



- (51) International Patent Classification: Not classified
- (21) International Application Number:
PCT/US2014/026873
- (22) International Filing Date:
13 March 2014 (13.03.2014)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
61/785,565 14 March 2013 (14.03.2013) US
- (71) Applicant: APPLE INC. [US/US]; 1 Infinite Loop, Cupertino, CA 95014 (US).
- (72) Inventor: SKINDER, Jason, A.; C/o Apple Inc., 1 Infinite Loop, Ms 36-2PAT, Cupertino, CA 95014 (US).
- (74) Agents: EIDE, Christopher, B. et al.; Morrison & Foerster LLP, 755 Page Mill Road, Palo Alto, CA 94304-1018 (US).
- (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, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, 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, 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, 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:

— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

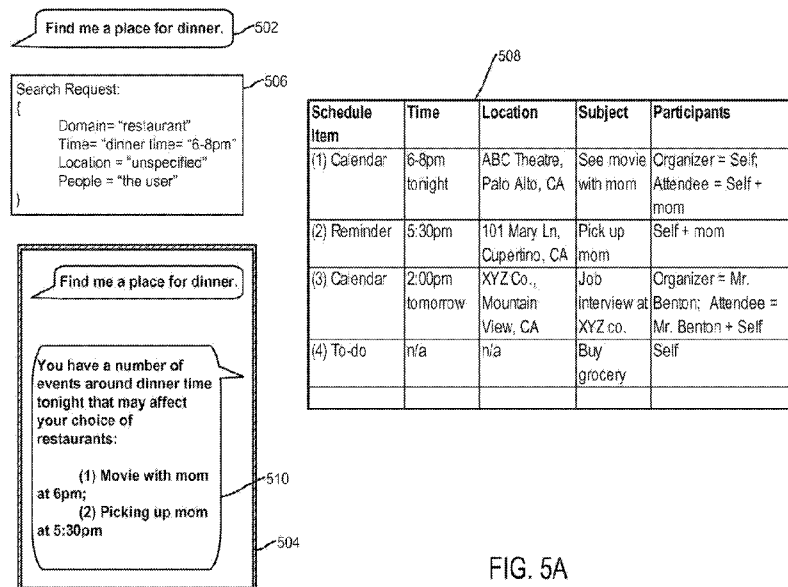
(54) Title: REFINING A SEARCH BASED ON SCHEDULE ITEMS

FIG. 5A

(57) **Abstract:** A method includes receiving a search request from a user, the search request specifying one or more search criteria for selecting one or more selection items from a respective selection domain; identifying one or more schedule items relevant to the search request; augmenting the search request based on information contained in at least one of the identified schedule items; and presenting the augmented search request or search results retrieved based on the augmented search request to the user.

REFINING A SEARCH BASED ON SCHEDULE ITEMS

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This Application claims the benefit of U.S. Provisional Application No. 61/785,565, filed on March 14, 2013, entitled REFINING A SEARCH BASED ON SCHEDULE ITEMS, which is hereby incorporated by reference in its entirety for all purposes.

TECHNICAL FIELD

[0002] The disclosed embodiments relate generally to digital assistants, and more specifically to digital assistants that intelligently perform searches in one or more selection domains based on a user request.

BACKGROUND

[0003] Just like human personal assistants, digital assistants or virtual assistants can perform requested tasks and provide requested advice, information, or services. An assistant's ability to fulfill a user's request is dependent on the assistant's correct comprehension of the request or instruction. Recent advances in natural language processing have enabled users to interact with digital assistants using natural language, in spoken or textual forms, rather than employing a conventional user interface (*e.g.*, menus or programmed commands). Such digital assistants can interpret the user's input to infer the user's intent; translate the inferred intent into actionable tasks and parameters; execute operations or deploy services to perform the tasks; and produce outputs that are intelligible to the user. Ideally, the outputs produced by a digital assistant should fulfill the user's intent expressed during the natural language interaction between the user and the digital assistant. A digital assistant can perform searches in a selection domain (*e.g.*, a restaurant domain, etc.) and present qualifying selection items (*e.g.*, restaurants) in response to a search request received from a user.

[0004] The ability of a digital assistant system to produce satisfactory responses to user requests depends on the natural language processing, knowledge base, and artificial intelligence implemented by the system. A well-designed response procedure can improve a user's experience in interacting with the system and promote the user's confidence in the system's services and capabilities.

SUMMARY

[0005] A conventional digital assistant system responds to a user's search request based on the search criteria provided in the user's direct input, and does not take into account of the user's schedule items that may be relevant to the search request. The embodiments disclosed herein provide methods, systems, computer readable storage medium and user interfaces for a digital assistant to intelligently identify schedule items relevant to a search request received from a user, and use additional information obtained from these relevant schedule items to further refine the search request received from the user. Various techniques for how to identify the relevant schedule items, and how to utilize the information in the identified schedule items can be used to improve the responsiveness and efficiency of the digital assistant, making the service provided by the digital assistant more user friendly, efficient, and intuitive.

[0006] Accordingly, some embodiments provide a method for operating a digital assistant, the method including, at a device including one or more processors and memory storing one or more programs: receiving a search request from a user, the search request specifying one or more search criteria for selecting one or more selection items from a respective selection domain; identifying one or more schedule items relevant to the search request; augmenting the search request based on information contained in at least one of the identified schedule items; and presenting the augmented search request or search results retrieved based on the augmented search request to the user.

[0007] In some embodiments, the relevant schedule items include one or more calendar items from a calendar associated with the user.

[0008] In some embodiments, the relevant schedule items further include one or more calendar items from a calendar that has been shared with the user by another user.

[0009] In some embodiments, the relevant schedule items further include one or more reminder items in the user's electronic schedule.

[0010] In some embodiments, the relevant schedule items further include one or more to-dos in the user's electronic schedule.

[0011] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying one or more schedule items that are relevant to the search request in at least one of location, time, associated people, and subject matter.

[0012] In some embodiments, augmenting the search request based on the information contained in the at least one of the identified schedule items includes: establishing, based on the information contained in the at least one of the identified schedule items, a refinement on an existing criterion specified in the search request; and modifying the search request based on the refinement on the existing criterion.

[0013] In some embodiments, augmenting the search request based on the information contained in the at least one of the identified schedule items includes: establishing, based on the information contained in the at least one of the identified schedule items, a new search criterion that has not been specified in the search request; and modifying the search request to include the new search criterion.

[0014] In some embodiments, augmenting the search request based on the information contained in the at least one of the identified schedule items includes: establishing, based on the information contained in the at least one of the identified schedule items, a new preference order that has not been specified in the search request; and modifying the search request to include the new preference order.

[0015] In some embodiments, augmenting the search request further includes: adding a new time criterion or modifying an existing time criterion based on an event time associated with the at least one of the identified schedule items.

[0016] In some embodiments, adding the new time criterion or modifying the existing time criterion is further based on an event location associated with at least one of the identified schedule items.

[0017] In some embodiments, augmenting the search request further includes: adding a new location criterion or modifying an existing location criterion based on a respective location associated with at least one of the identified schedule items.

[0018] In some embodiments, adding the new location criterion or modifying the existing location criterion is further based on an event time associated with at least one of the identified schedule items.

[0019] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying a respective group of relevant schedule items, wherein each of the respective group of relevant schedule items is associated with a respective time window overlapping with a present time window of a given duration.

[0020] In some embodiments, identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes neither any distant time criterion nor any distant location criterion.

[0021] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective local geographic region that meets a location criterion specified in the search request.

[0022] In some embodiments, identifying the additional group of relevant schedule items is performed in accordance with a determination that the search request includes a local location criterion and does not include any time criterion.

[0023] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective local geographic region that matches a current location of the user.

[0024] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective group of attendees that overlaps with a group of relevant people specified by a respective person criterion in the search request.

[0025] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective subject matter that relates to a relevant subject matter specified by a respective selection domain criterion in the search request.

[0026] In some embodiments, identifying the additional group of relevant schedule items is performed in accordance with a determination that the search request does not include any location or time criterion.

[0027] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying a respective group of relevant schedule items based on a respective distant time criterion specified in the search request, wherein each of the respective group of relevant schedule items is associated with a respective time window overlapping with a relevant time range specified by the respective distant time criterion.

[0028] In some embodiments, identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes the respective distant time criterion and does not include any distant location criterion.

[0029] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying a respective group of relevant schedule items based on a respective distant location criterion specified in the search request, wherein each of the respective group of relevant schedule items is associated with a respective geographic location overlapping with a relevant geographic location specified by the respective distant location criterion.

[0030] In some embodiments, identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes the respective distant location criterion and does not include any distant time criterion.

[0031] In some embodiments, identifying the one or more schedule items relevant to the search request further includes: identifying a respective group of relevant schedule items based on the respective distant time criterion and the respective distant location criterion of the search request, wherein each of the respective group of relevant schedule items is associated with (1) a respective time window overlapping with the relevant time range specified by the respective distant time criterion, and (2) a respective geographic location overlapping with the relevant geographic location specified by the respective distant location criterion.

[0032] In some embodiments, identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes the respective distant time criterion and the respective distant location criterion.

[0033] In some embodiments, identifying the one or more schedule items relevant to the respective selection domain or the one or more selection criteria further includes: determining whether the one or more search criteria include any distant time criterion or any distant location criterion, wherein a distant time criterion specifies a relevant time range outside of a present time window of a given duration, and a distant location criterion specifies a relevant geographic region outside of a local geographic region of the user; and based on an outcome of the determination, selecting a respective set of rules for identifying an initial set of potentially relevant schedule items from which the one or more relevant schedule items are identified for the search request.

[0034] In some embodiments, at a device having one or more processors and memory, a method of operating a digital assistant includes: receiving an input (*e.g.*, a speech input) from a user; processing the speech input to infer a respective user intent, the user intent being embodied in one or more search criteria for a search request; based on the search request, generating a constrained selection task for selecting one or more selection items from a respective selection domain based on the one or more search criteria; identifying one or more schedule items relevant to the search request based on the respective selection domain and the one or more search criteria; augmenting the search request based on information contained in at least one of the identified schedule items; and presenting the augmented search request or search results retrieved based on the augmented search request to the user.

[0035] In some embodiments, the method further implements features of any combination of the methods described above and in the remainder of this specification.

[0036] The above embodiments, and other embodiments described in this specification may help realize one or more of the following advantages. In some embodiments, user intent may be clarified during the interaction between the user and the digital assistant when the relevant schedule items are presented to the user. In some embodiments, scheduling conflicts may be identified for the user, such that the user can alter the search request or rearrange his or her schedule based on the information presented by the digital assistant. In some embodiments, the digital assistant identifies schedule items that may create some synergy (*e.g.*, convenience, efficiency, etc.) when performed in conjunction with the action hind the intent of the search request, and making the user's actions more rewarding or more pleasant. In some embodiments, the particular manners in which the digital assistant identifies the relevant schedule items, presenting the relevant schedule items, and/or making suggestions

based on the relevant schedule items are designed such that the interaction between the user and the digital assistant is smooth, efficient, and less cumbersome.

[0037] The details of one or more embodiments of the subject matter described in this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the subject matter will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0038] FIG. 1 is a block diagram illustrating an environment in which a digital assistant operates in accordance with some embodiments.

[0039] FIG. 2 is a block diagram illustrating a digital assistant client system in accordance with some embodiments.

[0040] FIG. 3A is a block diagram illustrating a digital assistant system or a server portion thereof in accordance with some embodiments.

[0041] FIG. 3B is a block diagram illustrating functions of the digital assistant shown in FIG. 3A in accordance with some embodiments.

[0042] FIG. 3C is a diagram of a portion of an ontology in accordance with some embodiments.

[0043] FIGS. 4A-4B are a flow chart of an exemplary process for intelligently augmenting a search request based on relevant schedule items in accordance with some embodiments.

[0044] FIGS. 5A-5L are exemplary user interfaces and interactions illustrating a process for intelligently augmenting a search request based on relevant schedule items in accordance with some embodiments.

[0045] FIGS. 6A-6C are a flow chart of an exemplary process for selecting a preliminary set of potentially relevant schedule items for a search request in accordance with some embodiments.

[0046] Like reference numerals refer to corresponding parts throughout the drawings.

DESCRIPTION OF EMBODIMENTS

[0047] FIG. 1 is a block diagram of an operating environment 100 of a digital assistant according to some embodiments. The terms “digital assistant,” “virtual assistant,” “intelligent automated assistant,” or “automatic digital assistant,” refer to any information processing system that interprets natural language input in spoken and/or textual form to infer user intent, and performs actions based on the inferred user intent. For example, to act on an inferred user intent, the system can perform one or more of the following: identifying a task flow with steps and parameters designed to accomplish the inferred user intent, inputting specific requirements from the inferred user intent into the task flow; executing the task flow by invoking programs, methods, services, APIs, or the like; and generating output responses to the user in an audible (*e.g.* speech) and/or visual form.

[0048] Specifically, a digital assistant is capable of accepting a user request at least partially in the form of a natural language command, request, statement, narrative, and/or inquiry. Typically, the user request seeks either an informational answer or performance of a task by the digital assistant. A satisfactory response to the user request is either provision of the requested informational answer, performance of the requested task, or a combination of the two. For example, a user may ask the digital assistant a question, such as “Where am I right now?” Based on the user’s current location, the digital assistant may answer, “You are in Central Park.” The user may also request the performance of a task, for example, “Please remind me to call mom at 4pm today.” In response, the digital assistant may acknowledge the request and then creates an appropriate reminder item in the user’s electronic schedule. During performance of a requested task, the digital assistant sometimes interacts with the user in a continuous dialogue involving multiple exchanges of information over an extended period of time. There are numerous other ways of interacting with a digital assistant to request information or performance of various tasks. In addition to providing verbal responses and taking programmed actions, the digital assistant also provides responses in other visual or audio forms, *e.g.*, as text, alerts, music, videos, animations, etc.

[0049] An example of a digital assistant is described in Applicant’s U.S. Utility Application Serial No. 12/987,982 for “Intelligent Automated Assistant,” filed January 10, 2011, the entire disclosure of which is incorporated herein by reference.

[0050] As shown in FIG. 1, in some embodiments, a digital assistant is implemented according to a client-server model. The digital assistant includes a client-side portion 102a,

102b (hereafter “DA client 102”) executed on a user device 104a, 104b, and a server-side portion 106 (hereafter “DA server 106”) executed on a server system 108. The DA client 102 communicates with the DA server 106 through one or more networks 110. The DA client 102 provides client-side functionalities such as user-facing input and output processing and communications with the DA-server 106. The DA server 106 provides server-side functionalities for any number of DA-clients 102 each residing on a respective user device 104.

[0051] In some embodiments, the DA server 106 includes a client-facing I/O interface 112, one or more processing modules 114, data and models 116, and an I/O interface to external services 118. The client-facing I/O interface facilitates the client-facing input and output processing for the digital assistant server 106. The one or more processing modules 114 utilize the data and models 116 to determine the user’s intent based on natural language input and perform task execution based on inferred user intent. In some embodiments, the DA-server 106 communicates with external services 120 through the network(s) 110 for task completion or information acquisition. The I/O interface to external services 118 facilitates such communications.

[0052] Examples of the user device 104 include, but are not limited to, a handheld computer, a personal digital assistant (PDA), a tablet computer, a laptop computer, a desktop computer, a cellular telephone, a smart phone, an enhanced general packet radio service (EGPRS) mobile phone, a media player, a navigation device, a game console, a television, a remote control, or a combination of any two or more of these data processing devices or other data processing devices. More details on the user device 104 are provided in reference to an exemplary user device 104 shown in FIG. 2.

[0053] Examples of the communication network(s) 110 include local area networks (“LAN”) and wide area networks (“WAN”), *e.g.*, the Internet. The communication network(s) 110 are, optionally, implemented using any known network protocol, including various wired or wireless protocols, such as *e.g.*, Ethernet, Universal Serial Bus (USB), FIREWIRE, Global System for Mobile Communications (GSM), Enhanced Data GSM Environment (EDGE), code division multiple access (CDMA), time division multiple access (TDMA), Bluetooth, Wi-Fi, voice over Internet Protocol (VoIP), Wi-MAX, or any other suitable communication protocol.

[0054] The server system 108 is implemented on one or more standalone data processing apparatus or a distributed network of computers. In some embodiments, the server system 108 also employs various virtual devices and/or services of third party service providers (*e.g.*, third-party cloud service providers) to provide the underlying computing resources and/or infrastructure resources of the server system 108.

[0055] Although the digital assistant shown in FIG. 1 includes both a client-side portion (*e.g.*, the DA-client 102) and a server-side portion (*e.g.*, the DA-server 106), in some embodiments, the functions of a digital assistant is implemented as a standalone application installed on a user device. In addition, the divisions of functionalities between the client and server portions of the digital assistant can vary in different embodiments. For example, in some embodiments, the DA client is a thin-client that provides only user-facing input and output processing functions, and delegates all other functionalities of the digital assistant to a backend server.

[0056] FIG. 2 is a block diagram of a user-device 104 in accordance with some embodiments. The user device 104 includes a memory interface 202, one or more processors 204, and a peripherals interface 206. The various components in the user device 104 are coupled by one or more communication buses or signal lines. The user device 104 includes various sensors, subsystems, and peripheral devices that are coupled to the peripherals interface 206. The sensors, subsystems, and peripheral devices gather information and/or facilitate various functionalities of the user device 104.

[0057] For example, a motion sensor 210, a light sensor 212, and a proximity sensor 214 are coupled to the peripherals interface 206 to facilitate orientation, light, and proximity sensing functions. One or more other sensors 216, such as a positioning system (*e.g.*, GPS receiver), a temperature sensor, a biometric sensor, a gyro, a compass, an accelerometer, and the like, are also connected to the peripherals interface 206, to facilitate related functionalities.

[0058] In some embodiments, a camera subsystem 220 and an optical sensor 222 are utilized to facilitate camera functions, such as taking photographs and recording video clips. Communication functions are facilitated through one or more wired and/or wireless communication subsystems 224, which can include various communication ports, radio frequency receivers and transmitters, and/or optical (*e.g.*, infrared) receivers and transmitters. An audio subsystem 226 is coupled to speakers 228 and a microphone 230 to facilitate voice-

enabled functions, such as voice recognition, voice replication, digital recording, and telephony functions.

[0059] In some embodiments, an I/O subsystem 240 is also coupled to the peripheral interface 206. The I/O subsystem 240 includes a touch screen controller 242 and/or other input controller(s) 244. The touch-screen controller 242 is coupled to a touch screen 246. The touch screen 246 and the touch screen controller 242 can, for example, detect contact and movement or break thereof using any of a plurality of touch sensitivity technologies, such as capacitive, resistive, infrared, surface acoustic wave technologies, proximity sensor arrays, and the like. The other input controller(s) 244 can be coupled to other input/control devices 248, such as one or more buttons, rocker switches, thumb-wheel, infrared port, USB port, and/or a pointer device such as a stylus.

[0060] In some embodiments, the memory interface 202 is coupled to memory 250. The memory 250 can include high-speed random access memory and/or non-volatile memory, such as one or more magnetic disk storage devices, one or more optical storage devices, and/or flash memory (*e.g.*, NAND, NOR).

[0061] In some embodiments, the memory 250 stores an operating system 252, a communication module 254, a graphical user interface module 256, a sensor processing module 258, a phone module 260, and applications 262. The operating system 252 includes instructions for handling basic system services and for performing hardware dependent tasks. The communication module 254 facilitates communicating with one or more additional devices, one or more computers and/or one or more servers. The graphical user interface module 256 facilitates graphic user interface processing. The sensor processing module 258 facilitates sensor-related processing and functions. The phone module 260 facilitates phone-related processes and functions. The application module 262 facilitates various functionalities of user applications, such as electronic-messaging, web browsing, media processing, Navigation, imaging and/or other processes and functions.

[0062] As described in this specification, the memory 250 also stores client-side digital assistant instructions (*e.g.*, in a digital assistant client module 264) and various user data 266 (*e.g.*, user-specific vocabulary data, preference data, and/or other data such as the user's electronic address book, to-do lists, shopping lists, etc.) to provide the client-side functionalities of the digital assistant.

[0063] In various embodiments, the digital assistant client module 264 is capable of accepting voice input (*e.g.*, speech input), text input, touch input, and/or gestural input through various user interfaces (*e.g.*, the I/O subsystem 244) of the user device 104. The digital assistant client module 264 is also capable of providing output in audio (*e.g.*, speech output), visual, and/or tactile forms. For example, output can be provided as voice, sound, alerts, text messages, menus, graphics, videos, animations, vibrations, and/or combinations of two or more of the above. During operation, the digital assistant client module 264 communicates with the digital assistant server using the communication subsystems 224.

[0064] In some embodiments, the digital assistant client module 264 utilizes the various sensors, subsystems and peripheral devices to gather additional information from the surrounding environment of the user device 104 to establish a context associated with a user, the current user interaction, and/or the current user input. In some embodiments, the digital assistant client module 264 provides the context information or a subset thereof with the user input to the digital assistant server to help infer the user's intent. In some embodiments, the digital assistant also uses the context information to determine how to prepare and delivery outputs to the user.

[0065] In some embodiments, the context information that accompanies the user input includes sensor information, *e.g.*, lighting, ambient noise, ambient temperature, images or videos of the surrounding environment, etc. In some embodiments, the context information also includes the physical state of the device, *e.g.*, device orientation, device location, device temperature, power level, speed, acceleration, motion patterns, cellular signals strength, etc. In some embodiments, information related to the software state of the user device 106, *e.g.*, running processes, installed programs, past and present network activities, background services, error logs, resources usage, etc., of the user device 104 are provided to the digital assistant server as context information associated with a user input.

[0066] In some embodiments, the DA client module 264 selectively provides information (*e.g.*, user data 266) stored on the user device 104 in response to requests from the digital assistant server. In some embodiments, the digital assistant client module 264 also elicits additional input from the user via a natural language dialogue or other user interfaces upon request by the digital assistant server 106. The digital assistant client module 264 passes the additional input to the digital assistant server 106 to help the digital assistant server 106 in intent inference and/or fulfillment of the user's intent expressed in the user request.

[0067] In various embodiments, the memory 250 includes additional instructions or fewer instructions. Furthermore, various functions of the user device 104 may be implemented in hardware and/or in firmware, including in one or more signal processing and/or application specific integrated circuits.

