
Control
221
250
Training and analysis
210
251
252
290
Control
220
253
258
255
254
256
257

FIG. 2

FIG. 3



DISPLAY
312
INPUT DEVICE
314
CURSOR
CONTROL
316
MAIN
MEMORY
306
ROM
308
STORAGE
DEVICE
310
BUS
302
PROCESSOR
304
COMMUNICATION
INTERFACE
318
300
SERVER
330
328
INTERNET
ISP
326
NETWORK
LINK
320
LOCAL
NETWORK
322
HOST
324