[0068] FIG. 3A is a block diagram of an example digital assistant system 300 in accordance with some embodiments. In some embodiments, the digital assistant system 300 is implemented on a standalone computer system. In some embodiments, the digital assistant system 300 is distributed across multiple computers. In some embodiments, some of the modules and functions of the digital assistant are divided into a server portion and a client portion, where the client portion resides on a user device (*e.g.*, the user device 104) and communicates with the server portion (*e.g.*, the server system 108) through one or more networks, *e.g.*, as shown in FIG. 1. In some embodiments, the digital assistant system 300 is an embodiment of the server system 108 (and/or the digital assistant server 106) shown in FIG. 1. It should be noted that the digital assistant system 300 is only one example of a digital assistant system, and that the digital assistant system 300 may have more or fewer components than shown, may combine two or more components, or may have a different configuration or arrangement of the components. The various components shown in FIG. 3A may be implemented in hardware, software instructions for execution by one or more processors, firmware, including one or more signal processing and/or application specific integrated circuits, or a combination of thereof.

[0069] The digital assistant system 300 includes memory 302, one or more processors 304, an input/output (I/O) interface 306, and a network communications interface 308. These components communicate with one another over one or more communication buses or signal lines 310.

[0070] In some embodiments, the memory 302 includes a non-transitory computer readable medium, such as high-speed random access memory and/or a non-volatile computer readable storage medium (*e.g.*, one or more magnetic disk storage devices, flash memory devices, or other non-volatile solid-state memory devices).

[0071] In some embodiments, the I/O interface 306 couples input/output devices 316 of the digital assistant system 300, such as displays, keyboards, touch screens, and microphones, to the user interface module 322. The I/O interface 306, in conjunction with the user interface module 322, receive user inputs (*e.g.*, voice input, keyboard inputs, touch

inputs, etc.) and process them accordingly. In some embodiments, *e.g.*, when the digital assistant is implemented on a standalone user device, the digital assistant system 300 includes any of the components and I/O and communication interfaces described with respect to the user device 104 in FIG. 2. In some embodiments, the digital assistant system 300 represents the server portion of a digital assistant implementation, and interacts with the user through a client-side portion residing on a user device (*e.g.*, the user device 104 shown in FIG. 2).

[0072] In some embodiments, the network communications interface 308 includes wired communication port(s) 312 and/or wireless transmission and reception circuitry 314. The wired communication port(s) receive and send communication signals via one or more wired interfaces, *e.g.*, Ethernet, Universal Serial Bus (USB), FIREWIRE, etc. The wireless circuitry 314 receives and sends RF signals and/or optical signals from/to communications networks and other communications devices. The wireless communications, optionally, use any of a plurality of communications standards, protocols and technologies, such as GSM, EDGE, CDMA, TDMA, Bluetooth, Wi-Fi, VoIP, Wi-MAX, or any other suitable communication protocol. The network communications interface 308 enables communication between the digital assistant system 300 with networks, such as the Internet, an intranet and/or a wireless network, such as a cellular telephone network, a wireless local area network (LAN) and/or a metropolitan area network (MAN), and other devices.

[0073] In some embodiments, memory 302, or the computer readable storage media of memory 302, stores programs, modules, instructions, and data structures including all or a subset of: an operating system 318, a communications module 320, a user interface module 322, one or more applications 324, and a digital assistant module 326. The one or more processors 304 execute these programs, modules, and instructions, and reads/writes from/to the data structures.

[0074] The operating system 318 (*e.g.*, Darwin, RTXC, LINUX, UNIX, OS X, WINDOWS, or an embedded operating system such as VxWorks) includes various software components and/or drivers for controlling and managing general system tasks (*e.g.*, memory management, storage device control, power management, etc.) and facilitates communications between various hardware, firmware, and software components.

[0075] The communications module 320 facilitates communications between the digital assistant system 300 with other devices over the network communications interface 308. For example, the communication module 320, optionally, communicates with the communication

interface 254 of the device 104 shown in FIG. 2. The communications module 320 also includes various components for handling data received by the wireless circuitry 314 and/or wired communications port 312.

[0076] The user interface module 322 receives commands and/or inputs from a user via the I/O interface 306 (*e.g.*, from a keyboard, touch screen, pointing device, controller, and/or microphone), and generates user interface objects on a display. The user interface module 322 also prepares and delivers outputs (*e.g.*, speech, sound, animation, text, icons, vibrations, haptic feedback, and light, etc.) to the user via the I/O interface 306 (*e.g.*, through displays, audio channels, speakers, and touch-pads, etc.).

[0077] The applications 324 include programs and/or modules that are configured to be executed by the one or more processors 304. For example, if the digital assistant system is implemented on a standalone user device, the applications 324, optionally, include user applications, such as games, a calendar application, a navigation application, or an email application. If the digital assistant system 300 is implemented on a server farm, the applications 324, optionally, include resource management applications, diagnostic applications, or scheduling applications, for example.

[0078] The memory 302 also stores the digital assistant module (or the server portion of a digital assistant) 326. In some embodiments, the digital assistant module 326 includes the following sub-modules, or a subset or superset thereof: an input/output processing module 328, a speech-to-text (STT) processing module 330, a natural language processing module 332, a dialogue flow processing module 334, a task flow processing module 336, a service processing module 338, and a search augmentation module 340. Each of these modules has access to one or more of the following data and models of the digital assistant 326, or a subset or superset thereof: ontology 360, vocabulary index 344, user data 348, task flow models 354, service models 356, and user schedule data 358.

[0079] In some embodiments, using the processing modules, data, and models implemented in the digital assistant module 326, the digital assistant performs at least some of the following: identifying a user's intent expressed in a natural language input received from the user; actively eliciting and obtaining information needed to fully infer the user's intent (*e.g.*, by disambiguating words, names, intentions, etc.); determining the task flow for fulfilling the inferred intent; and executing the task flow to fulfill the inferred intent. In this

specifications, more details regarding the search augmentation module 340 and its use of the user schedule data 358 are provided later in FIGS. 4A-6C and accompanying descriptions.

[0080] In some embodiments, as shown in FIG. 3B, the I/O processing module 328 interacts with the user through the I/O devices 316 in FIG. 3A or with a user device (*e.g.*, a user device 104 in FIG. 1) through the network communications interface 308 in FIG. 3A to obtain user input (*e.g.*, a speech input) and to provide responses (*e.g.*, as speech outputs) to the user input. The I/O processing module 328, optionally, obtains context information associated with the user input from the user device, along with or shortly after the receipt of the user input. The context information includes user-specific data, vocabulary, and/or preferences relevant to the user input. In some embodiments, the context information also includes software and hardware states of the device (*e.g.*, the user device 104 in FIG. 1) at the time the user request is received, and/or information related to the surrounding environment of the user at the time that the user request was received. In some embodiments, the I/O processing module 328 also sends follow-up questions to, and receives answers from, the user regarding the user request. When a user request is received by the I/O processing module 328 and the user request contains a speech input, the I/O processing module 328 forwards the speech input to the speech-to-text (STT) processing module 330 for speech-to-text conversions.

[0081] The speech-to-text processing module 330 receives speech input (*e.g.*, a user utterance captured in a voice recording) through the I/O processing module 328. In some embodiments, the speech-to-text processing module 330 uses various acoustic and language models to recognize the speech input as a sequence of phonemes, and ultimately, a sequence of words or tokens written in one or more languages. The speech-to-text processing module 330 can be implemented using any suitable speech recognition techniques, acoustic models, and language models, such as Hidden Markov Models, Dynamic Time Warping (DTW)-based speech recognition, and other statistical and/or analytical techniques. In some embodiments, the speech-to-text processing can be performed at least partially by a third party service or on the user's device. Once the speech-to-text processing module 330 obtains the result of the speech-to-text processing, *e.g.*, a sequence of words or tokens, it passes the result to the natural language processing module 332 for intent inference.

[0082] More details on the speech-to-text processing are described in U.S. Utility Application Serial No. 13/236,942 for “Consolidating Speech Recognition Results,” filed on September 20, 2011, the entire disclosure of which is incorporated herein by reference.

[0083] The natural language processing module 332 (“natural language processor”) of the digital assistant takes the sequence of words or tokens (“token sequence”) generated by the speech-to-text processing module 330, and attempts to associate the token sequence with one or more “actionable intents” recognized by the digital assistant. An “actionable intent” represents a task that can be performed by the digital assistant, and has an associated task flow implemented in the task flow models 354. The associated task flow is a series of programmed actions and steps that the digital assistant takes in order to perform the task. The scope of a digital assistant’s capabilities is dependent on the number and variety of task flows that have been implemented and stored in the task flow models 354, or in other words, on the number and variety of “actionable intents” that the digital assistant recognizes. The effectiveness of the digital assistant, however, is also dependent on the assistant’s ability to infer the correct “actionable intent(s)” from the user request expressed in natural language.

[0084] In some embodiments, in addition to the sequence of words or tokens obtained from the speech-to-text processing module 330, the natural language processor 332 also receives context information associated with the user request, *e.g.*, from the I/O processing module 328. The natural language processor 332, optionally, uses the context information to clarify, supplement, and/or further define the information contained in the token sequence received from the speech-to-text processing module 330. The context information includes, for example, user preferences, hardware and/or software states of the user device, sensor information collected before, during, or shortly after the user request, prior interactions (*e.g.*, dialogue) between the digital assistant and the user, and the like.

[0085] In some embodiments, the natural language processing is based on ontology 360. The ontology 360 is a hierarchical structure containing many nodes, each node representing either an “actionable intent” or a “property” relevant to one or more of the “actionable intents” or other “properties.” As noted above, an “actionable intent” represents a task that the digital assistant is capable of performing, *i.e.*, it is “actionable” or can be acted on. A “property” represents a parameter associated with an actionable intent, a domain concept or entity, or a sub-aspect of another property. A linkage between an actionable intent node and

a property node in the ontology 360 defines how a parameter represented by the property node pertains to the task represented by the actionable intent node.

[0086] In some embodiments, the ontology 360 is made up of actionable intent nodes and property nodes. Within the ontology 360, each actionable intent node is linked to one or more property nodes either directly or through one or more intermediate property nodes. Similarly, each property node is linked to one or more actionable intent nodes either directly or through one or more intermediate property nodes. For example, as shown in FIG. 3C, the ontology 360 may include a “restaurant reservation” node (*i.e.*, an actionable intent node). Property node “restaurant,” (a domain entity represented by a property node) and property nodes “date/time” (for the reservation) and “party size” are each directly linked to the actionable intent node (*i.e.*, the “restaurant reservation” node). In addition, property nodes “cuisine,” “price range,” “phone number,” and “location” are sub-nodes of the property node “restaurant,” and are each linked to the “restaurant reservation” node (*i.e.*, the actionable intent node) through the intermediate property node “restaurant.” For another example, as shown in FIG. 3C, the ontology 360 may also include a “set reminder” node (*i.e.*, another actionable intent node). Property nodes “date/time” (for the setting the reminder) and “subject” (for the reminder) are each linked to the “set reminder” node. Since the property “date/time” is relevant to both the task of making a restaurant reservation and the task of setting a reminder, the property node “date/time” is linked to both the “restaurant reservation” node and the “set reminder” node in the ontology 360.

[0087] An actionable intent node, along with its linked concept nodes, may be described as a “domain.” In the present discussion, each domain is associated with a respective actionable intent, and refers to the group of nodes (and the relationships therebetween) associated with the particular actionable intent. For example, the ontology 360 shown in FIG. 3C includes an example of a restaurant reservation domain 362 and an example of a reminder domain 364 within the ontology 360. The restaurant reservation domain includes the actionable intent node “restaurant reservation,” property nodes “restaurant,” “date/time,” and “party size,” and sub-property nodes “cuisine,” “price range,” “phone number,” and “location.” The reminder domain 364 includes the actionable intent node “set reminder,” and property nodes “subject” and “date/time.” In some embodiments, the ontology 360 is made up of many domains. Each domain may share one or more property nodes with one or more other domains. For example, the “date/time” property node may be associated with many different domains (*e.g.*, a scheduling domain, a travel reservation domain, a movie ticket

domain, etc.), in addition to the restaurant reservation domain 362 and the reminder domain 364.

[0088] While FIG. 3C illustrates two example domains within the ontology 360, other domains (or actionable intents) include, for example, “initiate a phone call,” “find directions,” “schedule a meeting,” “send a message,” and “provide an answer to a question,” and so on. A “send a message” domain is associated with a “send a message” actionable intent node, and optionally further includes property nodes such as “recipient(s),” “message type”, and “message body.” The property node “recipient” is optionally further defined, for example, by the sub-property nodes such as “recipient name” and “message address.”

[0089] In some embodiments, the ontology 360 includes all the domains (and hence actionable intents) that the digital assistant is capable of understanding and acting upon. In some embodiments, the ontology 360 is optionally modified, such as by adding or removing entire domains or nodes, or by modifying relationships between the nodes within the ontology 360.

[0090] In some embodiments, nodes associated with multiple related actionable intents are optionally clustered under a “super domain” in the ontology 360. For example, a “travel” super-domain optionally includes a cluster of property nodes and actionable intent nodes related to travels. The actionable intent nodes related to travels optionally includes “airline reservation,” “hotel reservation,” “car rental,” “get directions,” “find points of interest,” and so on. The actionable intent nodes under the same super domain (*e.g.*, the “travels” super domain) sometimes have many property nodes in common. For example, the actionable intent nodes for “airline reservation,” “hotel reservation,” “car rental,” “get directions,” “find points of interest” sometimes share one or more of the property nodes “start location,” “destination,” “departure date/time,” “arrival date/time,” and “party size.”

[0091] In some embodiments, each node in the ontology 360 is associated with a set of words and/or phrases that are relevant to the property or actionable intent represented by the node. The respective set of words and/or phrases associated with each node is the so-called “vocabulary” associated with the node. The respective set of words and/or phrases associated with each node can be stored in the vocabulary index 344 in association with the property or actionable intent represented by the node. For example, returning to FIG. 3B, the vocabulary associated with the node for the property of “restaurant” optionally includes words such as “food,” “drinks,” “cuisine,” “hungry,” “eat,” “pizza,” “fast food,” “meal,” and so on. For

another example, the vocabulary associated with the node for the actionable intent of “initiate a phone call” optionally includes words and phrases such as “call,” “phone,” “dial,” “ring,” “call this number,” “make a call to,” and so on. The vocabulary index 344, optionally, includes words and phrases in different languages.

[0092] The natural language processor 332 receives the token sequence (*e.g.*, a text string) from the speech-to-text processing module 330, and determines what nodes are implicated by the words in the token sequence. In some embodiments, if a word or phrase in the token sequence is found to be associated with one or more nodes in the ontology 360 (via the vocabulary index 344), the word or phrase will “trigger” or “activate” those nodes. Based on the quantity and/or relative importance of the activated nodes, the natural language processor 332 will select one of the actionable intents as the task that the user intended the digital assistant to perform. In some embodiments, the domain that has the most “triggered” nodes is selected. In some embodiments, the domain having the highest confidence value (*e.g.*, based on the relative importance of its various triggered nodes) is selected. In some embodiments, the domain is selected based on a combination of the number and the importance of the triggered nodes. In some embodiments, additional factors are considered in selecting the node as well, such as whether the digital assistant has previously correctly interpreted a similar request from a user.

[0093] In some embodiments, the digital assistant also stores names of specific entities in the vocabulary index 344, so that when one of these names is detected in the user request, the natural language processor 332 will be able to recognize that the name refers to a specific instance of a property or sub-property in the ontology. In some embodiments, the names of specific entities are names of businesses, restaurants, people, movies, and the like. In some embodiments, the digital assistant searches and identifies specific entity names from other data sources, such as the user’s address book, a movies database, a musicians database, and/or a restaurant database. In some embodiments, when the natural language processor 332 identifies that a word in the token sequence is a name of a specific entity (such as a name in the user’s address book), that word is given additional significance in selecting the actionable intent within the ontology for the user request.

[0094] For example, when the words “Mr. Santo” are recognized from the user request, and the last name “Santo” is found in the vocabulary index 344 as one of the contacts in the user’s contact list, then it is likely that the user request corresponds to a “send a message” or

“initiate a phone call” domain. For another example, when the words “ABC Café” are found in the user request, and the term “ABC Café” is found in the vocabulary index 344 as the name of a particular restaurant in the user’s city, then it is likely that the user request corresponds to a “restaurant reservation” domain.

[0095] User data 348 includes user-specific information, such as user-specific vocabulary, user preferences, user address, user’s default and secondary languages, user’s contact list, and other short-term or long-term information for each user. In some embodiments, the natural language processor 332 uses the user-specific information to supplement the information contained in the user input to further define the user intent. For example, for a user request “invite my friends to my birthday party,” the natural language processor 332 is able to access user data 348 to determine who the “friends” are and when and where the “birthday party” would be held, rather than requiring the user to provide such information explicitly in his/her request.

[0096] Other details of searching an ontology based on a token string is described in U.S. Utility Application Serial No. 12/341,743 for “Method and Apparatus for Searching Using An Active Ontology,” filed December 22, 2008, the entire disclosure of which is incorporated herein by reference.

[0097] In some embodiments, once the natural language processor 332 identifies an actionable intent (or domain) based on the user request, the natural language processor 332 generates a structured query to represent the identified actionable intent. In some embodiments, the structured query includes parameters for one or more nodes within the domain for the actionable intent, and at least some of the parameters are populated with the specific information and requirements specified in the user request. For example, the user may say “Make me a dinner reservation at a sushi place at seven o’clock.” In this case, the natural language processor 332 may be able to correctly identify the actionable intent to be “restaurant reservation” based on the user input. According to the ontology, a structured query for a “restaurant reservation” domain optionally includes parameters such as {Cuisine}, {Time}, {Date}, {Party Size}, and the like. In some embodiments, based on the information contained in the user’s utterance, the natural language processor 332 generates a partial structured query for the restaurant reservation domain, where the partial structured query includes the parameters {Cuisine= “Sushi”} and {Time = “7pm”}. However, in this example, the user’s utterance contains insufficient information to complete the structured

query associated with the domain. Therefore, other necessary parameters such as {Party Size} and {Date} are not specified in the structured query based on the information currently available. In some embodiments, the natural language processor 332 populates some parameters of the structured query with received context information. For example, in some embodiments, if the user requested a sushi restaurant “near me,” the natural language processor 332 populates a {location} parameter in the structured query with GPS coordinates from the user device 104.

[0098] In some embodiments, the natural language processor 332 passes the structured query (including any completed parameters) to the task flow processing module 336 (“task flow processor”). The task flow processor 336 is configured to receive the structured query from the natural language processor 332, complete the structured query, if necessary, and perform the actions required to “complete” the user’s ultimate request. In some embodiments, the various procedures necessary to complete these tasks are provided in task flow models 354. In some embodiments, the task flow models include procedures for obtaining additional information from the user, and task flows for performing actions associated with the actionable intent.

[0099] As described above, in order to complete a structured query, the task flow processor 336 sometimes needs to initiate additional dialogue with the user in order to obtain additional information, and/or disambiguate potentially ambiguous utterances. When such interactions are necessary, the task flow processor 336 invokes the dialogue processing module 334 (“dialogue processor 334”) to engage in a dialogue with the user. In some embodiments, the dialogue processor 334 determines how (and/or when) to ask the user for the additional information, and receives and processes the user responses. The questions are provided to and answers are received from the users through the I/O processing module 328. In some embodiments, the dialogue processor 334 presents dialogue output to the user via audio and/or visual output, and receives input from the user via spoken or physical (*e.g.*, clicking) responses. Continuing with the example above, when the task flow processor 336 invokes the dialogue flow processor 334 to determine the “party size” and “date” information for the structured query associated with the domain “restaurant reservation,” the dialogue flow processor 335 generates questions such as “For how many people?” and “On which day?” to pass to the user. Once answers are received from the user, the dialogue flow processor 334 can then populate the structured query with the missing information, or pass

the information to the task flow processor 336 to complete the missing information from the structured query.

[0100] In some cases, the task flow processor 336 receives a structured query that has one or more ambiguous properties, *e.g.*, when a structured query for the “send a message” domain indicates that the intended recipient is “Bob,” and the user happens to have multiple contacts named “Bob.” In such cases, the task flow processor 336, optionally, requests that the dialogue processor 334 disambiguate this property of the structured query. In turn, the dialogue processor 334, optionally, asks the user “Which Bob?”, and displays (or reads) a list of contacts named “Bob” from which the user may choose.

[0101] Once the task flow processor 336 has completed the structured query for an actionable intent, the task flow processor 336 proceeds to perform the ultimate task associated with the actionable intent. Accordingly, the task flow processor 336 executes the steps and instructions in the task flow model according to the specific parameters contained in the structured query. For example, the task flow model for the actionable intent of “restaurant reservation”, optionally, includes steps and instructions for contacting a restaurant and actually requesting a reservation for a particular party size at a particular time. For example, using a structured query such as: {restaurant reservation, restaurant = ABC Café, date = 3/12/2012, time = 7pm, party size = 5}, the task flow processor 336, optionally, performs the steps of (1) logging onto a server of the ABC Café or a restaurant reservation system such as OPENTABLE®, (2) entering the date, time, and party size information in a form on the website, (3) submitting the form, and (4) making a calendar entry for the reservation in the user’s calendar.

[0102] In some embodiments, the task flow processor 336 employs the assistance of a service processing module 338 (“service processor”) to complete a task requested in the user input or to provide an informational answer requested in the user input. For example, the service processor 338 can act on behalf of the task flow processor 336 to make a phone call, set a calendar entry, invoke a map search, invoke or interact with other user applications installed on the user device, and invoke or interact with third party services (*e.g.* a restaurant reservation portal, a social networking website, a banking portal, etc.). In some embodiments, the protocols and application programming interfaces (API) required by each service can be specified by a respective service model among the services models 356. The service processor 338 accesses the appropriate service model for a service and generates

requests for the service in accordance with the protocols and APIs required by the service according to the service model.

[0103] For example, if a restaurant has enabled an online reservation service, the restaurant can submit a service model specifying the necessary parameters for making a reservation and the APIs for communicating the values of the necessary parameter to the online reservation service. When requested by the task flow processor 336, the service processor 338 can establish a network connection with the online reservation service using the web address stored in the service model, and send the necessary parameters of the reservation (*e.g.*, time, date, party size) to the online reservation interface in a format according to the API of the online reservation service.

[0104] In some embodiments, the natural language processor 332, dialogue processor 334, and task flow processor 336 are used collectively and iteratively to infer and define the user's intent, obtain information to further clarify and refine the user intent, and finally generate a response (*i.e.*, an output to the user, or the completion of a task) to fulfill the user's intent.

[0105] In some embodiments, after all of the tasks needed to fulfill the user's request have been performed, the digital assistant 326 formulates a confirmation response, and sends the response back to the user through the I/O processing module 328. If the user request seeks an informational answer, the confirmation response presents the requested information to the user. In some embodiments, the digital assistant also requests the user to indicate whether the user is satisfied with the response produced by the digital assistant 326.

[0106] More details on the digital assistant can be found in the U.S. Utility Application No. 12/987,982, entitled "Intelligent Automated Assistant," filed January 18, 2010, U.S. Utility Application No. 61/493,201, entitled "Generating and Processing Data Items That Represent Tasks to Perform," filed June 3, 2011, the entire disclosures of which are incorporated herein by reference.

[0107] In some embodiments, one of the task flow models 354 implemented by a digital assistant includes a constrained selection task (*e.g.*, "{task = Constrained Selection}, {selection domain = "restaurant"}, {cuisine = "sushi"}). A constrained selection task is a generic task for selecting one or more selection items (*e.g.*, sushi restaurants) from a set of selection items (*e.g.*, restaurants) in a selection domain (*e.g.*, the restaurant domain) based on a set of constraints (*e.g.*, cuisine = sushi).

[0108] In some embodiments, the digital assistant, optionally, initiates a constrained selection task for the domain of restaurants when a user has provided one or more restaurant search criteria or constraints (*e.g.*, type of cuisine, location, time, price range, atmosphere, etc.) to the digital assistant. For example, in response to a speech input (*e.g.*, “I want to have some Asian food for lunch today.”) from a user, the digital assistant, optionally, generates a constrained selection task for the restaurant domain using constraint parameters or search criteria (*e.g.*, {cuisine = “Asian”}, {time = “noon today”}) extracted from the user’s speech input. In some embodiments, the digital assistant, optionally, augments the constrained selection task, *e.g.*, using additional constraints generated based on other knowledge the digital assistant has acquired about the user. The additional constraints include, for example, the user’s current location, or a list of the user’s favorite restaurants. Once the digital assistant has identified a set of selection items (*e.g.*, Asian restaurants) satisfying the search criteria, the digital assistant, optionally, presents these selection items to the user for selection and/or other follow-on actions (*e.g.*, making a reservation at a selected restaurant).

[0109] In some embodiments, the digital assistant implements a data model for the constrained selection task. The data model, optionally, includes selection items, item features, and selection criteria and constraints, as parameters of the constrained selection task.

- “Selection items” are instances of a selection class (*e.g.*, “restaurants”).
- “Item features” are common properties and attributes associated with a selection item. For example, the name and phone number of a restaurant are item features of a restaurant selection item. In some embodiments, item features include intrinsic features (*e.g.*, the name or cuisine of a restaurant) and/or relational features (*e.g.*, the distance from a location of interest). In some embodiments, item features also include static features (*e.g.*, restaurant name) and/or dynamic features (*e.g.*, rating).
- “Selection criteria” are item features that are used to compare the value or relevance between selection items to a user. That is, selection criteria are ways to evaluate which items are preferred by the user. For example, *proximity* (defined as distance from the location of interest) is a selection criterion.
- Selection criteria sometimes have an inherent “preference order.” For example, the proximity criterion has an inherent preference order indicating that a closer distance is more preferable to the user. In another example, a *rating*

criterion has an inherent preference order indicating that higher rating is more preferable to the user. In yet another example, an *ambiance* criterion has an inherent preference order indicating that a more lively and active ambiance is more preferable than a more subdued and quiet ambiance. In some embodiments, the default or inherent preference order is set by the user, or automatically selected for the user based on the user's demographic characteristics. For example, the above-mentioned preference order for the *ambiance* criterion is optionally set for a young person, and a reversed preference order is optionally set for an older person.

- “Constraints” are restrictions on the desired values of the selection criteria. Formally, constraints are optionally represented as set membership (*e.g.*, desired cuisine type includes “Italian,” operating hours include “Saturdays,” facilities includes “pharmacies,” etc.), pattern matches (*e.g.*, restaurant review text includes “romantic”), fuzzy inequalities (*e.g.*, distance less than a few miles), qualitative thresholds (*e.g.*, highly rated), and/or more complex functions (*e.g.*, a value function measuring a good value for money), etc.

[0110] To summarize, in some embodiments, a selection domain is modeled as selection classes with item features that are important to users. Some of the item features are used to select and order items offered to the user - these features are called selection criteria. Constraints are limits imposed on the selection criteria that narrow the set of items that satisfy the selection criteria. In this specification, selection criteria and constraints for a constrained selection task are collectively referred to as “search criteria” for the constrained selection task. A particular constrained selection task is also referred to as a search request.

[0111] In general, constrained selection is useful to a user when the user knows the category of his or her interest and wishes to select one or more instances of the category with some desired properties. Frequently, a user initiates a constrained selection task by providing a natural language request (*e.g.*, a natural language speech input). In the request, the user optionally describes a desire outcome, and the digital assistant operationalizes the desired outcome into one or more search criteria, and uses the search criteria to identify a relevant selection domain and one or more selection constraints. For example, instead of saying “find one or more restaurants less than 2 miles from the center of Palo Alto whose cuisine includes Italian food,” the user may simply say “I want some Italian food.” The digital assistant generates the formal parameters of the search request (*e.g.*, {selection domain= “restaurant”},

{cuisine type = “Italian”}, {location= “current location”= “Palo Alto”}, {distance= “nearby” = “less than 2 miles from current location”}) based on the user’s direct speech input and context information (*e.g.*, the user’s current location).

[0112] Sometimes, the user provides a request specifying one or more explicit search criteria, and the assistant optionally adds one or more implicit search criteria (*e.g.*, the user’s preferred price range, the user’s current location, etc.) to the search request. In some embodiments, the digital assistant optionally suggests several useful selection criteria to the user, and the user can select the ones that are important to him/her at the moment. Sometimes, the digital assistant provides some guidance to the user with respect to the selection criteria to focus the search to reduce the number of choices ultimately presented to the user.

[0113] One type of useful constrained selection task is associated searching for a category of “places” where a user can visit and carry out certain types of activity commonly associated with that category of places. For example, by performing a constrained selection task, a digital assistant can help a user to find places to eat (*e.g.*, restaurants), places to stay (*e.g.*, hotels), places to rent a car (*e.g.*, car rental locations), places to send a letter (*e.g.*, post offices or postal drop-off locations), places to buy clothes (*e.g.*, shopping malls), places to exercise (*e.g.*, gyms, parks), places to buy power tools (*e.g.*, hardware stores, home improvement stores), places to read (*e.g.*, book stores, libraries, coffee shops), places to see sports games (*e.g.*, stadiums), and places to buy groceries (*e.g.*, grocery stores), etc.

[0114] Another type of useful constrained selection task is associated with searching for a category of “events or activities” that a user can join or participate at particular times and locations. For example, by performing a constrained selection task, a digital assistant can help a user find the particular locations and times that some events and activities of interest (*e.g.*, movies, concerts, street fairs, farmer’s markets, sporting events, etc.) will take place, such that the user can visit those particular locations at the particular times to watch, participate, and/or join the events.

[0115] Both types of constrained selection tasks mentioned above involve a respective selection domain (*e.g.*, restaurant domain, movie domain) in which each selection item (*e.g.*, each restaurant entity or movie entity) is associated with a respective physical location. In addition, for each selection item, a useful purpose of the selection item is accomplished by the user being present at that physical location (*e.g.*, eating at the restaurant). Since the

search results for these two types of searches are tied to physical locations, these two types of search requests are also sometimes referred to as “local search requests.”

[0116] For the search requests involving selection items whose useful purposes for a user are accomplished by the user being present at the physical locations associated with the selection items, relevance of a particular selection item sometimes changes depending on a variety of non-static factors. For example, whether a particular Italian restaurant in Palo Alto is of interest to the user sometimes changes depending on factors such as the current location of the user, the kinds of transportation that the user currently has access to, the time that the user plans to eat, whether the user has other types of engagements before or after the time that the user plans to eat, the people that the user intends to dine with, and/or the amount of money the user intends to spend on a meal, etc. In another example, whether a particular action movie showing at a particular theatre at a certain show time is of interest to the user sometimes changes depending on factors such as who will be accompanying the user for the movie, whether the user has other engagement at, before, or after that show time, whether the user has enough time to reach the theatre from his engagement prior to the movie show time, and/or whether the user can conveniently accomplish some other errands near the particular show time and show location, etc.

[0117] As described in this specification, in some embodiments, a digital assistant evaluates schedule items (*e.g.*, calendar entries, reminder items, and to-do items) in an electronic schedule associated with a user to identify particular schedule items that are potentially relevant to a local search request received from the user. The digital assistant optionally uses additional information contained in one or more of these potentially relevant schedule items to augment the search request and/or to filter and/or re-rank search results obtained for the search request.

[0118] FIGS. 4A-4B illustrate an exemplary process 400 for identifying one or more relevant schedule items from an electronic schedule associated with the user, and using additional information contained in the one or more relevant schedule items to augment a search request received from the user and/or to filter the search results of the search request. In some embodiments, the process 400 is performed by the search augmentation module 340 of the digital assistant 326 based on user schedule items 358, *e.g.*, shown in FIGS. 3A and 3B.

[0119] In the process 400, the digital assistant receives (402) a search request from a user, where the search request specifies one or more search criteria for selecting one or more selection items from a respective selection domain. In some embodiments, the search request is a speech input received from the user. In some embodiments, the search request is a user input received through other input modalities (*e.g.*, a text input, a selection from a drop-down menu, etc.). In some embodiments, the search request is a structured query that has been generated based on direct user input (*e.g.*, a speech input) to the digital assistant. In some embodiments, the search request only specifies a respective selection domain, and does not include any other search criteria (*e.g.*, when the direct user input is “Find me a post office.”). In some embodiments, the search request specifies a respective selection domain, and at least one search criterion (*e.g.*, when the direct user input is “Find a post office near my house.”). In some embodiments, the one or more search criteria include both a selection criterion and at least one constraint (*e.g.*, when the direct user input is “Find a post office near my house that opens on Saturdays.”).

[0120] In some embodiments, the digital assistant interprets the user input (*e.g.*, a speech input) to determine the user’s intent, and generates an associated task to accomplish the user’s intent. In some embodiments, the natural language processing capabilities allows the digital assistant to correctly infer the user’s intent based on a variety of expressions in natural language. In some embodiments, the digital assistant determines at least a selection domain from the speech input from the user.

[0121] In some embodiments, the digital assistant processes the input from a user to generate a local search request. For example, when the user provides a speech input asking about a place to eat (*e.g.*, “Find me a place to eat in Palo Alto,” “Is there a restaurant here?” or “I want Italian food.”), the digital assistant infers that the user wishes to perform a restaurant search, and generates a constrained selection task in the restaurant domain. In another example, when the user provides a text input asking about a movie showing events (*e.g.*, “What movies are showing this weekend?” “What’s on at the Fillmore theatre?” “Show me some new movies?”) in a dialogue interface of the digital assistant, the digital assistant infers that the user wishes to perform a movie search, and generates a constrained selection task in the movie domain. In some embodiments, depending on the direct input from the user, the digital assistant selectively generate constrained selection tasks in domains such as grocery stores, hardware stores, post offices, movie theatres, parks, Department of Motor Vehicles (DMVs), hotels, car-rental places, hospitals, police stations, thrift stores,

pharmacies, stadiums, street fairs, public lectures, concerts, farmer's markets, sporting games, etc.

[0122] After receiving the search request, the digital assistant identifies (404) one or more schedule items relevant to the search request, *e.g.*, based on information (*e.g.*, relevant selection domain and search criteria) contained in the search request. In some embodiments, the digital assistant identifies the relevant schedule items from an electronic schedule associated with the user. In some embodiments, an electronic schedule associated with the user includes an electronic calendar associated with the user, and the schedule items in the electronic schedule include calendar items (*e.g.*, calendar entries) stored in the electronic calendar. In some embodiments, the electronic schedule associated with the user includes a reminder list associated with the user, and the schedule items in the electronic schedule include reminder items stored in the reminder list. In some embodiments, the electronic schedule associated with the user includes a to-do list associated with the user, and the schedule items in the electronic schedule include task items or to-do items stored in the to-do list. In some embodiments, the electronic schedule associated with the user includes the user's own calendar(s), reminder list(s), and/or to-do list(s). In some embodiments, the electronic schedule associated with the user further includes the calendar(s), reminder list(s), and to-do list(s) that have been shared with the user by another user (*e.g.*, the user's spouse, supervisor, children, etc.). For a schedule item in a shared calendar to be potentially relevant to a search request, the schedule item has to list the user or another person mentioned in the search request as a participant.

[0123] In some embodiments, each schedule item is associated with at least one of a time, a location, a subject matter, and one or more participants. The location, time, subject matter, and participants associated with a schedule item are optionally determined based on various predetermined data fields of the schedule item, and/or from the freeform notes that the user has entered in the schedule item. For example, a calendar entry for "pick up mom from the airport" is associated with a time (*e.g.*, {event time="Feb. 3, 2013 at 5:45pm"}), a location (*e.g.*, {event location="San Francisco Airport"}), a subject matter (*e.g.*, {subject="Airport pick-up"}), and one or more participants (*e.g.*, {participants="the user + the user's mom"}). In another example, a to-do item for "buy groceries" is associated with a time (*e.g.*, {event time="tonight"}), and a subject matter (*e.g.*, {subject="purchase groceries"}), but no specified location or participants. In another example, a reminder item for "Submit job application" is associated with a time (*e.g.*, {deadline="Jan 1, 2013"}), a subject matter

(*e.g.*, {subject= “submit job application”}), a default participant (*e.g.*, {default participant= “the user”}), but no specified location. In another example, a task item “buy lottery ticket” is associated with a subject matter (*e.g.*, {subject= “purchase lottery ticket”}) and a default participant (*e.g.*, {default participant= “the user”}), but no specified location or time. Presumably, the user is familiar with the time that the lottery is announced each week and the location he/she usually purchases the ticket, and does not need to include that information in the task item.

[0124] In some embodiments, the digital assistant selects (406) an initial set of potentially relevant schedule items based on a first set of criteria (*e.g.*, a looser relevance filter, or a relevance filter based on a first set of search criteria), and selects the one or more relevant schedule items from the initial set of potentially relevant schedule items based on a second set of criteria (*e.g.*, a stricter relevance filter, or a relevance filter based on a second set of search criteria). In some embodiments, more than one preliminary step is performed to select an initial set of potentially relevant schedule items. In some embodiments, when identifying the one or more relevant schedule items, the digital assistant evaluates each known schedule items associated with the user. In some embodiments, the digital assistant only evaluates the schedule items associated with a predetermined threshold time period (*e.g.*, a week or a month) from the present time. By implementing a threshold time period, schedule items that are too remote in time from the present search request are not considered, even if those schedule items are concerned with the same location, subject matter, and/or people specified in the search request. By implementing multiple sets of criteria and selects the set of potentially relevant schedule items in multiple stages (*e.g.*, alternating stages of inclusion and exclusions), the risk of being over-inclusive or under-inclusive in the initial set of potentially relevant schedule items is reduced. Additional details for identifying the initial set of potentially relevant schedule items from which the relevant schedule items are later identified are provided later with respect to FIGS. 6A-6C.

[0125] In some embodiments, when choosing the initial set of potentially relevant schedule items, the digital assistant excludes (408) schedule items that are associated with time and/or locations that are unlikely to be relevant to the search request. For example, in some embodiments, if the search request explicitly specifies a date or day, schedule items associated with a different date or day are unlikely to be relevant to the search request, and the digital assistant optionally excludes these schedule items from the initial set of potentially relevant schedule items. In one specific example, if the search request is “Find a restaurant

for dinner today,” schedule items associated with days other than “today” are excluded from the initial set of potentially relevant schedule items. In another specific example, if the search request is “Find a restaurant for my date this Thursday,” schedule items associated with days other than “this Thursday” are excluded from the initial set of potentially relevant schedule items.

[0126] In another example, in some embodiments, if the search request explicitly specifies a geographic location, schedule items associated with another geographic location too remote (*e.g.*, involving long distance travel or outside same-day driving distance) from the specified geographic location are unlikely to be relevant to the search request, and the digital assistant optionally excludes these schedule items from the initial set of potentially relevant schedule items. In one specific example, if the search request is “Find restaurant in Palo Alto,” schedule items associated with a remote location “Washington DC” (which is too far away from Palo Alto, California) are excluded from the initial set of potentially relevant schedule items. More details on how an initial set of potentially relevant schedule items are selected are provided in FIGS. 6A-6C and accompanying descriptions.

[0127] In some embodiments, to identify the one or more relevant schedule items, the digital assistant identifies (410) one or more schedule items that are relevant to the search request in at least one of location, time, associated people, and subject matter. When a schedule item is relevant to the search request in at least one of location, time, associated people, and subject matter, it is likely that the schedule item is relevant to the search request, and the additional information contained in the schedule item can be used to augment the search request.

[0128] Referring now to FIG. 4B, in some embodiments, to determine whether a particular schedule item is relevant to the search request in time, the digital assistant identifies (412) a first time window associated with the search request and a second time window associated with the particular schedule item, and determines whether there is an overlap between the first time window and the second time window.

[0129] In some embodiments, the time specified in a schedule item is a single time range (*e.g.*, “3:00-4:00pm on Jan 1, 2013”) or multiple discrete time ranges (*e.g.*, “5:00- 6:00pm every Monday”). In some embodiments, the time window associated with a schedule item is the same as the time range(s) specified in the schedule item. In some embodiments, the time window associated with a schedule item includes a buffer time period (*e.g.*, a reminder time,

preparation time, and/or travel time) in addition to a specified event time. In some embodiments, the time specified in a schedule item (*e.g.*, a to-do item) is a deadline for completing a task, and the time window associated with the schedule item is a time range extending from the present time to the deadline. In some embodiments, the time window associated with the schedule item is the actual amount of time needed to complete the task, and the digital assistant dynamically selects the start and end times of the time window based on the event time(s) and duration(s) of other schedule items on the user's electronic schedule. In some embodiments, if no event time is specified in a schedule item (*e.g.*, the task item "buy lottery ticket" mentioned above), the digital assistant optionally attaches time information to the schedule item based on an Internet search (*e.g.*, "lottery time"). In some embodiments, the digital assistant sometimes determines that no relevant time window can be identified from a particular schedule item directly, *e.g.*, when no schedule time is specified in the particular schedule item.

[0130] In some embodiments, a time window associated with a search request is a time range specified in the user's direct input. For example, for a speech input "Find a romantic restaurant for my date on Saturday evening."), the digital assistant identifies a time window "Saturday evening." In some embodiments, the digital assistant optionally refines the explicit time range with additional information available to the digital assistant. For example, the digital assistant optionally uses a time range "6pm-8pm" to further refine the term "evening." In another example, the digital assistant optionally uses a time range "12:00pm-2:00pm" to further refine the term "after lunch" in a search request "What's showing after lunch?"

[0131] In some embodiments, the user does not explicitly specify a time in the search request, the digital assistant optionally identifies a relevant time window based on the present time and/or the search criteria (*e.g.*, subject matter) specified in the search request. For example, for a speech input "Find me a place for lunch," the digital assistant identifies a relevant time window ("11:30am-2pm today" or "an one-hour window between 11:30-2pm today") based on the subject matter (*e.g.*, "lunch").

[0132] In some embodiments, the digital assistant determines that no relevant time window can be identified from a particular search request directly, *e.g.*, when no time criterion is specified in the particular search request. In some embodiments, when the search request does not include a time criterion, the digital assistant assumes that the relevant time

window for a search request is a “present” time window of a predetermined duration (*e.g.*, the present day, the present week, or the present month, the next two hours, etc.).

[0133] In some embodiments, the digital assistant allows a certain degree of “fuzziness” around the time windows determined from the search request and/or the schedule item to avoid under-inclusion of potentially relevant schedule items. For example, if the respective time windows of a search request and a schedule item are spaced apart by a short amount of time (*e.g.*, an amount of time within the “fuzziness” threshold), the digital assistant optionally includes the schedule item as a relevant schedule item for the search request even though the two time windows do not actually overlap. In some embodiments, the “fuzziness” is built into the way the relevant time windows are determined. For example, in some embodiments, the digital assistant automatically adds a five-minute buffer or a thirty-minute buffer to a specified time range in the search request and/or a schedule item, when calculating the relevant time window for the search request and/or schedule item.

[0134] In some embodiments, to determine whether a particular schedule item is relevant to the search request in location, the digital assistant identifies (414) a first geographic area associated with the search request and a second geographic area associated with the particular schedule item, and determines whether there is an overlap between the first geographic area and the second geographic area.

[0135] In some embodiments, the location specified in a schedule item is a street address (*e.g.*, “123 Main Street, Palo Alto, CA”) or a distinct geographic location (*e.g.*, “south gate of Central Park,” “office lobby,” etc.). In some embodiments, the geographic area associated with a schedule item is a geographic region of a predetermined size and includes the location specified in the schedule item. In one example, the geographic area is optionally the city including the location specified in the schedule item. In another example, the geographic area is optionally an area within a certain distance (*e.g.*, 500 feet, 5 miles, 10 miles, 50 miles, etc.) from the location specified in the schedule item. In some embodiments, the digital assistant dynamically determines the size of the geographic area based on the transportation means (*e.g.*, car vs. foot) available to the user.

[0136] In some embodiments, the schedule item does not explicitly specify a location, and the digital assistant optionally attaches location information to the schedule item based on the subject matter specified in the schedule item. For example, if the schedule item is a to-do item “mail wedding invitations,” the digital assistant optionally attaches locations of one

or more nearby post offices to the schedule item. In some embodiments, the digital assistant sometimes determines that no relevant geographic area can be identified from a particular schedule item directly, *e.g.*, when no location is specified in the particular schedule item.

[0137] In some embodiments, a relevant geographic area associated with a search request is a geographic area specified in the user's direct input. For example, for a speech input "Find me a romantic restaurant in Palo Alto."), the digital assistant identifies a geographic area "Palo Alto" from the speech input. In some embodiments, the digital assistant optionally broadens the relevant geographic area to include neighboring cities of the location explicitly specified in the search request, *e.g.*, when the specified city and its neighboring cities are within short driving distances of one another.

[0138] In some embodiments, the relevant geographic area of a search request includes an area surrounding the user's current location. For example, if the user's search request is "What grocery stores are open now?" or "What's nearby?" the digital assistant optionally identifies a relevant geographic area for the search request based on the user's current location and a short radius (*e.g.*, less than 15 minute's walk or drive) from the user's current location.

[0139] In some embodiments, the digital assistant sometimes determines that no relevant geographic area can be identified from a particular search request directly, *e.g.*, when no location criterion is specified in the particular search request. In some embodiments, when the search request does not include a location criterion, the digital assistant assumes that the relevant geographic area for a search request is a "local" geographic area of certain size (*e.g.*, the city in which the user is currently located, an area within driving distance from the user's current location, an area within driving distance from the user's home/work location, etc.).

[0140] In some embodiments, to determine whether a particular schedule item is relevant to the search request in associated people, the digital assistant identifies (416) a first group of people including all people (if any) specified in the search request, and a second group of people including all people specified as participants or otherwise mentioned in the particular schedule item. The digital assistant then determines whether there is any overlap between the first group of people and the second group of people. For example, if the search request is "Find lodging options for my bridal party," a schedule item for "dinner with my bridesmaids" are relevant in associated people, since there is at least some overlap between the members of the user's bridal party and her bridesmaids. In some embodiments, the digital assistant does

not consider inclusion of the user him/herself in the first and second groups when determining whether the first group and the second group overlap in associated people.

[0141] In some embodiments, to determine whether a particular schedule item is relevant to the search request in subject matter, the digital assistant identifies (418) a first domain associated with the search request and a second domain associated with the particular schedule item, and determines whether the first domain and the second domain are related domains (*e.g.*, parent-child domains, or sibling domains) in an ontology of domains implemented by the digital assistant. For example, the “hotel” domain is a child domain of the “travel” domain. In another example, the “restaurant” domain is a sibling domain of the “movies” domain under the “entertainment” domain. Therefore, in some embodiments, a schedule item associated with traveling to a remote location (*e.g.*, a reminder item for taking a flight on Sunday) is relevant to a search request for the hotel domain (*e.g.*, a search for “Find hotels in Atlanta.”) in subject matter. In some embodiments, a schedule item associated the restaurant domain (*e.g.*, a calendar item for “Dinner with blind date”) is relevant to a search request for the movie domain (*e.g.*, a search for “What movies are showing this Saturday?”) in subject matter.

[0142] In some embodiments, the digital assistant explicitly defines particular subject matters that are considered relevant to each other. Other examples of pairs of schedule item and search request that are relevant to each other include: (1) search request = “Find me a restaurant”, and schedule item = “Lunch with boss”, “Order take-out for lunch meeting,” or “Date with Linda,” etc.; (2) search request = “Find gift shop”, and schedule item = “Birthday party,” “Anniversary dinner with wife” “Attend Tim’s graduation,” “Christmas shopping,” “Visit morn in hospital,” etc.; (3) search request = “Find me a pharmacy,” and schedule item = “Doctor’s Appointment”; (4) search request = “Find gas stations,” and schedule item = “Ski trip to Tahoe,” “Doctor’s appointment,” “Visit the Zoo,” etc; (5) search request = “What movies are on this Saturday?” and schedule item = “Date with Kate,” “Dinner with wife,” “Hang out with nephews and nieces,” etc; and (6) search request = “Find me a perfume shop”, and schedule item = “Buy gift for wife’s birthday”; etc.

[0143] In some embodiments, to determine whether a particular schedule item is relevant to the search request, the digital assistant combines multiple factors to assess whether the particular schedule item is relevant to the search request. For example, in some embodiments, although a schedule item “Dinner with mom in Palo Alto on Mother’s Day”

may not be considered to be relevant to a search request “Find me a flower shop in Palo Alto,” it would be considered relevant to another search request “Where can I buy flowers for mom in Palo Alto?”

[0144] In some embodiments, referring back to FIG. 4A, after the digital assistant has identified one or more relevant schedule items for the search request, the digital assistant augments (420) the search request based on information contained in at least one of the identified schedule items. In some embodiments, the digital assistant presents (422) the augmented search request to the user. In some embodiments, the digital assistant requests user confirmation before proceeding with the search using the augmented search request. In some embodiments, the digital assistant filters (424) the search results based on augmented search request.

[0145] In some embodiments, the digital assistant presents (426) the search results to the user. In some embodiments, only the filtered search results are presented to the user. In some embodiments, the digital assistant optionally presents the identified relevant schedule items to the user, and confirms with the user whether these identified schedule items are indeed relevant to the search request.

[0146] Some exemplary use cases for using relevant schedule items identified for a search request to augment the search request and/or to filter search results are described below. These exemplary uses cases illustrate how information extracted from one or more relevant schedule items individually or in combination are used to augment the search request in accordance with various embodiments. In some embodiments, the order by which the schedule items occur, and the relationships (*e.g.*, synergies, conflicts, and dependencies) between various schedule events and are also used to augment the search request.

[0147] As shown in FIG. 5A, the user provides a speech input 502 (*e.g.*, “Find me a place for dinner tonight.”) to a user device (*e.g.*, a user device 104) on which a digital assistant is operating. The digital assistant captures the speech input 502 and optionally presents corresponding text of the speech input in a dialogue panel 504 displayed on the user device. The digital assistant processes the speech input 502 and generates a search request 506 based on the speech input 502. The digital assistant initiates a constrained selection task based on the search request 506. The search request 506 specifies a selection domain {selection domain=restaurant}, and an implicit time criterion = {time=dinner tonight}, but not any explicit location constraint or participant constraint. The digital assistant optionally

determines that the implicit time criterion for the search request 506 is “6pm-8pm.” The digital assistant further determines that the relevant time window for the search request 506 is the present day (*i.e.*, today). The digital assistant optionally determines that, given that no location is explicitly specified in the speech input, a relevant geographic area for the search request 506 is a local area within driving distance from the user’s current location (*i.e.*, no extensive travel or long-time driving required). The digital assistant optionally determines that the associated people for the search request include at least the user.

[0148] After the digital assistant has determined the relevant geographic area, time window, subject matter, and/or associated people of the received search request 506, the digital assistant evaluates each of an initial set of potentially relevant schedule items 508 associated with the user to determine which schedule items are particularly relevant to the search request 506. For illustration purposes, the set of potentially relevant schedule items 508 include four schedule items from the user’s schedule of the current week, as shown in the table below:

Item Type		Time	Location	Subject	Participants
(1)	Calendar	Event time = 6:00-8:00pm tonight	ABC theatre in Palo Alto, CA	See movie with mom	Organizer = self; Attendee = self + mom
(2)	Reminder	Event time= 5:30pm today; Reminder time =15 minutes before event time	101, Mary Ln., Cupertino, CA	Pick up mom	self + mom
(3)	Calendar	Event time = 2:00pm tomorrow	XYZ company, Mountain View, CA	Job interview at XYZ Co.	Organizer = Mr. Benton; Attendee = Mr. Benton + self
(4)	To-do	n/a	n/a	Buy grocery	self

[0149] The digital assistant identifies calendar item (1) and the reminder item (2) as relevant to the search request 506 because they each overlap with the search request 506 in time. For example, suppose that the relevant time window of the search request is determined to be the present day. Both calendar item (1) and the reminder item (2) are associated with time windows within the present day, and therefore are relevant to the search

request 506. In some embodiments, if a specific time window (*e.g.*, “dinner” + “today” → “6:00pm-8:00pm today”) is used for the search request 506, the calendar item (1) “See movie with mom” is identified as a relevant schedule item because the calendar item (1) is associated with a time window (*e.g.*, “6:00pm-8:00pm today”) overlapping with the relevant time window of the search request (*e.g.*, “6:00pm-8:00pm today”). In addition, the reminder item (2) “Pick up mom” is associated with a time range (*e.g.*, “5:15pm-5:30pm today”). The digital assistant optionally identifies the reminder item (2) as a relevant schedule item based on a thirty-minute fuzziness threshold attached to the respective time windows associated with the search request and the reminder item (2).

[0150] In some embodiments, the digital assistant includes all schedule items associated with the present day (*e.g.*, using a time window of “one day” for all schedule items) as relevant to the search request. In some embodiments, the digital assistant assigns more relevance to schedule items that have a greater extent of overlap in time with the search request. For example, in this particular scenario, the digital assistant optionally treats the calendar item (1) as a more relevant schedule item to the search request 506 than the reminder item (2). In some embodiments, the digital assistant includes all schedule items associated with the present day as relevant to all search request that does not explicitly specifies a different day (*e.g.*, “next Saturday”) in its search criteria. In some embodiments, the more relevant schedule item is given more weight when considering what and how information in the relevant schedule items are used to augment the search request.

[0151] In this particular example, the digital assistant also determines that the relevant geographic areas associated with the schedule items (1)-(3) all overlap with the relevant geographic area for the search request (*e.g.*, a local area within driving distance from the user’s current location). However, since the user explicitly stated a time criterion of “today”, and a time window associated with the schedule item (3) is for a different day (*e.g.*, “tomorrow”), the digital assistant does not consider the schedule item (3) as a relevant schedule item for the search request 506. In addition, the to-do item (4) does not overlap with the search request 506 in any of location, time, subject matter, and people (excluding the user him/herself), and is therefore not considered to be a relevant schedule item for the search request 506.

[0152] In some embodiments, the digital assistant optionally presents the relevant schedule items identified for the search request to the user to facilitate the user’s subsequent

review and selection of available selection items. As shown in FIG. 5A, the digital assistant optionally provides a response 510 (*e.g.*, text or speech output) to the user's speech input 502, stating that, "You have a number of events around dinner time tonight that may affect your choice of restaurants: (1) Movie with mom at 6pm; and (2) Picking up mom at 5:30pm."

[0153] In some embodiments, after presenting the relevant schedule items to the user, the digital assistant optionally proceeds with the search using the original search criteria specified in the search request. In some embodiments, as illustrated below, the digital assistant optionally augments the search request using information extracted from the relevant schedule items, makes suggestions of refinement to the original search criteria specified in the search request, adds new search criteria to the search request, automatically categorizes the search results based on information extracted from the relevant schedule items, makes recommendations based on information extracted from the relevant schedule items, and/or provide other information and assistance to help the user resolve potential scheduling conflicts and/or improve efficiency in the user's subsequent action plan.

[0154] In some embodiments, the digital assistant reviews the calendar item (1), and recognizes that time window specified in the search request and the time window of the calendar item (1) substantially coincide with each other. In response to this discovery, the digital assistant determines whether the schedule item poses a conflict with the inferred purpose of the search request (*e.g.*, "to have dinner"). For example, the digital assistant may determine that the subject of the search request 506 and the subject of the schedule item (*e.g.*, calendar item (1)) are things to be done at different locations (*e.g.*, a restaurant and a movie theater, respectively), and there isn't enough time to do both things within their respective time windows. In such a scenario, the digital assistant optionally presents this information to the user, and asks the user to pick a different time window for the search request, or confirm the original time window specified in the search request. In this example, based on the calendar item (1), the digital assistant presents a clarification request 512 to the user based information extracted from the calendar item (1). In some embodiments, the digital assistant proposes a modification to the search request to the user. As shown in FIG. 5B, the digital assistant asks the user, "You have a movie from 6:00pm-8:00pm tonight. Would you like to have dinner before or after the movie?" The proposed time window(s) avoid the conflict with the time window of the relevant calendar item (1).

[0155] In some embodiments, if the user replies “After the movie,” the digital assistant optionally uses the end time of the calendar item (1) to generate a refinement of the time criterion (*e.g.*, “after 8pm tonight”) for the search request 506. In some embodiments, the digital assistant optionally presents an output 514 showing the refinement to the search request (*e.g.*, a speech output saying, “Searching for restaurants with open tables after 8:00pm today...”), as shown in FIG. 5C. In some embodiments, the digital assistant then presents the search results satisfying the refined search request to the user. In some embodiments, the digital assistant optionally adds a location criterion to the search request 506 based on the location (*e.g.*, “ABC theatre in Palo Alto”) specified in the calendar item (1). Since the user has clarified that the dinner is for after the movie, the digital assistant optionally adds to the search request a location criterion {restaurant location = “near Palo Alto”} with a default preference order indicating that a closer distance from the movie theatre is preferred. In some embodiments, the digital assistant presents the search results to the user in a subsequent output 516. In some embodiments, as shown in FIG. 5C, the digital assistant only presents a small number of search results that are ranked highest (*e.g.*, the top two results) based on the augmented search request, *i.e.*, the search request including the refined and/or additional search criteria generated based on the relevant schedule items.

[0156] In some embodiments, upon a discovery that the relevant time windows of the calendar item (1) and the search request 506 substantially coincide with each other, the digital assistant also determines whether the purpose of the calendar item (1) aligns with the inferred purpose of the search request 506 in one or more aspects. For example, the digital assistant may infer that the subject of the search request (*e.g.*, “to have dinner”) and the subject of the calendar item (1) (*e.g.*, “to see movie”) are things likely to be done at proximate time(s) and location(s), and/or with the same people. In such a scenario, the digital assistant optionally presents this information to the user, and asks the user to confirm such an inference made by the digital assistant. In some embodiments, once the inference is confirmed by the user, the digital assistant optionally uses additional information extracted from the schedule item to generate additional search criteria for the search request. As shown in FIG. 5D, the digital assistant infers from the calendar item (1) (“See movie with mom”) that the user may wish to have the dinner with mom as well, and adds the additional people criterion to the search request 506. For example, the digital assistant optionally presents an output 518 asking the user, “You have a movie from 6:00pm-8:00pm tonight with your mom, would you like to have dinner with your mom?”

[0157] In some embodiments, the digital assistant provides the response 518 (“Will you be having dinner with your mom?”) after the user has confirmed that the dinner is to occur after the movie (*e.g.*, after the digital assistant had provided the output 512). In some embodiments, the digital assistant provides a response that combines the inquiry in both the responses 512 and 518. For example, digital assistant may ask, “You are seeing a movie with mom from 6-8pm tonight. Will you be having dinner with your mom after the movie?”

[0158] In some embodiments, if the user replies that he would be having the dinner with mom, as shown in FIG. 5E, the digital assistant optionally adds additional search criteria, such as the ambiance, price range, and/or cuisine type generally suitable for dinner with one’s mother, to the search request. In some embodiments, the digital assistant optionally uses the price, cuisine, and ambiance preferences of the user’s mother as additional search criteria for the search request, provided that such preferences are known to the digital assistant. In some embodiments, the digital assistant also adds a location criterion based on the home location of the user’s mom if the user has not specified that the dinner is to occur after the movie. In some embodiments, the digital assistant uses a location criterion based on both the home location of the user’s mom and the location of the movie theatre (*e.g.*, {restaurant location = “vicinity of Cupertino & Palo Alto”}). In some embodiments, as shown in FIG. 5E, the digital assistant presents these additional search criteria to the user for selection and approval before proceeding with the search. For example, the digital assistant optionally provide an output 520 showing the refined and/or additional search criteria generated based on the relevant calendar item (1). In some embodiments, the digital assistant also presents user interface elements (*e.g.*, check boxes, drop-down selection menus, editable ranges, etc.) for modifying and/or deleting the additional search criteria provided by the digital assistant. In some embodiments, the digital assistant also accepts verbal modifications of the refined and/or additional search criteria. In some embodiments, once the user has approved the search criteria presented by the digital assistant, *e.g.*, via a verbal approving command or a selection input on a “Proceed” button in the dialogue interface 504, the digital assistant proceeds with the augmented search.

[0159] In some embodiments, in lieu of and/or in addition to presenting the refinement to existing search criteria and/or the newly generated search criteria to the user, the digital assistant performs the search using the original search criteria of the search request, and filters the search results based on the refinement and/or additional search criteria. In such embodiments, the user can selectively review the respective search results satisfying each

augmented search request that includes a particular set of one or more refined search criteria and/or newly generated search criteria. As shown in FIG. 5F, the digital assistant presents a response 522 containing different categories (e.g., 524a-c) of search requests, each category corresponding to a respective set of search results satisfying a particular augmented search request based on the original search request and one or more relevant schedule item (e.g., calendar item (1)).

[0160] In this particular example, the digital assistant optionally presents the following three categories of search results: (1) restaurants with open tables before 6:00pm, (2) restaurants with open tables after 8:00pm, and (3) restaurants near the movie theatre. For the first category of search results, the digital assistant optionally filters the search results of the original search request using a refined time criterion {time= “before the movie event” = “before 6pm tonight”} and a preferred restaurant feature {feature= “*service speed*”}, with a preference order {preference order = “faster service speed is more preferable”}. This search augmentation is based on the assumption that the dinner is to occur before the movie and there isn’t a lot of time for the dinner. For the second category of search results, the digital assistant optionally filters the search results of the original search request using a refined time criterion {time= “after the movie event” = “after 8pm tonight”} and a preferred restaurant feature {feature= “*ambiance*”}, with a preference order {preference order = “good ambiance is more preferable”}. This search augmentation is based on the assumption that the dinner is to occur after the movie with the user’s mother. For the third category of search results, the digital assistant optionally filters the search results of the original search request using an additional location criterion {location = “near the movie theatre” = “near Palo Alto”}. Other categories of results can be presented depending on the particular embodiments used.

[0161] In some embodiments, the digital assistant presents the different categories of search results after presenting the relevant schedule item to the user. In some embodiments, the digital assistant highlights at least some of the refinement and/or additional search criteria to the user when presenting the different categories of search results to the user. In some embodiments, the user can select a particular highlighted search criterion and modify that search criterion directly. For example, in some embodiments, the user can edit “before 6:00pm” shown in the response 522, and change it to “before 5:30pm” or “before 6:00pm in Cupertino” and in response, a new group of search results meeting the user-modified search criteria will be presented to the user in place of the results 524a.

[0162] In some embodiments, the digital assistant uses the location in a schedule item to generate a new location criterion or modify an existing location criterion in the search request, when the digital assistant determines that the subject of the search request (*e.g.*, “to have dinner”) and the subject of the schedule item (*e.g.*, “to see movie”) are things to be done at proximate time(s). In this particular example, the search request does not explicitly include any location criterion, and the digital assistant uses the location of the calendar item (1) to generate the location criterion for the restaurant search request. In some embodiments, the digital assistant optionally presents an output to the user indicating the location search criterion (*e.g.*, “These are search results near the ABC movie theater.”). In some embodiments, if the search request does explicitly specify a location criterion (*e.g.*, “in Palo Alto”), and if the location of the calendar item (1) (*e.g.*, ABC theatre in Palo Alto) does not conflict with the explicitly specified location criterion, the digital assistant optionally uses the location of the calendar item (1) to refine the location criterion specified in the search request. For example, the digital assistant optionally augments the search request “Find me a restaurant in Palo Alto” to “Find me a restaurant near ABC theatre in Palo Alto.”

[0163] In some embodiments, the digital assistant generates a number of suggestions for the user’s subsequent action plan in light of the user’s search request and the relevant schedule items that have been identified. In some embodiments, the digital assistant presents the different suggestions and search results suitable for each of the suggestions to the user without first asking the user to clarify the search request in light of the relevant schedule items. For example, as shown in FIG. 5G, the digital assistant presents an output 526 stating, “You are seeing a movie with mom from 6-8pm. I have the following suggestions: (1) Dinner alone with mom before the movie, (2) Dinner with mom after the movie, and (3) Dinner alone after the movie.” For each of the suggested scenario, the digital assistant presents some suitable search results (*e.g.*, 528a-c) obtained or filtered based on a respective augmented search request. For example, for the first suggestion, the digital assistant optionally performs the search using a refined time criterion {time= “before the movie event” = “before 6pm tonight”} and a preferred restaurant feature {feature= “*service speed*”}, with a preference order {preference order = “faster service speed is more preferable”}. For the second suggestion, the digital assistant optionally performs the search using a refined time criterion {time= “after the movie event” = “after 8pm tonight”} and a preferred restaurant feature {feature= “*ambiance*”}, with a preference order {preference order = “good ambiance is more preferable”}. For the third suggestion, the digital assistant optionally performs the

search using an additional location criterion {location = “near the movie theatre” = “near Palo Alto”}.

[0164] In this example, when reviewing the second relevant schedule item that has been identified (*e.g.*, the reminder item (2) for picking up mom at 5:30pm in Cupertino), the digital assistant determines that the relevant time window associated with reminder item (2) is also proximate to the relevant time window associated with the search request 506. In some embodiments, the digital assistant presents this information to the user and asks the user to confirm the association between the search request 506 and the reminder item (2). For example, as shown in FIG. 5H, the user has answered the digital assistant’s inquiry regarding the timing of the dinner by saying “Before the movie.” In response, the digital assistant optionally presents an output 530 “You have a reminder to pick up mom at 5:30pm today, would you be having dinner with your mom?” If the user answers in the affirmative to this question, the digital assistant will use the associated person information in the reminder item (2) as an additional people criterion to augment the search request 506. For example, once the user confirms that his/her mom will be at the dinner, the digital assistant optionally adds additional search criteria related to the ambiance, price range, and/or cuisine type generally suitable for dinner with one’s mother to the search request 506. In some embodiments, the user’s mom’s preferences for restaurants are optionally used as additional search criteria for the restaurant search. For example, the digital assistant optionally uses the personal price, cuisine, and ambiance preferences of the user’s mother as additional search criteria for the search request 506, provided that such personal preferences are known to the digital assistant. In some embodiments, the digital assistant optionally changes the default preference order associated with a search criterion of the search request 506. For example, if the user’s usual preference order for restaurant search is that a lively and loud atmosphere is more preferable than a quiet and intimate atmosphere, in this particular scenario, however, the digital assistant optionally changes the preference order for the restaurant search 506 to indicate that a quiet, intimate atmosphere is more preferable than a lively and loud atmosphere.

[0165] In some embodiments, the digital assistant correlates the location and/or time information in multiple identified relevant schedule items with the location and/or time criteria in the search request, and attempts to reconcile any potential conflicts between them and/or streamline the process the user has to take to meet his/her schedule. In this particular example, the digital assistant determines that the calendar item (1), the reminder item (2), and the search request are associated with times and/or locations proximate to one another. In

some embodiments, the digital assistant presents this information to the user and tries to provide a recommendation that would accommodate all or most of the constraints imposed by the multiple relevant schedule items and the original search request 506. As shown in FIG. 5I, the digital assistant optionally presents an output 532 indicating that given the location and time constraints imposed by the multiple relevant schedule items (*e.g.*, the calendar item (1) and the reminder item (2)), only search results meeting a combined search criterion (*e.g.*, {restaurant location = “along the route from mom’s house in Cupertino to the movie theatre in Palo Alto”} and {dinner time = “between 5:30pm-6:00pm”}) would be presented to the user. In some embodiments, the output 532 states, “Since you only have half an hour for dinner before the movie, I will only recommend fast food restaurants on your route from mom’s house to the movie theatre.” In some embodiments, the digital assistant optionally highlights the combined search criterion generated based on the information extracted from the multiple relevant schedule items (*e.g.*, the calendar item (1) and the reminder item (2)). In some embodiments, the digital assistant presents the search results 534 obtained using the combined search criterion to the user for review and selection in the dialogue panel 504, as shown in FIG. 5I.

[0166] In some embodiments, once the digital assistant determines that multiple schedule items (*e.g.*, the calendar item (1) and the reminder item (2)) are proximate in locations and/or time to the search request, the digital assistant optionally presents the relevant schedule items (1) and (2) to the user, and asks the user to clarify whether the inferred purpose (*e.g.*, “to have dinner”) of the search request is to be carried out before the earliest schedule item, after the last schedule item, or between any adjacent pair of schedule items. In this particular example, the digital assistant optionally presents an output saying, “You have a reminder to pick up mom at 5:30pm, and a calendar item for see movie with mom from 6:00pm to 8:00pm tonight, would you like to have dinner (A) before picking up your mom, (B) before the movie with mom, or (C) after seeing the movie?” Each option presented to the user corresponds to a respective augmented search request with appropriate search criteria generated based on the scenario embodied in the options presented.

[0167] In some embodiments, the digital assistant presents the search results that best accommodate all the constraints obtained from the relevant schedule items, and provides a suggestion for an alternative plan of action (provided that the user has not previously rejected this alternative plan of action). For example, suppose that after the user provided the speech input “Find me a place for dinner tonight,” the digital assistant provided a response “You

have to pick up mom at 5:30, and see a movie with mom from 6-8pm. Would you like to have dinner (A) before picking up your mom, (B) before the movie with mom, or (C) after seeing the movie?” If the user responds “Before the movie with mom,” the digital assistant presents a response 536 based on the user’s clarification, and presents search result (*e.g.*, 538a-b) that are suitable for dinner with mom before the movie. In addition to presenting the results 538a-b according to the action plan provided by the user (*e.g.*, having dinner with mom before the movie), the digital assistant optionally recognizes that the action plan proposed by the user may not be ideal. Upon such recognition, the digital assistant optionally generates and presents an alternative action plan to the user. As shown in FIG. 5J, in addition to the response 536 and the search results 538, the digital assistant also provides an output 540 specifying an alternative action plan, and presents search results 542 obtained based on an augmented search request generated based on the alternative action plan. Specifically, the augmented search request based on the user’s proposed action plan includes a location criterion {restaurant location = “near the movie theatre or en route from the user’s mom’s house to the movie theatre”} and a feature criterion {feature = “fast service speed”}. The augmented search request based on the digital assistant’s alternative action plan includes a person criterion {party = “user + user’s mom”} and other criteria derived from the person criterion (*e.g.*, {ambiance = “quiet and intimate”}, {price = “\$\$-\$\$\$”}, {restaurant location = “vicinity of Cupertino and Palo Alto”}, and {time= “after 8pm tonight”}.

[0168] In some embodiments, if the user’s proposed action plan is to have dinner after picking up mom and before the movie, the digital assistant optionally determines that the amount of time available between the scheduled time for the movie and the reminder time for picking up mom is not sufficient for the inferred purpose of the search request (*e.g.*, “to have dinner”). For example, if the travel time between the pick-up location and the movie theatre is close to thirty minutes, there is not sufficient time to insert a dinner event between the two scheduled events (1) and (2). In such a scenario, the digital assistant optionally notifies the user of this scheduling conflict, and optionally suggests an alternative action plan. For example, as shown in FIG. 5K, the digital assistant presents an output 544, “You are seeing a movie with mom from 6-8pm in Palo Alto and you are picking up mom at 5:30pm in Cupertino. There is too little time for dinner with mom before the movie. Would you like to have dinner after the movie instead?”

[0169] In some embodiments, if the user insists on having the dinner event fit between picking up mom and seeing the movie, the digital assistant optionally generates a location

criterion for the restaurant search to only include restaurants along the route from the location specified in the reminder item (2) and the location specified in the calendar item (1). In addition, the digital assistant optionally add an additional constraints to ensure that the dinner can be completed within a short amount of time (*e.g.*, 30 minutes minus the necessary travel time between the location in the reminder item (1) and the location in the calendar item (1)). In some embodiments, the digital assistant optionally generates a restaurant type criterion to include only fast food chains with a drive-through option. In some embodiments, the digital assistant presents the option to the user before performing the search. For example, the digital assistant optionally says, “You would only have time for drive-through fast food on your way to the movie. Shall I proceed with the search for fast food restaurants that have a drive-through option along your route to the movie?”

[0170] In some embodiments, the user, upon the suggestion by the digital assistant (*e.g.*, via output 544), recognizes that the amount of time available for dinner between the time in the reminder item (2) and the time in the calendar item (1) is too short for a dinner, and decides to take the suggestion by the digital assistant. In such a scenario, the digital assistant optionally performs a search based on an augmented search request generated according to the alternative action plan proposed by the digital assistant, and presents the search results to the user. As shown in FIG. 5K, after the user accepted the alternative action plan proposed by the digital assistant, the digital assistant performs a search based on a set of search criteria suitable for dinner with mom after the movie (*e.g.*, {ambiance = “quiet and intimate”}, {price = “\$\$-\$\$\$”}, {restaurant location = “vicinity of Cupertino and Palo Alto”}, {time= “after 8pm tonight”})), and presents the search results 548 to the user in an output 546, as shown in FIG. 5K.

[0171] In some embodiments, if the digital assistant determines that the amount of time available for dinner between the time in the reminder item (2) and the time in the calendar item (1) is too short for a dinner, the digital assistant assesses whether either or both of the schedule items (1) and (2) can be shifted in time or location to accommodate the dinner. In some embodiments, the digital assistant optionally asks the user whether the location or time of either scheduled event may be moved to accommodate the dinner in between. In some embodiments, the digital assistant proposes the time and location change to the user in an output 550. For example, in the output 550, the digital assistant optionally states, “You are seeing a movie with mom from 6-8pm in Palo Alto, and you have scheduled to pick her up at

5:30 in Cupertino. There is too little time for dinner before the movie. Would you like to call mom to change the pick-up time to 4:30pm?”

[0172] In some embodiments, if the user agrees to change the scheduled pick-up time according to the assistant’s suggestion, the digital assistant proceeds to contact the participants to make the proposed changes known to the other participants. For example, after the user accepted the proposal by the digital assistant, the digital assistant proceeds to establish a telephone connection to the user’s mom. In some embodiments, the digital assistant listens in on the conversation between the user’s and the user’s mom to determine whether the proposal for change is accepted by both parties.

[0173] In some embodiments, once the digital assistant determines that the proposed changes to a schedule item are accepted by all participants of the schedule item, the digital assistant proceeds to make the changes to the schedule item in the user’s electronic schedule. The digital assistant also proceeds to perform the search based on an augmented search request in accordance with an action plan including the changed schedule item, as shown in FIG. 5L. In some embodiments, if the proposed change is not accepted (*e.g.*, if the attempt to reach the user’s mom is not successful, or if the conversation between the user and the user’s mom does not result in any changes to the schedule item), the digital assistant optionally present search results based on the constraints imposed by the relevant schedule items (*e.g.*, the calendar item (1) and the reminder item (2)). In some embodiments, the digital assistant provides more than one alternative changes to the relevant schedule items.

[0174] In some embodiments, the digital assistant optionally determines whether the user is the organizer of one or more of the relevant schedule items, and if so, the digital assistant determines that the user may be able to change the scheduled time for at least some of the schedule items. In some embodiments, the digital assistant also considers the identities of other participant(s) of the scheduled event, and determines whether the user may be able to change the scheduled event time based on the identities of the other participants. For example, if the other participants include only the user’s family members or personal contacts (*e.g.*, the user’s mother or friends), the digital assistant optionally asks the user to consider changing the scheduled time of at least one of the relevant schedule items. If the other participants include at least one professional contact, the digital assistant assumes by default that the scheduled time cannot be changed by the user.

[0175] In some embodiments, the digital assistant determines that the user is the organizer of an event, and assumes that the user may be able to change the location or time of the event if the user is the organizer of the event. In some embodiments, the digital assistant determines whether the user is a sole participant of the event, and assumes that the user may be able to change the location or time of the event if the user is the sole participant of the event. In some embodiments, the digital assistant determines whether the event is for a public event that the user participates in, and whether the time for participation is flexible. For example, a public event such as a movie is somewhat flexible in time and location because there are multiple show times and theatres locations that may work for the user. A public event such as a street fair is somewhat flexible in time but not in location, since the user can go to the street fair at any time during the scheduled time of the fair, but only at the scheduled location of the fair. A public event such as a sports game is flexible in neither time nor location, because the game that the user wishes to watch only occurs at its scheduled time and location.

[0176] In some embodiments, the digital assistant determines whether the user is a mandatory participant or optional participant of an event for which the user is an invitee. If the user is an invitee of a scheduled event, but is not a mandatory attendee, the digital assistant assumes that the user can be late to or absent from the event. If the user is a mandatory attendee, the digital assistant assumes that the user may not be able to change the location or time of the scheduled event.

[0177] In some embodiments, the digital assistant evaluates various combinations of the above factors to determine the likelihood that the user may be able to change the location and/or time of the relevant schedule item. In some embodiments, if the digital assistant determines that the user may be likely to change the location and/or time of the relevant schedule item, the digital assistant proposes a change to the location and/or time of the relevant schedule item to accommodate the inferred purpose of the search request. In some embodiments, the digital assistant generates one or more proposals for the time and/or location change for the relevant schedule item based on the subject, location, and time, and/or people associated with the other relevant schedule items.

[0178] The actions, interactions, and user interfaces described in the above examples in FIGS. 5A-5L are merely illustrative. Various combinations of the features described above are possible.

[0179] In some embodiments, the search request received by the digital assistant includes explicitly stated location, time, associated people, and domain for the constrained selection task. Sometimes, the original search request does not include an explicit location and/or time criterion. Although the digital assistant may infer a default or implicit location or time (*e.g.*, the user's current location or the current time) for a search request, such inference may result in an under-inclusion of potentially relevant schedule items. On the other hand, if no inference is drawn regarding the location or time criteria, the digital assistant may have to review schedule items associated with any locations and time as being potentially relevant to the search request. Therefore, when a location or time criterion is not explicitly stated in the original search request, there is a high likelihood of over-inclusion or under-inclusion of relevant schedule items in the initial set of potentially relevant schedule items.

[0180] To improve the efficiency and accuracy of the digital assistant's identification of relevant schedule items for a search request, it is important that a suitable initial set of potentially relevant schedule items is obtained first, and then the digital assistant can select the relevant schedule items from the initial set based on the overlap in location, time, subject matter, and/or associated people between the schedule item and the search request. FIGS. 6A-6C illustrate an exemplary process 600 for selecting an initial set of potentially relevant schedule items.

[0181] In some embodiments, in the process 600, to find the initial set of potentially relevant schedule items, the digital assistant determines (602) whether the search request specifies a distant location criterion and a distant time criterion. When the user explicitly specifies a remote location in the search request, it is likely that the user is interested in places and/or events associated with that remote location, and therefore, schedule items that are associated with local locations are not likely to be relevant to the search request. Similarly, when the user explicitly specifies a remote time (*e.g.*, a particular day apart from the present day) in the search request, it is likely that the user is interested in places and/or events associated with that particular day, and therefore, schedule items that are associated with the present day and other days are not likely to be relevant to the search request. In contrast, when the user is merely interested in events in the present timeframe (*e.g.*, the present day) and local places and events, the user frequently omits the location and time criteria in the search request. In further contrast, when the user is merely interested in local places for events occurring on a particular future date, the user sometimes omits the location criterion in the search request and only specifies the future date in the search request. In further contrast,

when the user is interested in places and events for a particular remote location for either the present day or a future date, the user sometimes omits the time criterion in the search request and only specifies the remote location in the search request.

[0182] Due to the subtle asymmetry and differences in user behavior described above, in some embodiments, depending on whether the search request specifies: (1) both a distant location criterion and a distant time criterion, (2) a distant location criterion, but no distant time criterion, (3) a distant time criterion, but no distant location criterion, (4) a local location criterion, but not any time criterion, (5) a local time criterion, but not any location criterion, or (6) neither location nor time criterion, the digital assistant optionally uses different strategies to select the initial set of potentially relevant schedule items.

[0183] In some embodiments, a distant location criterion can be found in the search request in the form of a location that is “remote” from the user’s current location. A distant time criterion can be found in the search request in the form of a time that is “remote” from the current time. The exactly standard for determining whether a location or time is “remote” can vary in different embodiments. In general, a “remote” location is a location that is not typically reachable during the user’s daily or routine movements. For example, regions that are not accessible by the user’s routine transportation means would be considered “remote.” For most people, regions outside of distances accessible by walking, driving, biking, public buses, and commuter trains, etc. from the user’s current or home location are considered “remote.” For some people who frequently travel by air between two particular cities, regions outside of distances accessible by walking, driving, biking, public buses, and commuter trains, etc. from these two particular cities would be considered “remote.” Typically, a “remote” time is a time outside of the present day (*i.e.*, the day that the user provided the search request).

[0184] In some embodiments, in accordance with a determination that the search request includes a respective distant time criterion and a respective distant location criterion: the digital assistant selects (604) a respective group of potentially relevant schedule items based on the respective distant time criterion and the respective distant location criterion, where each of the respective group of potentially relevant schedule items is associated with (1) a respective time window overlapping with a relevant time range specified by the respective distant time criterion, and (2) a respective geographic area overlapping with a relevant geographic location specified by the respective distant location criterion. For example, if the

user asks the digital assistant to “find a romantic restaurant in Atlanta for next Saturday,” and Atlanta is a remote location to the user, then, schedule items associated with locations other than Atlanta are unlikely to be relevant. Furthermore, among the schedule items associated with the remote location Atlanta, only schedule items that have a time window overlapping with next Saturday are likely to be relevant to the search request. Therefore, in some embodiments, the digital assistant identifies only schedule items that are associated with both the remote location “Atlanta” and the remote time “next Saturday” as being relevant to the search request. This filtering process performed by the digital assistant generates a small set of potentially relevant schedule items for further evaluation. As a result, the amount of data processing and the complexity of the processing tasks may be reduced. In addition, it is easier and more likely for the digital assistant to find the more relevant schedule items from the smaller set of potentially relevant items.

[0185] In this particular example, the user has specified a search request “Find a nice restaurant in Atlanta for next Saturday,” and the digital assistant has generated a constrained selection task based on the search request (*e.g.*, {selection domain= “restaurant”}, {location= “Atlanta”}, {time= “next Saturday”}, {keyword = “nice”}). Suppose that the digital assistant also identifies several schedule items associated with the user, including:

- (1) a calendar entry with {subject = “meet wife for dinner”}, {location = “Palo Alto”}, and {time= “today at 6pm”};
- (2) a reminder item with {subject = “flight to Atlanta”}, and {time = “next Thursday at 9pm”};
- (3) a calendar item with {subject = “meet with old friends”}, {location = “Atlanta”}, and {time = “next Saturday at 6pm”};
- (4) a to-do item with {subject = “buy gift for wife”}, {location = “Atlanta”}, {time = “before next Saturday”}; and
- (5) a calendar item with {subject = “Job interview”}, {time = “9am-2pm next Friday”}, and location= “Atlanta”}.

[0186] Among these five schedule items, the digital assistant eliminates schedule items that are associated with geographic areas that do not overlap with a relevant geographic area specified by the distant location criterion in the search request. For example, schedule item (1) (*e.g.*, calendar entry with {location = “Palo Alto”}) is eliminated from the set of potentially relevant schedule items for the search request. The digital assistant also

eliminates schedule items that are associated with time windows that do not overlap with a relevant time window specified by the distant time criterion in the search request. For example, schedule item (1) (*e.g.*, {time="today at 6pm"}, schedule item (2) (*e.g.*, {time="Thursday at 9pm"}), and schedule item (5) (*e.g.*, {time = "9am-2pm next Friday"} are also eliminated from the set of potentially relevant schedule items for the search request. The remaining schedule items (3) and (4) are each associated with a respective geographic area that overlaps with the relevant geographic area specified by the distant location criterion in the search request, and also associated with a respective time window that overlaps with the relevant time specified by the distant time criterion in the search request. Therefore, the digital assistant identifies each of schedule items (3), and (4) as a potentially relevant schedule item for the search request.

[0187] In some embodiments, once the digital assistant has identified the set of potentially relevant schedule items, the digital assistant optionally determines whether the respective time, location, associated people, subject matter, and/or other information in the potentially relevant schedule items can be used to refine the search request.

[0188] In this example, the selection domain of the search request is the restaurant domain, which overlaps with the subject matter of schedule item (3) (*e.g.*, "dinner with old friends"). In response to determining an overlap in subject matter, the digital assistant optionally infers that the search request is for accomplishing the activity expressed in the schedule item (3). In some embodiments, the digital assistant obtains one or more additional parameters from the schedule item (3), such as a time parameter = "6pm" and an attendee parameter = "Pete, John, and self." Based on the additional parameters obtained from the schedule item (3), the digital assistant optionally augments the search request with additional search criteria. For example, the augmented search request includes the following search criteria: {selection domain = "restaurant"}, {location = "Atlanta"}, {time = "6pm next Saturday"}, and {party size = "3"}.

[0189] In some embodiments, the digital assistant presents an output identifying the relevant schedule item used to augment the search request and the augmented search request. For example, in some embodiments, the digital assistant provides a speech output "You have a dinner scheduled for three people at 6pm next Saturday in Atlanta, would you like to find restaurants that have open table a party of three for 6pm?" In some embodiments, the digital assistant automatically proceed to filter the search results obtained for the user request using

the additional search criteria obtained from the relevant schedule item, and present the filtered search results to the user.

[0190] In some embodiments, the digital assistant identifies that the potentially relevant schedule item (4) does not provide further refinement to the specified location and time, and does not overlap with the search request in terms of subject matter and associated people. Upon such determination, the digital assistant does not use the information from schedule item (4) to change the search request.

[0191] In some embodiments, in accordance with a determination that the search request includes a respective distant location criterion and does not include any distant time criterion, the digital assistant selects (606) a respective group of potentially relevant schedule items based on the respective distant location criterion, where each of the respective group of potentially relevant schedule items is associated with (1) a respective geographic area overlapping with a relevant geographic location specified by the respective distant location criterion. For example, if the user asks “Where can I get some souvenirs in Atlanta?” Based on this user input, the digital assistant generates a search request with a distant location criterion {Location = Atlanta}, but no distant time criterion, and excludes only item (1) (*e.g.*, “meet wife for dinner today in Palo Alto”) from the set of potentially relevant schedule items.

[0192] In some embodiments, in accordance with a determination that the search request includes a respective distant time criterion and does not include any distant location criterion, the digital assistant selects (608) a respective group of potentially relevant schedule items based on the respective distant time criterion, where each of the respective group potentially relevant schedule items is associated with (1) a respective time window overlapping with a relevant time range specified by the respective distant time criterion. For example, if the user had asked “What movies are showing next Saturday?” instead of “Find a nice restaurant in Atlanta next Saturday,” the digital assistant would generate a search request with a distant time criterion {time = “next Saturday”}, but no distant location criterion. Based on this search request, the digital assistant would exclude schedule items that are not associated with “next Saturday” from the set of potentially relevant schedule items, but would not exclude items based on locations.

[0193] In some embodiments, in accordance with a determination that the search request does not include any distant location criterion or any distant time criterion, the digital assistant selects (610) a respective group of potentially relevant schedule items, where each

of the respective group of potentially relevant schedule items is associated with (1) a respective time window overlapping with a present time window of a given duration. For example, if the user input is “What’s happening in Mountain View today?” (provided that Mountain View is local to the user) or “What new movies have come out lately?,” the digital assistant automatically includes schedule items associated with the present day in the set of potentially relevant schedule items. In some embodiments, only items associated with the present day are included in the set of potentially relevant schedule items if the search request specifies both a local location criterion and a local time criterion.

[0194] In some embodiments, inclusion of the schedule items associated with the present day does not exclude inclusion of additional group(s) of schedule items in the set of potentially relevant schedule items. In other words, in some embodiments, if the search request includes neither a distant location criterion nor a distant time criterion, the digital assistant gradually expands the set of potentially relevant schedule items by including additional groups of schedule items. In contrast, in some embodiments, if the search request does include a distant location criterion and/or a distant time criterion, the digital assistant reduces the set of potentially relevant schedule items by excluding schedule items based on the distant location criterion and/or the distant time criterion (as shown in steps 604, 606, and 608 above).

[0195] In some embodiments, in accordance with a determination that the search request includes a local location criterion and does not include any time criterion, the digital assistant identifies (612) an additional group of potentially relevant schedule items, wherein each of the additional group of potentially relevant schedule items is associated with (1) an expanded time window around the present time window, and (2) a respective local geographic area that meets the local location criterion specified in the search request. For example, if the user asks “What’s a good Sushi place in Palo Alto” (provided that Palo Alto is local to the user), the digital assistant includes both schedule items associated with the present day regardless of location (*e.g.*, included in a first group), and schedule items associated with Palo Alto and within a few days after present day (*e.g.*, in the additional group) in the set of potentially relevant schedule items.

[0196] In some embodiments, in accordance with a determination that the search request includes a local time criterion and does not include any location criterion, the digital assistant identifies (614) an additional group of potentially relevant schedule items, wherein each of

the additional group of potentially relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective local geographic area surrounding the user's current location. For example, if the user asks "I want Sushi for lunch today" while the user is located in Palo Alto, the digital assistant includes both schedule items associated with the present day regardless of location (*e.g.*, included in a first group), and schedule items associated with Palo Alto (*e.g.*, in the additional group) in the set of potentially relevant schedule items.

[0197] In some embodiments, in accordance with a determination that the search request includes neither any time criterion nor any location criterion, the digital assistant identifies (616) one or more additional groups of potentially relevant schedule items. In some embodiments, the one or more additional groups include (618) an additional group of potentially relevant schedule items, where each of the additional group of potentially relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective local geographic area that matches a current location of the user. In some embodiments, the one or more additional groups include (620) an additional group of potentially relevant schedule items, where each of the additional group of potentially relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective group of participants that overlaps with a group of relevant people specified by a respective person criterion in the search request. In some embodiments, the one or more additional groups include (622) an additional group of potentially relevant schedule items, where each of the additional group of potentially relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective subject matter that relates to a respective selection domain specified in the search request.

[0198] The foregoing description, for purpose of explanation, has been described with reference to specific embodiments. However, the illustrative discussions above are not intended to be exhaustive or to limit the invention to the precise forms disclosed. Many modifications and variations are possible in view of the above teachings. The embodiments were chosen and described in order to best explain the principles of the invention and its practical applications, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated.

What is claimed is:

1. A method of operating a digital assistant, comprising:
at a device having one or more processors and memory:
receiving a search request from a user, the search request specifying one or more search criteria for selecting one or more selection items from a respective selection domain;
identifying one or more schedule items relevant to the search request;
augmenting the search request based on information contained in at least one of the identified schedule items; and
presenting the augmented search request or search results retrieved based on the augmented search request to the user.
2. The method of claim 1, wherein the relevant schedule items include one or more calendar items from a calendar associated with the user.
3. The method of claim 1, wherein the relevant schedule items further include one or more calendar items from a calendar that has been shared with the user by another user.
4. The method of claim 1, wherein the relevant schedule items further include one or more reminder items in the user's electronic schedule.
5. The method of claim 1, wherein the relevant schedule items further include or one or more to-dos in the user's electronic schedule.
6. The method of claim 1, wherein identifying the one or more schedule items relevant to the search request further comprises:
identifying one or more schedule items that are relevant to the search request in at least one of location, time, associated people, and subject matter.
7. The method of claim 1, wherein augmenting the search request based on the information contained in the at least one of the identified schedule items comprises:
establishing, based on the information contained in the at least one of the identified schedule items, a refinement on an existing criterion specified in the search request; and
modifying the search request based on the refinement on the existing criterion.

8. The method of claim 1, wherein augmenting the search request based on the information contained in the at least one of the identified schedule items comprises:

establishing, based on the information contained in the at least one of the identified schedule items, a new search criterion that has not been specified in the search request; and
modifying the search request to include the new search criterion.

9. The method of claim 1, wherein augmenting the search request based on the information contained in the at least one of the identified schedule items comprises:

establishing, based on the information contained in the at least one of the identified schedule items, a new preference order that has not been specified in the search request; and
modifying the search request to include the new preference order.

10. The method of claim 1, wherein augmenting the search request further comprises:

adding a new time criterion or modifying an existing time criterion based on an event time associated with the at least one of the identified schedule items.

11. The method of claim 10, adding the new time criterion or modifying the existing time criterion is further based on an event location associated with at least one of the identified schedule items.

12. The method of claim 1, wherein augmenting the search request further comprises:

adding a new location criterion or modifying an existing location criterion based on a respective location associated with at least one of the identified schedule items.

13. The method of claim 12, adding the new location criterion or modifying the existing location criterion is further based on an event time associated with at least one of the identified schedule items.

14. The method of claim 1, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying a respective group of relevant schedule items, wherein each of the respective group of relevant schedule items is associated with a respective time window overlapping with a present time window of a given duration.

15. The method of claim 14, wherein identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes neither any distant time criterion nor any distant location criterion.

16. The method of claim 14, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective local geographic region that meets a location criterion specified in the search request.

17. The method of claim 16, wherein identifying the additional group of relevant schedule items is performed in accordance with a determination that the search request includes a local location criterion and does not include any time criterion.

18. The method of claim 14, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective local geographic region that matches a current location of the user.

19. The method of claim 14, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective group of attendees that overlaps with a group of relevant people specified by a respective person criterion in the search request.

20. The method of claim 14, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying an additional group of relevant schedule items, wherein each of the additional group of relevant schedule items is associated with (1) a respective time window that does not overlap with the present time window, and (2) a respective subject matter that

relates to a relevant subject matter specified by a respective selection domain criterion in the search request.

21. The method of claim 18, 19, or 20, wherein identifying the additional group of relevant schedule items is performed in accordance with a determination that the search request does not include any location or time criterion.

22. The method of claim 1, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying a respective group of relevant schedule items based on a respective distant time criterion specified in the search request, wherein each of the respective group of relevant schedule items is associated with a respective time window overlapping with a relevant time range specified by the respective distant time criterion.

23. The method of claim 22, wherein identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes the respective distant time criterion and does not include any distant location criterion.

24. The method of claim 1, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying a respective group of relevant schedule items based on a respective distant location criterion specified in the search request, wherein each of the respective group of relevant schedule items is associated with a respective geographic location overlapping with a relevant geographic location specified by the respective distant location criterion.

25. The method of claim 24, wherein identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes the respective distant location criterion and does not include any distant time criterion.

26. The method of claim 1, wherein identifying the one or more schedule items relevant to the search request further comprises:

identifying a respective group of relevant schedule items based on the respective distant time criterion and the respective distant location criterion of the search request, wherein each of the respective group of relevant schedule items is associated with (1) a

respective time window overlapping with the relevant time range specified by the respective distant time criterion, and (2) a respective geographic location overlapping with the relevant geographic location specified by the respective distant location criterion.

27. The method of claim 26, wherein identifying the respective group of relevant schedule items is performed in accordance with a determination that the search request includes the respective distant time criterion and the respective distant location criterion.

28. The method of claim 1, wherein identifying the one or more schedule items relevant to the respective selection domain or the one or more selection criteria further comprises:

determining whether the one or more search criteria include any distant time criterion or any distant location criterion, wherein a distant time criterion specifies a relevant time range outside of a present time window of a given duration, and a distant location criterion specifies a relevant geographic region outside of a local geographic region of the user; and

based on an outcome of the determination, selecting a respective set of rules for identifying an initial set of potentially relevant schedule items from which the one or more relevant schedule items are identified for the search request.

29. A method comprising features of any combination of the method 1-28.

30. A non-transitory computer readable medium having instructions stored thereon, the instructions, when executed by one or more processors, cause the processors to perform any of the method of claims 1-28.

31. A system, comprising:
one or more processors; and
memory having instructions stored thereon, the instructions, when executed by the one or more processors, cause the processors to perform any of the method of claims 1-28.

32. A method of operating a digital assistant, comprising:

at a device having one or more processors and memory:

receiving a speech input from a user;

processing the speech input to infer a respective user intent, the user intent being embodied in one or more search criteria for a search request;

based on the search request, generating a constrained selection task for selecting one or more selection items from a respective selection domain based on the one or more search criteria;

identifying one or more schedule items relevant to the search request based on the respective selection domain and the one or more search criteria;

augmenting the search request based on information contained in at least one of the identified schedule items; and

presenting the augmented search request or search results retrieved based on the augmented search request to the user.

33. The method of claim 32, further comprising features of any combination of the method 1-28.

34. A non-transitory computer readable medium having instructions stored thereon, the instructions, when executed by one or more processors, cause the processors to perform any of the method of claims 32-33.

35. A system, comprising:
one or more processors; and
memory having instructions stored thereon, the instructions, when executed by the one or more processors, cause the processors to perform any of the method of claims 32-33.

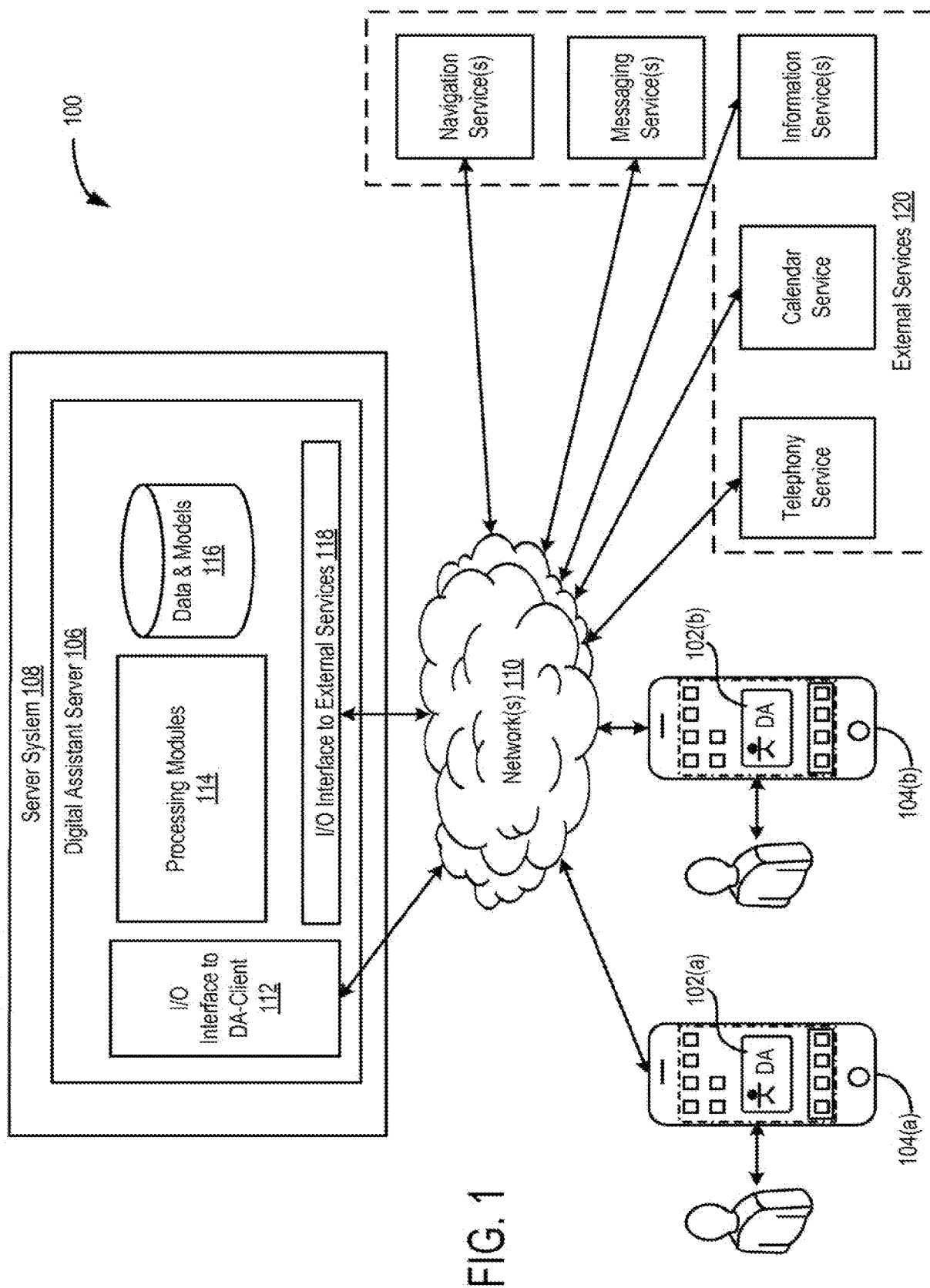


FIG. 1

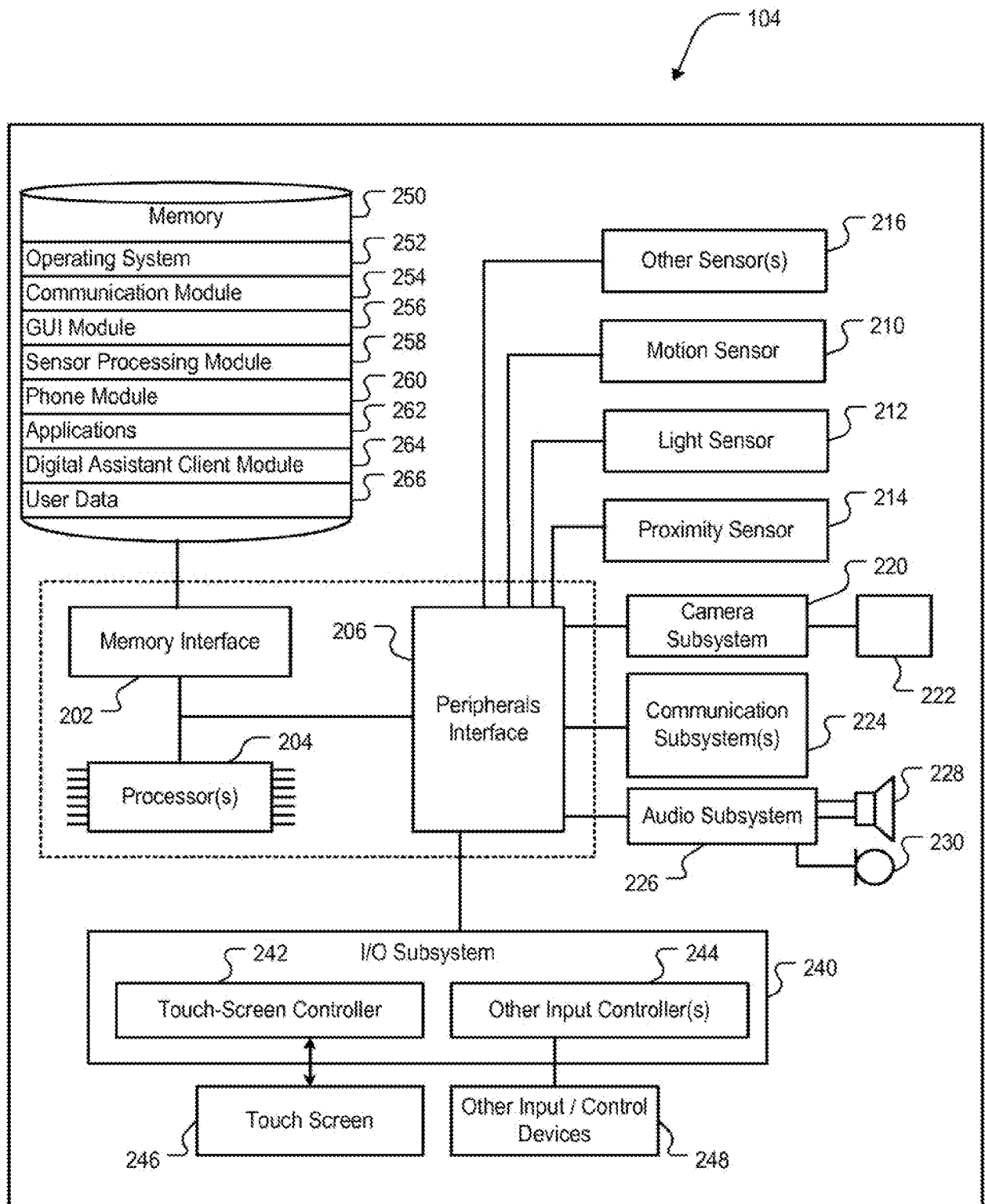


FIG. 2

FIG. 3A

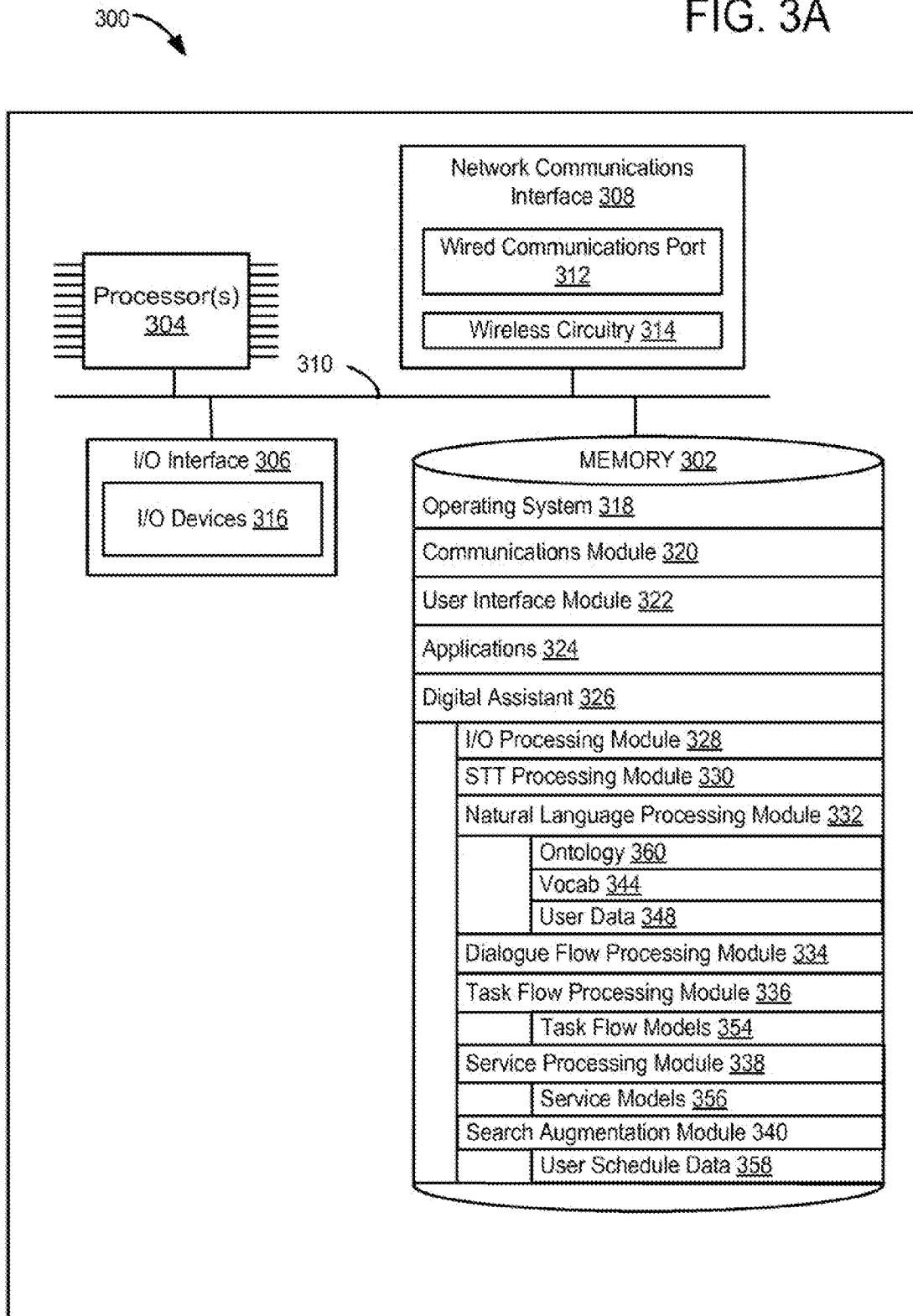
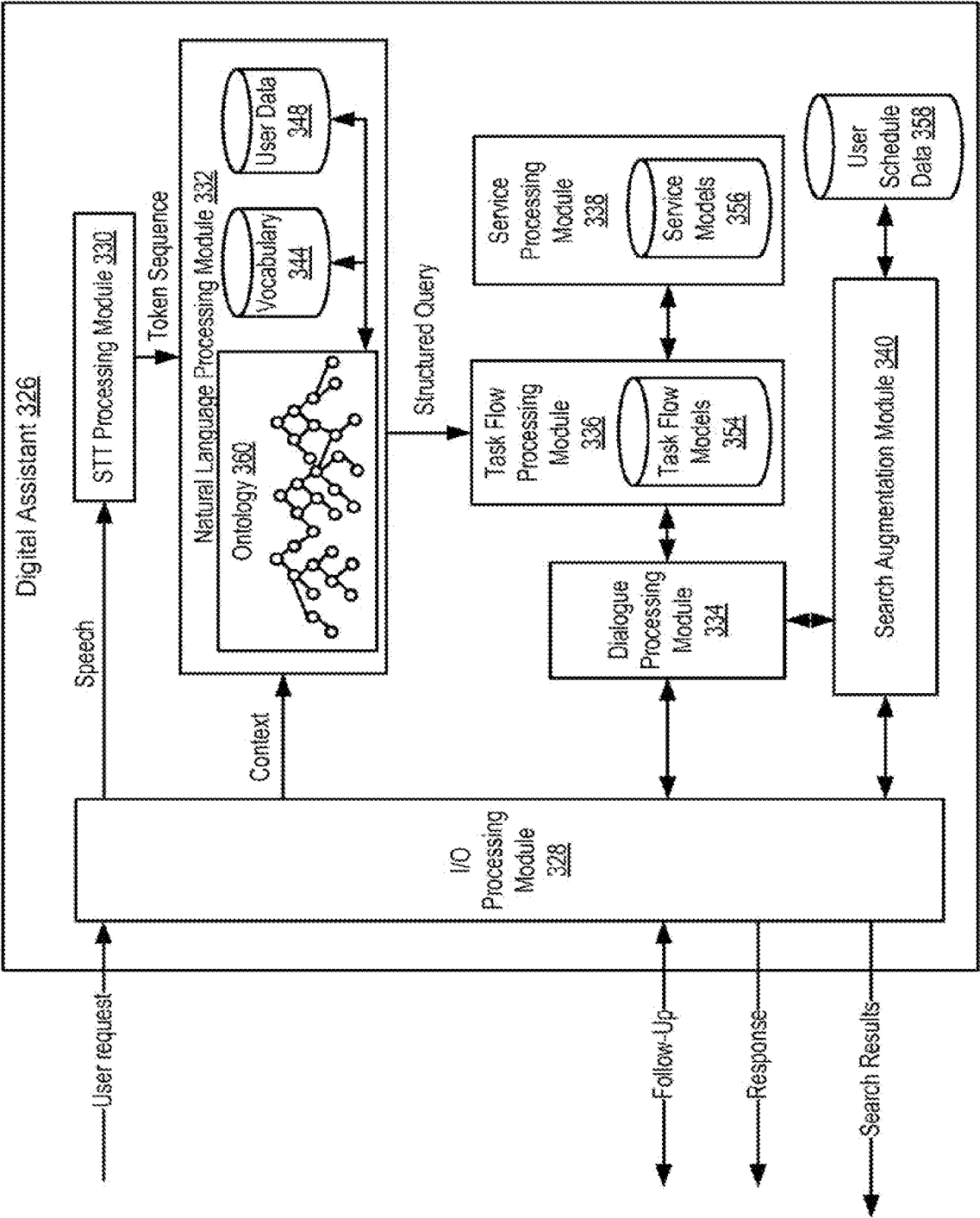


FIG. 3B



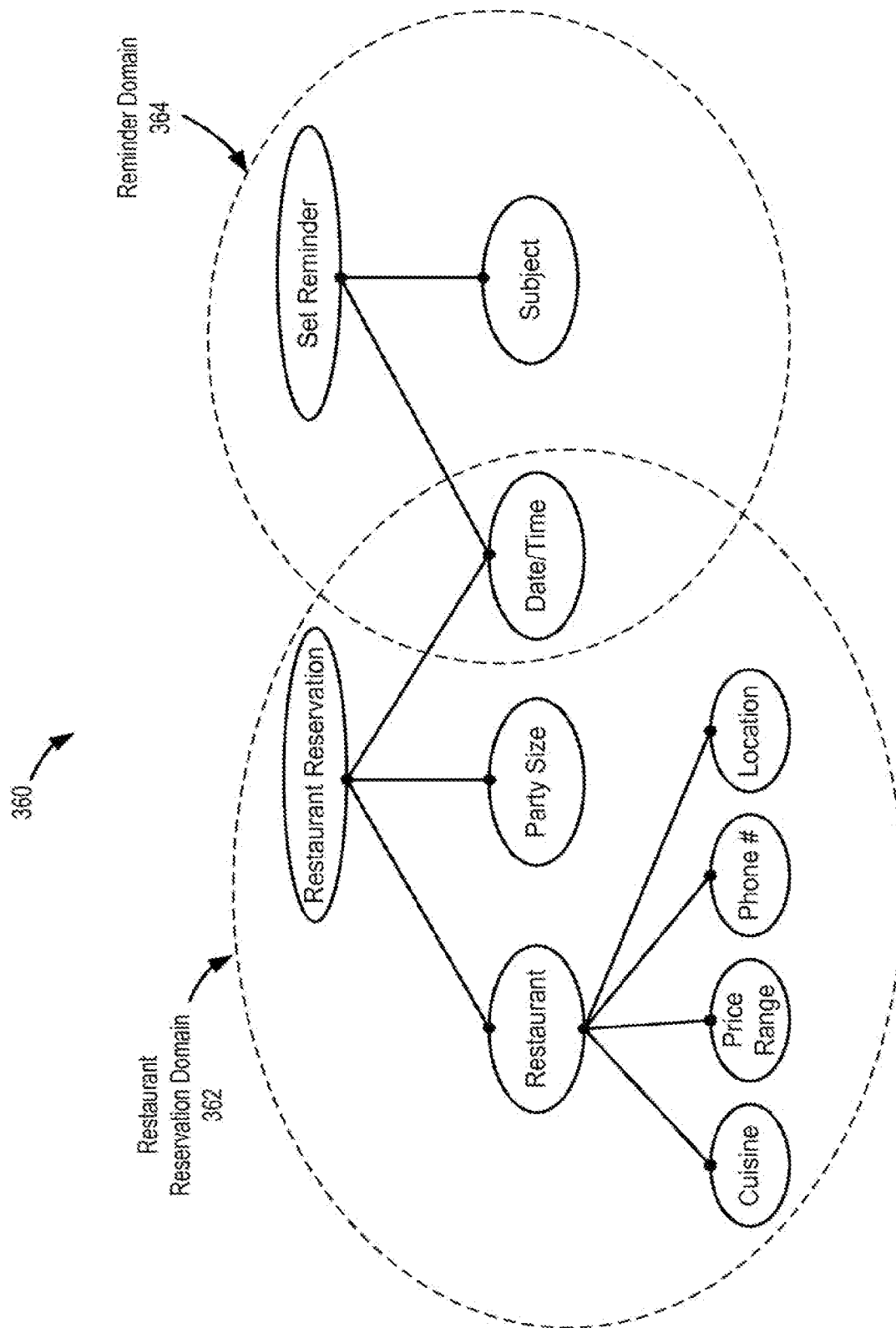


FIG. 3C

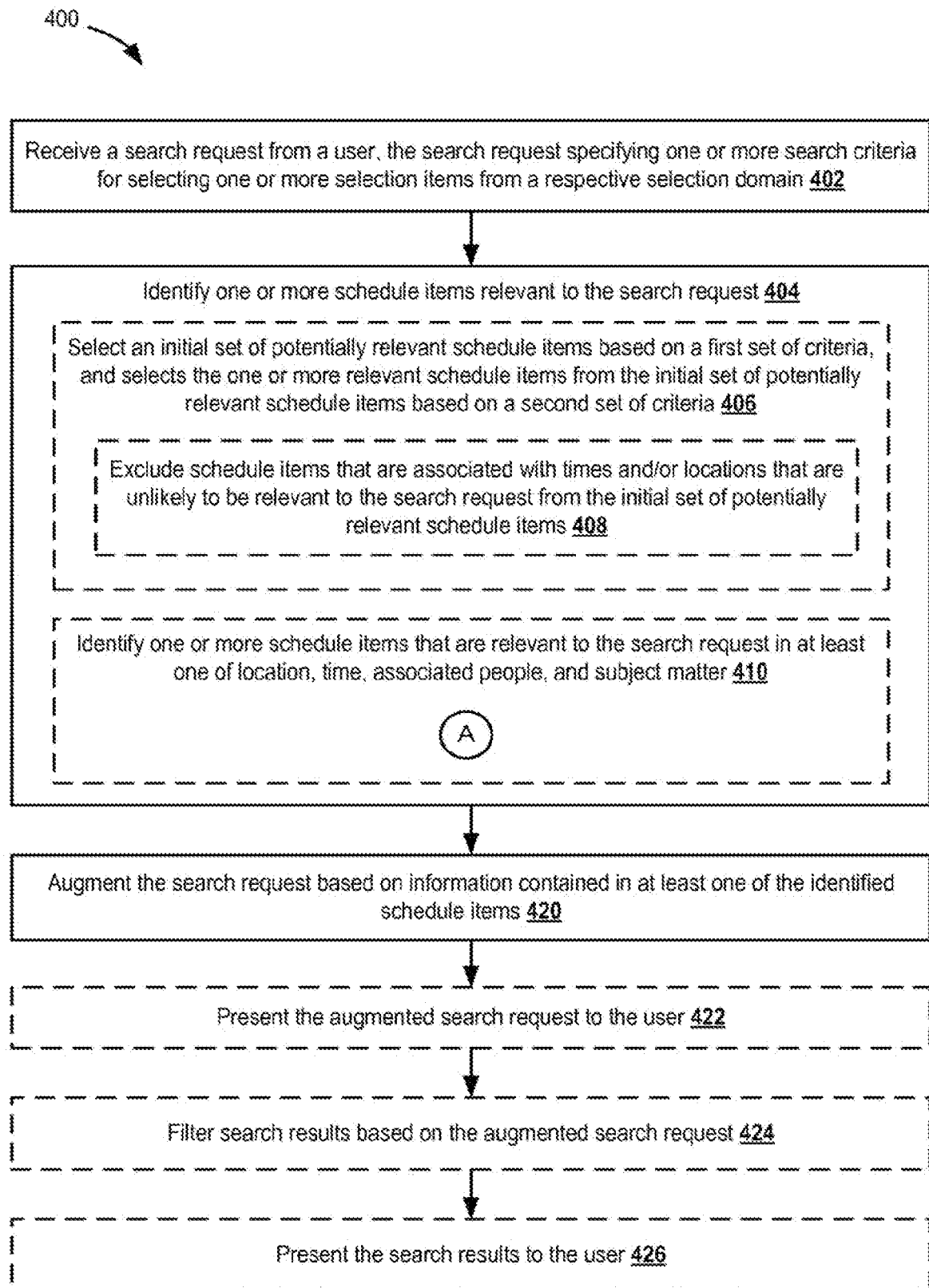


FIG. 4A

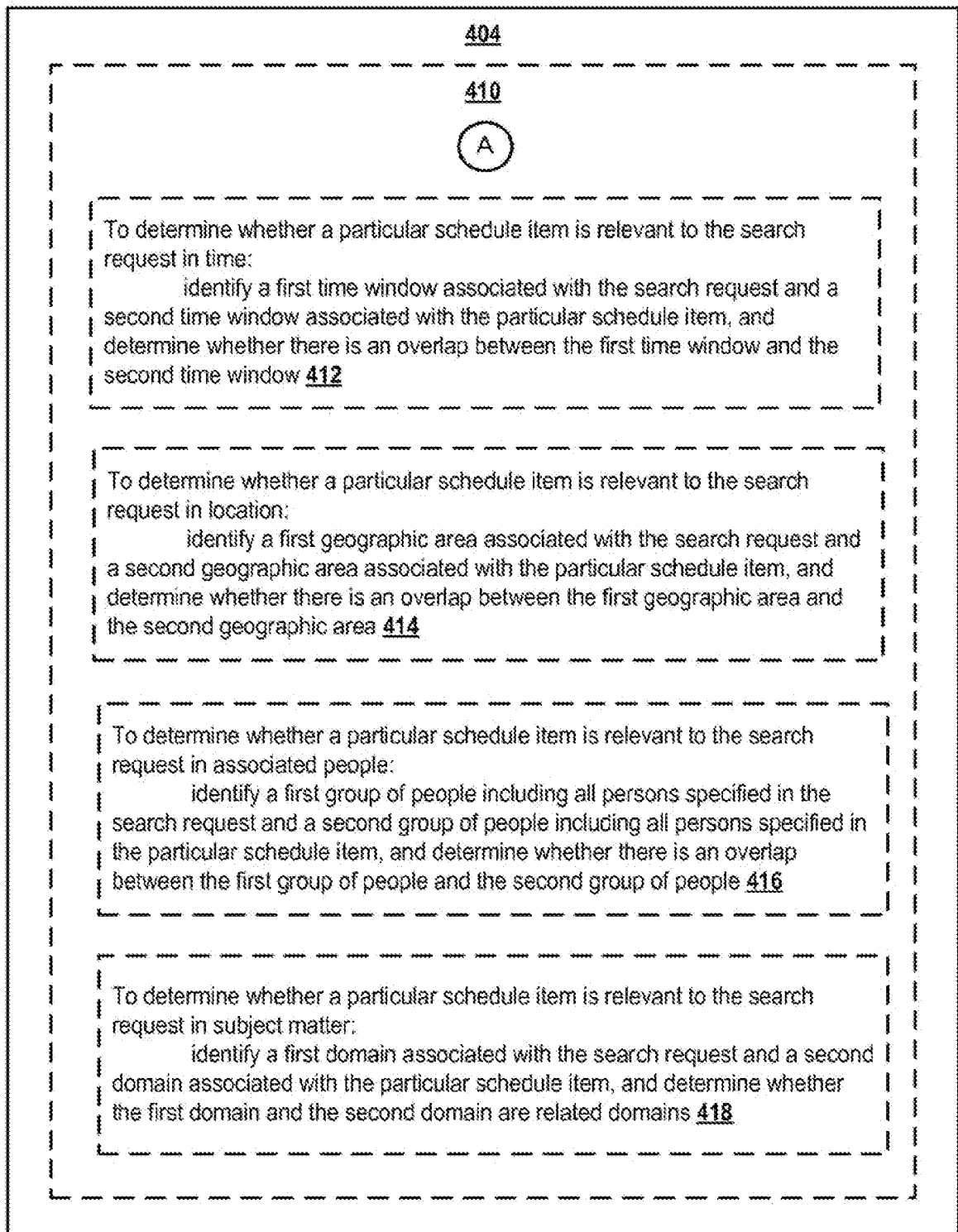
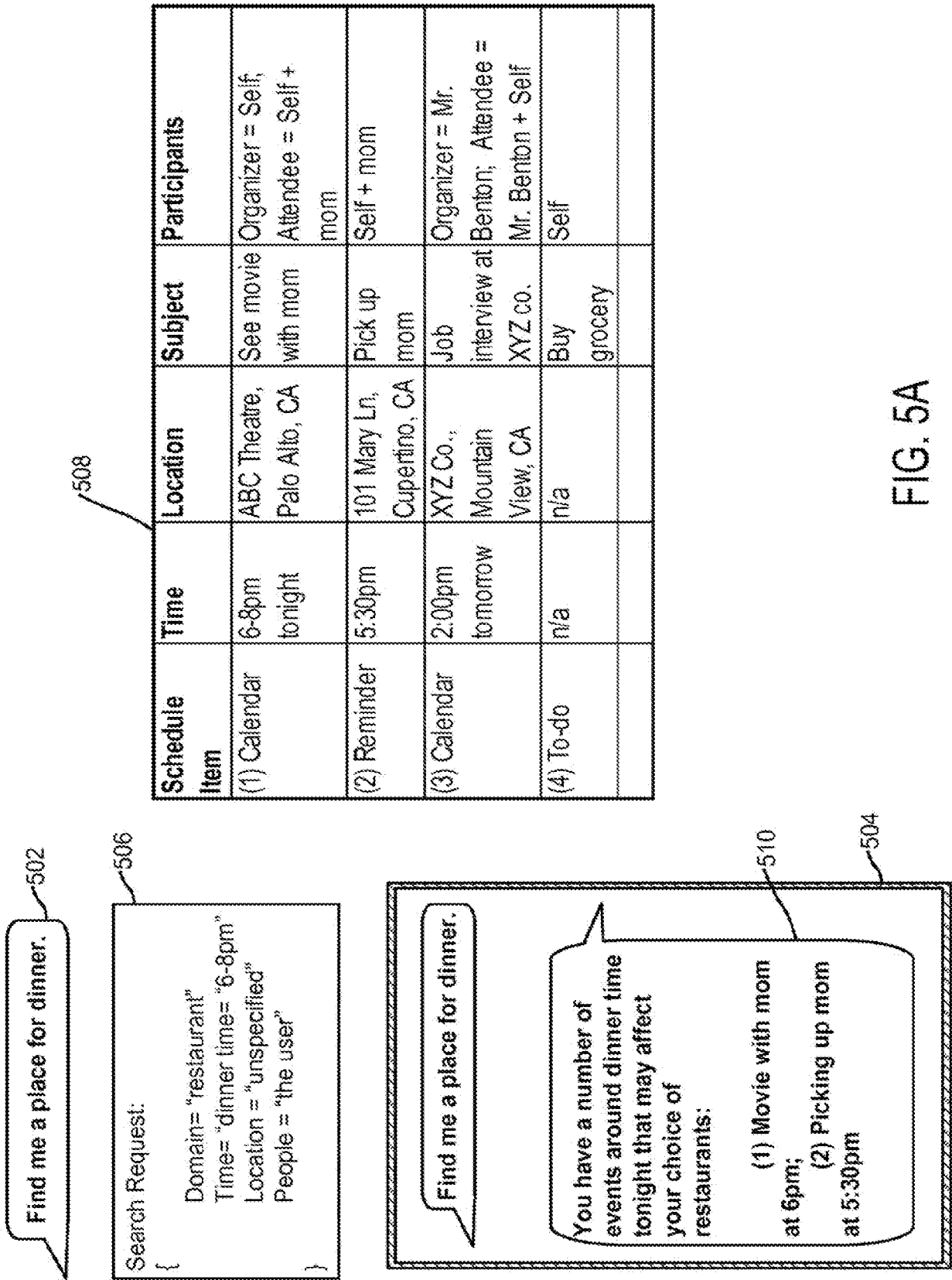


FIG. 4B



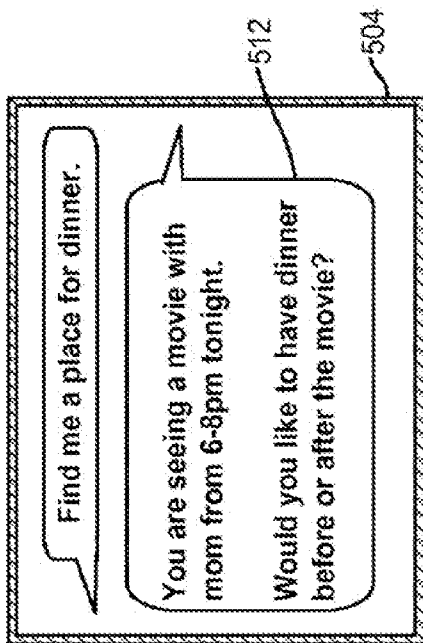


FIG. 5B

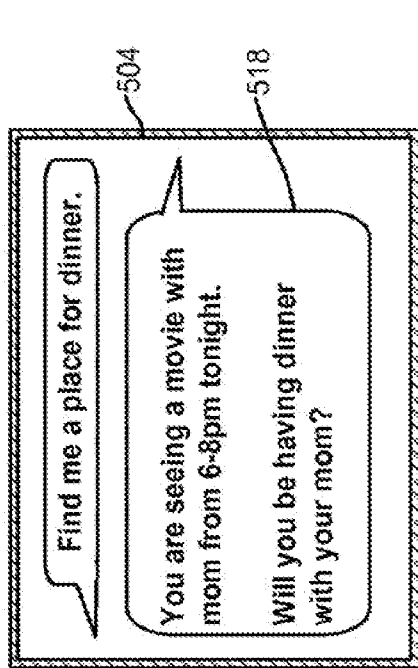


FIG. 5D

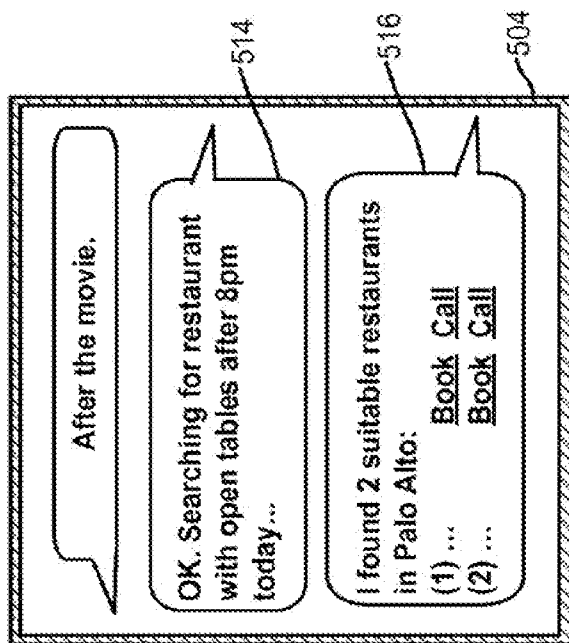


FIG. 5C

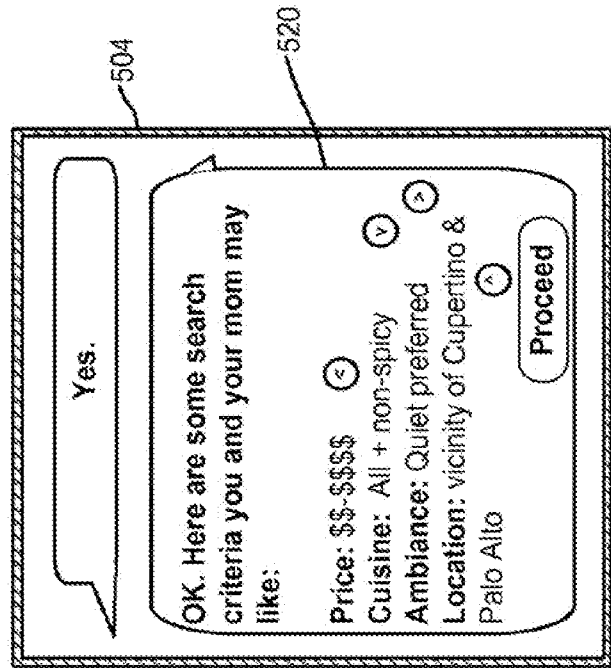


FIG. 5E

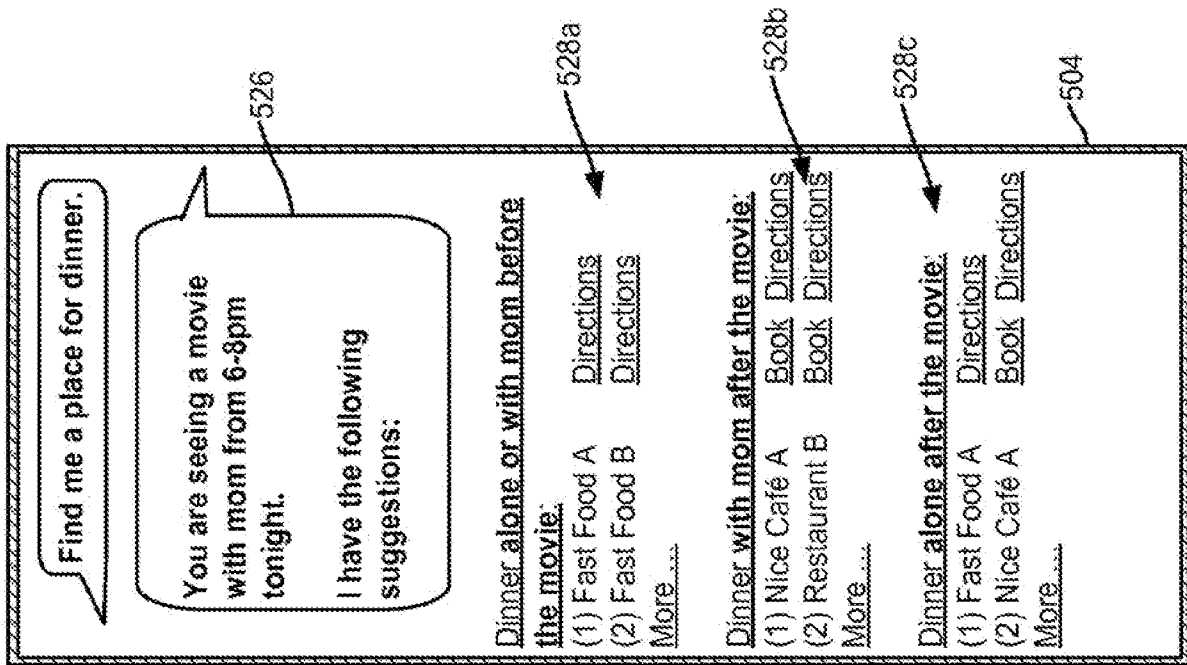


FIG. 5G

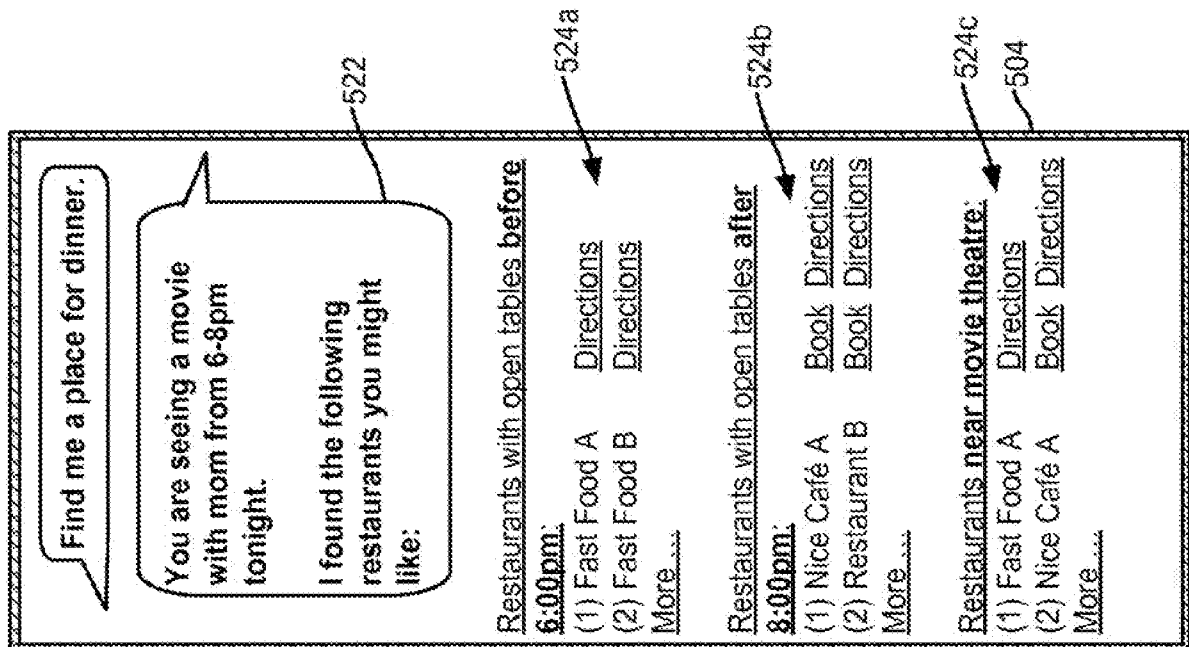


FIG. 5F

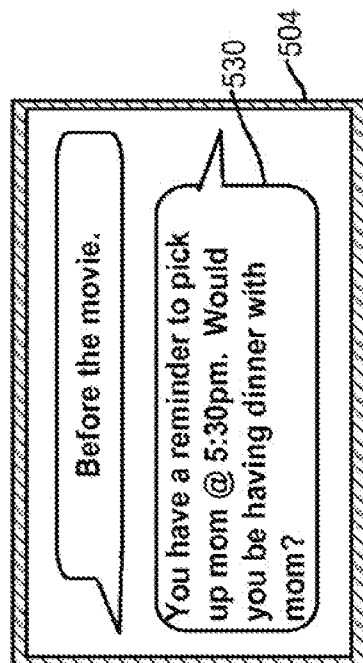


FIG. 5H

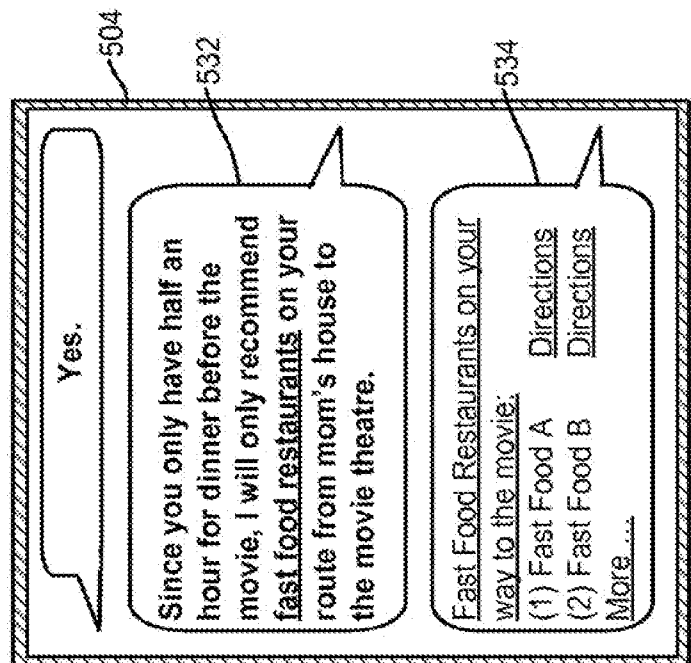


FIG. 5I

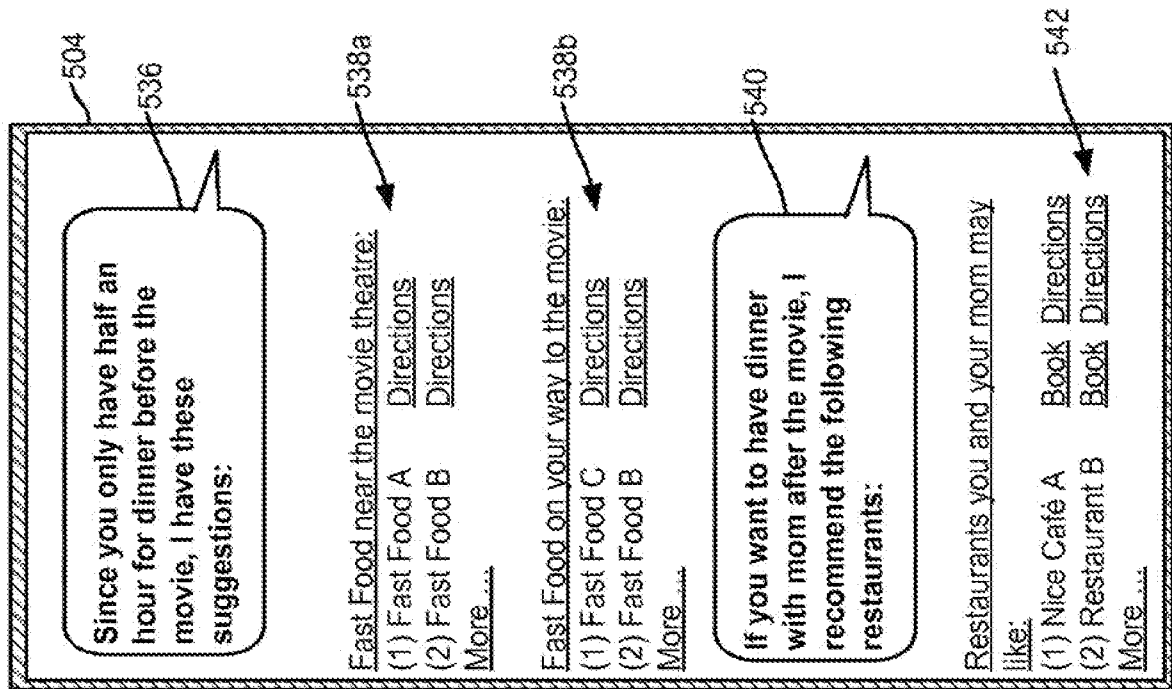


FIG. 5J

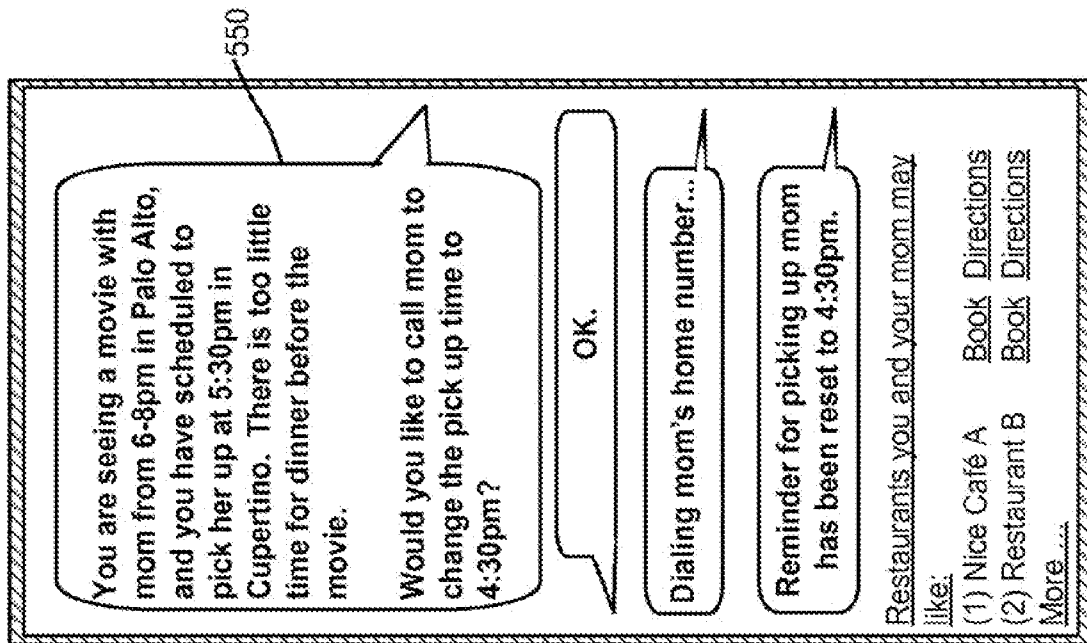


FIG. 5L

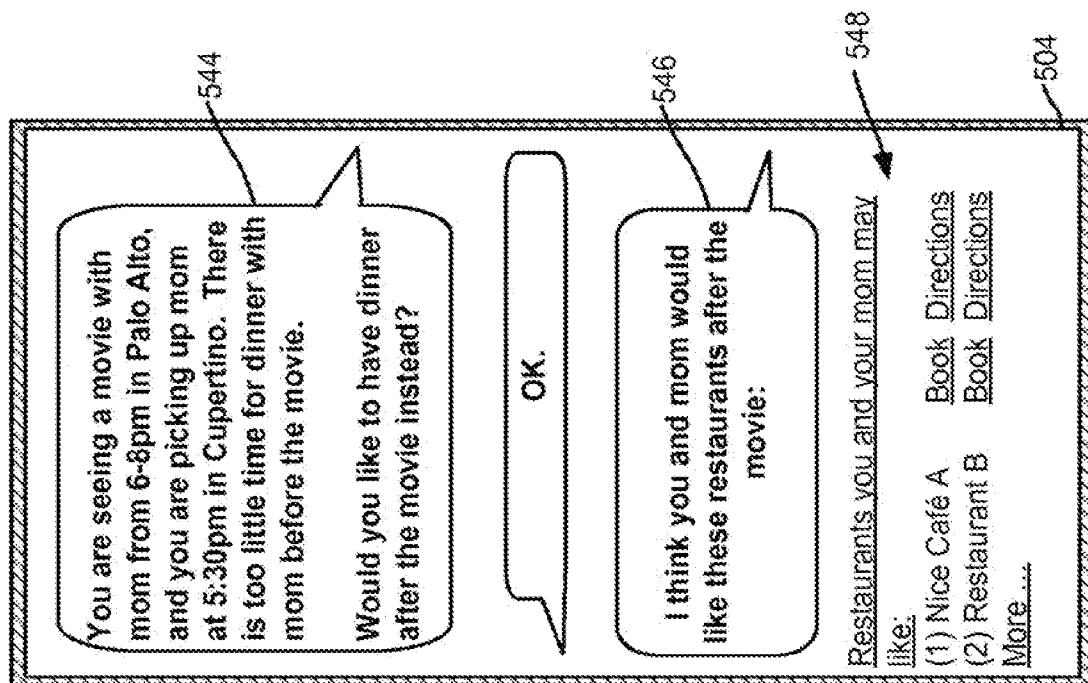


FIG. 5K

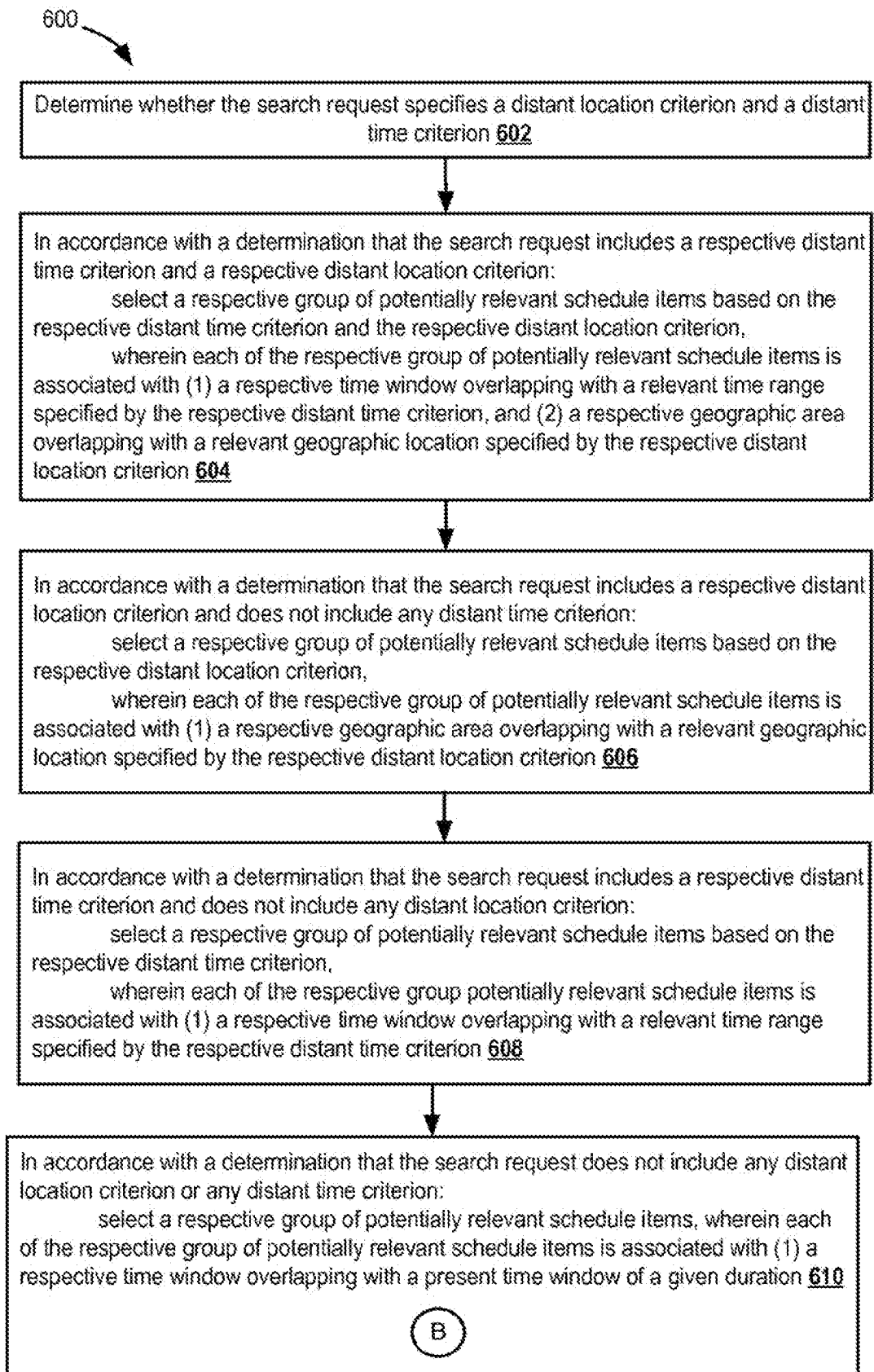


FIG. 6A

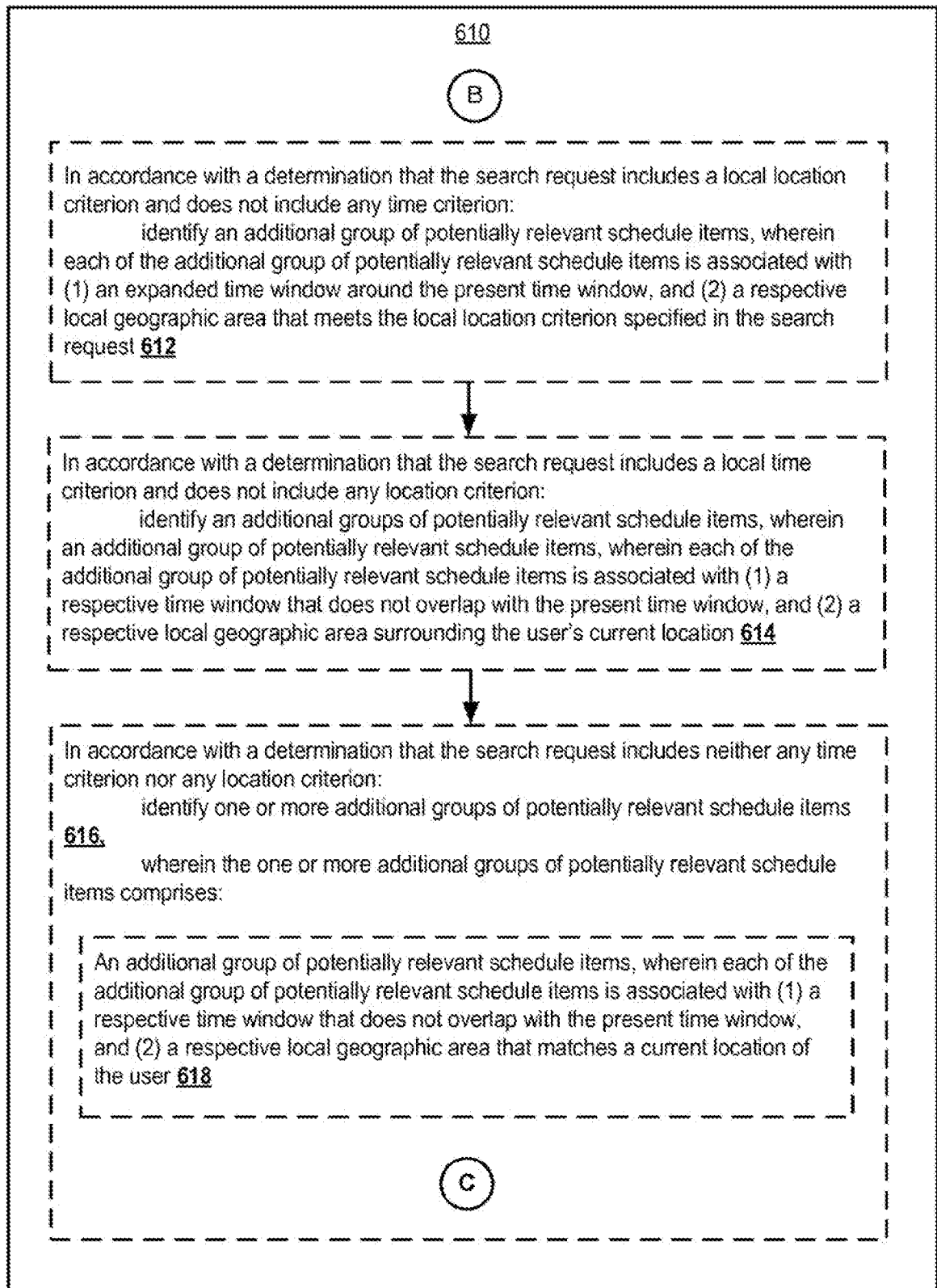


FIG. 6B

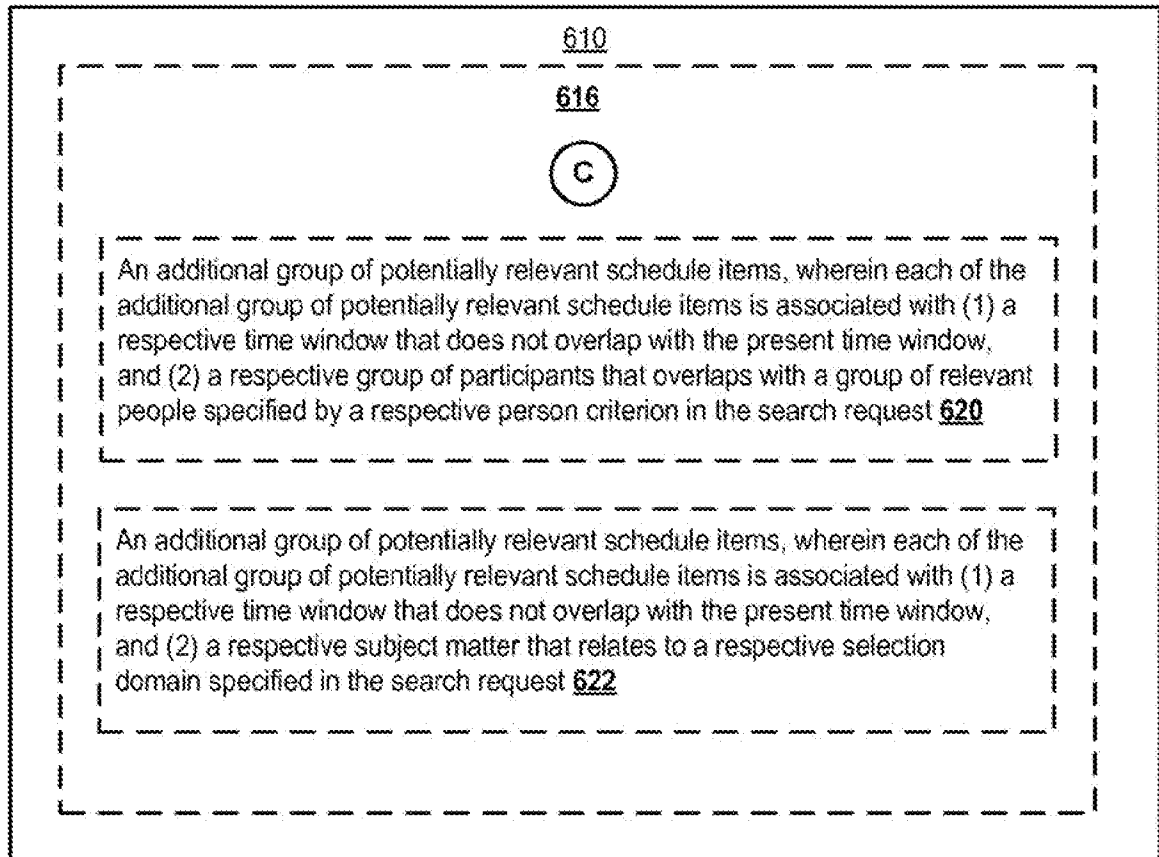


FIG. 6C